Helsinki University of Technology, Electronic Circuit Design Laboratory
Espoo 2006 Report 43

DIGITAL SIGNAL PROCESSING AND DIGITAL-TO-ANALOG CONVERTERS FOR WIDE-BAND TRANSMITTERS

Marko Kosunen

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Department of Electrical and Communications Engineering for public examination and debate in Auditorium S4 at Helsinki University of Technology (Espoo, Finland) on the 10th of November, 2006, at 12 noon.

Helsinki University of Technology
Department of Electrical and Communications Engineering
Electronic Circuit Design Laboratory

Teknillinen korkeakoulu
Sähkö- ja tietoliikennetekniikan osasto
Piiritekniikan laboratorio

Distribution:
Helsinki University of Technology
Department of Electrical and Communications Engineering
Electronic Circuit Design Laboratory
P.O.Box 3000
FIN-02015 HUT
Finland
Tel. +358 9 4512271
Fax: +358 9 4512269

ISBN-10 951-22-8409-X
ISBN-10 951-22-8410-3 (PDF)
ISBN-13 978-951-22-8409-2
ISBN-13 978-951-22-8410-8 (PDF)
ISSN 1455-8440

Otamedia Oy
Espoo 2006

Abstract

In this thesis, the implementation methods of digital signal processing and digital-to-analog converters for wide-band transmitters are researched. With digital signal processing, the problems of analog signal processing, such as sensitivity to interference and nonidealities of the semiconductor processes, can be avoided. Also, the programmability can be implemented digitally more easily than by means of analog signal processing.

During the past few years, wireless communications has evolved from analog to digital, and signal bandwidths have increased, enabling faster and faster data transmission. The evolution of semiconductor processes, decreasing linewidth and supply voltages, has decreased the size of the electronics and power dissipation, enabling the integration of larger and larger systems on single silicon chips.

There is little overall benefit in decreasing linewidths to meet the needs of analog design, since it makes the design process more difficult as the device sizes cannot be scaled according to minimum linewidth and because of the decreasing supply voltage. On the other hand, the challenges of digital signal processing are related to the efficient realization of signal processing algorithms in such a way that the required area and power dissipation does not increase extensively.

In this book, the problems related to digital filters, upconversion algorithms and digital-to-analog converters used in digital transmitters are researched. Research results are applied to the implementation of a transmitter for a third-generation WCDMA base-station.

In addition, the theory of factors affecting the linearity and performance of digital-to analog converters is researched, and a digital calibration algorithm for enhancement of the static linearity has been presented. The algorithm has been implemented together with a 16-bit converter; its functionality has been demonstrated with measurements.

Keywords: Direct digital synthesizer, Digital transmitter, modulator,
CORDIC, digital-to-analog converter, calibration.

Tiivistelmä

Tässä väitöskirjassa on tutkittu digitaalisen signaalinkäsittelyn toteuttamista ja digitaalisesta analogiseksi-muuntimia laajakaistaisiin lähettimiin. Digitaalisella signaalinkäsittelyllä voidaan välttää monia analogiseen signaalinkäsittelyyn liittyviä ongelmia, kuten häiriöherkkyyttä ja puolijohdeprosessien epäideaalisuuksien vaikutuksia. Myös ohjelmoitavuus on helpommin toteutettavissa digitaalisesti kuin analogisen signaalinkäsittelyn keinoin.

Viime vuosina on langattomien tietoliikennejärjestelmien kehitys kulkenut analogisesta digitaaliseen, ja käytettävät signaalikaistanleveydet ovat kasvaneet mahdollistaen yhä nopeamman tiedonsiirron. Puolijohdeprosessien kehitys, kapeneva minimiviivanleveys ja pienemmät käyttöjännitteet, on pienentänyt elektroniikan kokoa ja tehonkulutusta mahdollistaen yhä suurempien kokonaisuuksien integroimisen yhdelle piisirulle. Viivanleveyksien pieneneminen ei kuitenkaan suoraan hyödytä analogiasuunnittelua, jossa piirielementtien kokoa ei välttämättä voida pienentää viivanleveyden pienentyessä, ja jossa madaltuva käyttöjännite ennemminkin hankaloittaa kuin helpottaa suunnittelua. Siksi yhä suurempi osa signaalinkäsittelystä pyritään tekemään digitaalisesti. Digitaalisen signaalinkäsittelyn ongelmat puolestaan liittyvät algoritmien tehokkaaseen toteuttamiseen siten, että piirien pinta-ala ja tehonkulutus eivät kasva liian suuriksi.

Tässä kirjassa on tutkittu digitaalisessa lähettimessä tarvittavien digitaalisten suodattimien, ylössekoitusalgoritmien ja digitaalisesta analogiseksi-muuntimien toteuttamiseen liittyviä ongelmia. Tutkimustuloksia on sovellettu kolmannen sukupolven WCDMA-tukiasemalähettimen toteutuksessa.

Lisäksi on tutkittu digitaalisesta analogiseksi-muuntimien lineaarisuuteen ja suorituskykyyn vaikuttavien seikkojen teoriaa, ja esitetty digitaalinen kalibrointialgoritmi muuntimen staattisen suorituskyvyn parantamiseksi. Algoritmi on toteutettu 16-bittisen muuntimen yhteydessä ja se on osoitettu toimivaksi mittauksin.

Avainsanat: Suora digitaalinen syntetisaattori, digitaalinen lähetin, modulaattori, CORDIC, digitaalisesta analogiseksi-muunnin, kalibrointi.

Preface

The research for this thesis was carried out at the Electronic Circuit Design Laboratory (ECDL) of Helsinki University of Technology during the years 1998-2005. The work presented in this book has been carried out in research projects funded by Nokia Networks, Nokia Research Center and Finnish National Technology Agency. I also thank the Nokia Foundation, the Finnish Society of Electronics Engineers, and the Foundation of Technology for their financial support.

I express my gratitude to Professor Kari Halonen for his guidance and support, and for the opportunity to carry out this research in such an inspiring and relaxed environment as the Electronic Circuit Design Laboratory. My warmest thanks go to Dr. Jouko Vankka, my supervisor, for his guidance and new ideas during the years we worked together. I would also like to express my gratitude to Jussi Pirkkalaniemi for his contribution to the research presented in this book. I also warmly thank Professors Jiren Yuan and Trond Ytterdal for reviewing this thesis, and for their valuable comments and suggestions.

I gratefully acknowledge all my colleagues with whom I have been honored to work throughout the years. Special thanks go to Mikko Waltari, Jonne Lindeberg, Väinö Hakkarainen and Jacek Flack for the very interesting conversations we had during the years through which we shared out room. In addition to those mentioned above, I would like to express my sincerest gratitude to Rami Ahola, Asko Kananen, Jaakko Ketola, Kimmo Koli, Lauri Koskinen, Saska Lindfors, Mika Länsirinne, Esa Tiiliharju, Ari Paasio, Jussi Pirkkalaniemi and Helena Yllö for their professional assistance, and, most of all, for their active participation in our not-so-professional recreational activities, including our good-doing at 4 a.m., the endless planning of trips to the museum of former president Kekkonen, and listening to my never-ending environmental preaching. Special thanks also to Marja and Outi.

I would like to thank my former teachers Varpu Rönnholm, Kaarina Koskinen, and Eino Sandelin for guidance they have given me, and, express my warmest thanks to all my friends, with whom I have shared many great moments throughout the years.

Pikkutomi, Tumppi, Attila, Kimmo and the other ”Boys of Puotila”, Natski, Jomi, Oippa, Pamppu, Kartsa et al., thanks for the fun, and for keeping me relatively sane.

I would also like to thank my in-laws for their support, beer, great cooking, beautiful daughter, and great sense of humor.

I would like to thank my mother and father for their love, teachings, and support. Thanks also to my sister Marjo and my nephew Antti.

The rest of this page was reserved for the praise of my intelligent and good-looking wife, and my beautiful and loving daughters. Being an engineer, not a poet, my writings seem to turn out to be either dimwitted or naive. Therefore I chose to be short and compact. Katri, Auri, and Ira, I could not love you more. Thanks for your existence.

Puotila, Helsinki, September 2006

Marko Kosunen

Contents

Chapter 1

Introduction

1.1 Motivation

The rapid growth of the wireless communications market has been the primus motor of the development of integrated circuit technology during the past decade. This is because the portable electronic devices for wireless communication has to be optimized for low power consumption, light weight, and low manufacturing cost, and the fact that these requirements can be met simultaneously by increasing the integration level of the electronics. The minimum line-width of the semiconductor processes has seemed to be ever-decreasing, enabling the integration of increasingly complicated systems on a single chip.

Even though the evolution of the silicon processes has reduced the minimum line width, operation voltages and, thereafter, the power dissipation of the digital parts, the devices required for analog processing have not scaled along with the minimum line-width, and the decreasing operation voltages tends to make the analog circuit design more challenging.

In wireless communication systems, the trend has been to move from analog to digital signal processing and increase the bandwidth. Wireless digital communications have evolved from GSM through services such as GPRS and EDGE towards WCDMA and 3G, which are capable of handling both the narrow voice band and wide data bands. Simultaneously, wireless data transmission systems such as WLAN/WiFi and Wimax have gained popularity. It is beneficial to perform most of the computation of the system in the digital the domain. Analog signal processing is sensitive to noise generated in several sources, whereas the accuracy of the digital signal processing (DSP) may be selected almost arbitrarily. Digital signal processing also enables flexibility in the system, since programmability and reconfigurability can be implemented more easily digitally. Benefits of flexibility are obvious in systems like transmitters and receivers,

in which the usage of DSP enables the realization of multi-mode transmitters such as GSM/EDGE/WCDMA or software-configurable radio.

Digital signal processing can be performed with a general signal processor; however, it is not capable of handling the high data rates typical of digital-IF transmitters. A dedicated DSP system for transmitter purposes is therefore usually used. The first part of this work is based on research into hardware efficient realization methods of the digital signal processing required for base-band and the intermediate frequency of the multi-carrier wide-band code division multiple access (WCDMA) base-station transmitter of the 3^{rd} generation (3G) wireless communication system. This system is also often referred to as the Universal Mobile Telecommunications System (UMTS). Area efficient digital IF transmitters reduce the manufacturing cost of the transmitter chip, decrease power consumption, and thus reduce the need for cooling and maintenance. The second part of this work is based on research into the current-steering digital-to-analog converters, which are the performance bottleneck of the digital-IF transmitter. Research results are applied in the design of two prototype circuits, the first consisting of the digital-IF WCDMA transmitter and the second being a current-steering D/A converter with a digital calibration algorithm.

1.2 Organization of the thesis and research contribution

The research of the digital-IF transmitters presented in this book was performed during the years 1998-2001 under the supervision of Dr. Jouko Vankka, who also performed most of the algorithm design of the transmitter prototype. The author participated in the algorithm design and is responsible for the design and implementation of the digital signal processing blocks and D/A converter of the transmitter prototype.

The research of current-steering D/A converters was carried out during the years 2001-2005. The author, with help from Dr. Mikko Waltari, is responsible for the design of the calibration algorithm, and also designed and implemented the D/A converter core and analog parts related to calibration. The digital part of the calibration was implemented by Jussi Pirkkalaniemi, M.Sc. under the supervision of the author.

During the research, the new ideas and circuits presented in this thesis have been partially reported in related publications [1]-[15]. The results obtained are also applied in design presented in [16] and [17].

The thesis is organized as follows. Chapters 2-5 represent the background for the design of a digital-IF transmitter. In Chapter 2, the basic principles of the spread-spectrum quadrature amplitude modulation is presented and the performance metrics of the transmitter are given. In Chapter 3, the most common transmitter architectures

are briefly introduced. In Chapter 4, methods for resource-efficient digital filter design are introduced, and, in Chapter 5, the methods for direct digital frequency synthesis and digital modulation are discussed.

Chapter 6 contains the theory of the current-steering D/A converter design. Sections 6.1 and 6.2 give an introduction to current-steering converters and their performance metrics. In Section 6.3, the static linearity of the converter is analyzed. The previously published linearity yield models are compared, and the yield model developed and published by the author, Dr. Vankka and Ilari Teikari M.Sc. [13] is presented. Section 6.4 is about calibration techniques. Previously published calibration methods are discussed, and the method developed by the author, Mikko Waltari, and Jussi Pirkkalaniemi [14] is presented. Section 6.5 considers the distortion effects due to the output impedance variation. In Section 6.6, the signal conversion from discrete-time digital to continuous time analog is discussed, and distortion mechanisms are analyzed. In Section 6.7, timing-related nonlinearities are discussed. Results are also partially published in [15]. Section 6.8 considers the layout techniques used to reduce the effect of the process gradients on the current source mismatch. Section 6.9 is a survey of published D/A converters.

Chapter 7 describes the designed prototypes. Section 7.1 describes the design and experimental results of the WCDMA transmitter prototype.In this prototype, the author is responsible of hardware optimization and implementation of digital signal processing blocks, system simulations, and the design and implementation of the D/A converter. Dr. Waltari also gave valuable instructions for the D/A converter design. The design project was supervised by Dr. Jouko Vankka, who also developed algorithms related to filter design and digital upconverter. The measurements of the prototype were carried out by the author and Dr. Vankka. The results of this section are partially published in [1]-[12].

Section 7.2 describes the design and experimental results of the D/A converter prototype. The author is responsible of design and implementation of the D/A converter core including the comparator chain. Calibration algorithm is developed by the author, Jussi Pirkkalaniemi and Dr. Mikko Waltari. The digital parts of the calibration algorithm were designed and implemented by Jussi Pirkkalaniemi under supervision of the author. Measurements were carried out by the author and Jussi Pirkkalaniemi.

Finally, conclusions are drawn.

Chapter 2

Direct sequence spread-spectrum quadrature amplitude modulation

In CDMA systems, the data of different users are transmitted on the frequency band common to all users. The capacity of the frequency band is divided among the users by assigning a code channel to a single user by using a spreading code. A user in the system may use one or multiple code channels simultaneously. The spreading also improves system capacity by introducing the gain to the signal-to-noise ratio (*SNR*).

In the following sections, the fundamentals of direct sequence spread-spectrum QAM are presented in order to give some insight into the design presented in Chapter 7. More detailed information on the subject can be found in textbooks [18] and [19].

2.1 Principle of QAM

In the quadrature amplitude modulation scheme, two carriers, in phase and quadrature, are modulated with the data sequences $i(t)$ and $q(t)$ (Fig. 2.1)

$$\begin{aligned} s_t(t) &= i(t)\cos(\omega_c t) + q(t)\sin(\omega_c t) \\ &= \sqrt{i(t)^2 + q(t)^2}\cos(\omega_c t - \phi_d(t)), \end{aligned} \tag{2.1}$$

$$\phi_d(t) = \arctan\left(\frac{q(t)}{i(t)}\right). \tag{2.2}$$

In other words the information is shifted in frequency around the carrier frequency ω_c by multiplying with two orthogonal signals. While receiving, the signal $s_t(t)$ is

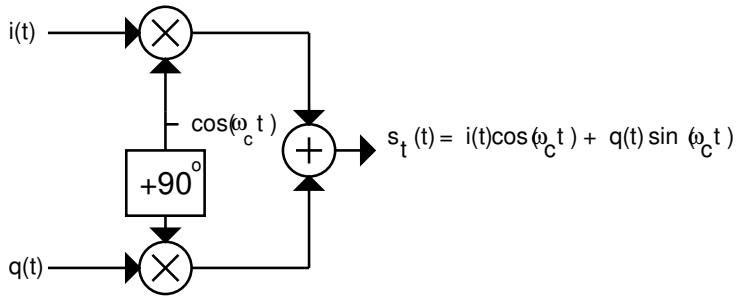

Figure 2.1 The principle of QAM.

downconverted around zero frequency.

$$\begin{aligned} s_{ri}(t) &= s_t(t)\cos(\omega_c t) \\ &= \frac{1}{2}i(t)(1+\cos(2\omega_c t)) + \frac{1}{2}q(t)\sin(2\omega_c t) \qquad (2.3) \\ s_{rq}(t) &= s_t(t)\sin(\omega_c t) \\ &= \frac{1}{2}i(t)\sin(2\omega_c t) + \frac{1}{2}q(t)(1-\cos(2\omega_c t)) . \qquad (2.4) \end{aligned}$$

After downconversion, the data signals $i(t)$ and $q(t)$ can be extracted by low-pass filtering.

The benefit of the QAM is that twice as much data as in bare in-phase modulation can be transmitted over the same frequency band due to the fact that both the amplitude and the phase of the carrier are modulated (see. Eq. (2.1)).

2.2 Spreading

Let's assume that the data signal $d(t)$ is an infinite random sequence of pulses with amplitude a_d and duration T_d (Fig. 2.2). This signal has an autocorrelation function

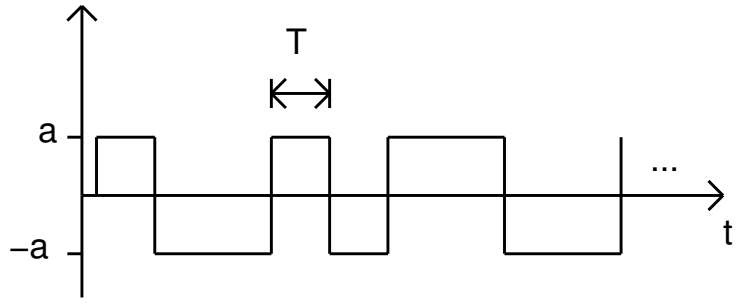

Figure 2.2 The data signal.

$$R_d(\tau) = \begin{cases} a_d^2\left(1-\frac{|\tau|}{T_d}\right) & ,|\tau| \le T_d \\ 0 & ,|\tau| > T_d \end{cases}, \tag{2.5}$$

where τ is the time misalignment between the two pulse sequences [18].

The power density function $D(f)$ of the data signal $d(t)$ is the Fourier transform of its autocorrelation function

$$\begin{aligned} D(f) &= \int_{-\infty}^{+\infty} R_d(\tau) e^{-j2\pi f\tau} d\tau \\ &= a_d^2 T_d \frac{\sin^2(\pi f T_d)}{(\pi f T_d)^2} \\ &= a_d^2 T_d \mathrm{sinc}^2(\pi f T_d) \\ &= \frac{a_d^2}{F_d}\mathrm{sinc}^2\left(\frac{\pi f}{F_d}\right), \end{aligned} \tag{2.6}$$

where F_d is the data rate. The main lobe, which contains most of the signal energy, has the width of $B_d = \frac{2}{T_d} = 2F_d$ centered at the zero frequency, so the upconverted data occupies the frequency band of $2F_d$.

Next we may define a sequence $c(t)$, which is a pulse sequence with magnitude value $a_c = 1$ and pulse duration T_c and pulse rate $F_c = \frac{1}{T_c}$. The autocorrelation function of $c(t)$ is

$$R_c(\tau) = \begin{cases} \left(1-\frac{|\tau|}{T_c}\right) & ,|\tau| \le T_c \\ 0 & ,|\tau| > T_c \end{cases}. \tag{2.7}$$

Next the data signal $d(t)$ is multiplied by $c(t)$.

$$s(t) = c(t)\,d(t) \tag{2.8}$$

Since $c(t)$ and $d(t)$ are independent of each other, the autocorrelation function of $s(t)$ is a product of the autocorrelation functions $R_c(t)$ and $R_d(t)$.

$$R_s(\tau) = R_c(\tau)R_d(\tau) = \begin{cases} a^2\left(1-\frac{|\tau|}{T_d}-\frac{|\tau|}{T_c}+\frac{\tau^2}{T_dT_c}\right) & ,|\tau| < T_c \\ 0 & ,|\tau| > T_c \end{cases}. \tag{2.9}$$

The power density function of $s(t)$ may now be calculated as [18].

$$\begin{aligned} S(f) &= \int_{-\infty}^{\infty} R_s(\tau) e^{-j2\pi f\tau} d\tau \\ &\simeq a^2\left(\frac{1}{T_d}+\frac{1}{T_c}\right)\frac{\sin^2(\pi T_c f)}{\pi^2 f^2}. \end{aligned} \tag{2.10}$$

When $T_d \gg T_c$, Eq. (2.10) can be approximated as

$$\begin{aligned} S(f) &\cong a_c^2 T_c \frac{\sin^2(\pi T_c f)}{T_c^2 \pi^2 f^2} \\ &= a_c^2 T_c \mathrm{sinc}^2(\pi T_c f) \\ &= \frac{\alpha^2}{F_c} \mathrm{sinc}^2\left(\frac{\pi f}{F_c}\right), \end{aligned} \tag{2.11}$$

which has a main lobe of width $B_s = \frac{2}{T_c} = 2F_c$ centered at zero frequency. This means that the energy of the signal $d(t)$ is spread to $G_s = \frac{2T_d}{2T_c} = \frac{F_c}{F_d}$ times wider frequency band. G_s is called spreading gain for the reason given in the next paragraph. The spreading pulses are also called chips, so T_c is also called the chip time and F_c the chip rate. After the spreading, data may be transmitted with, for example, an ordinary QAM structure. Such a structure is presented in Fig. 2.1.

While receiving, the sequence of spread data is multiplied with the same sequence that has been spread with

$$r(t) = d(t)\, c(t)\, c(t-\tau), \tag{2.12}$$

where τ is the timing misalignment. This multiplication despreads the data signal to its original frequency band while it spreads all possible jamming signals. Other data signals that have been spread with the codes uncorrelated with the spreading signals are not despread. The quality of despreading improves as the τ diminishes. In the ideal case $\tau = 0$, the power of the received signal is maximized relative to the noise. This is the main idea of the code division multiple access; users in the system can send their data on the same frequency band and the data sequences can be separated from each other by using uncorrelated spreading sequences for each user. This is presented in Figs. 2.3, 2.4 and 2.5.

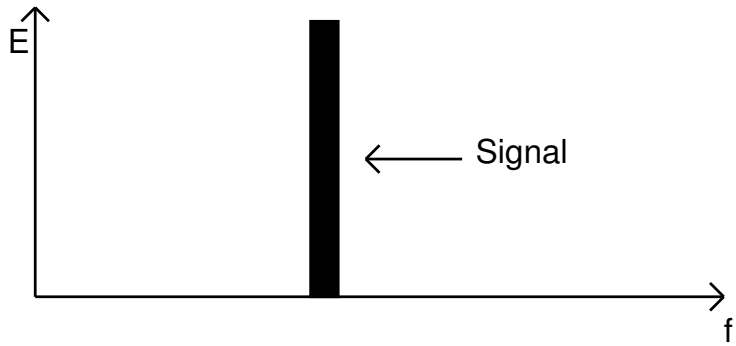

Figure 2.3 Signal before spreading.

The gain due to the despreading of the signal is G_s, which means, that after de-

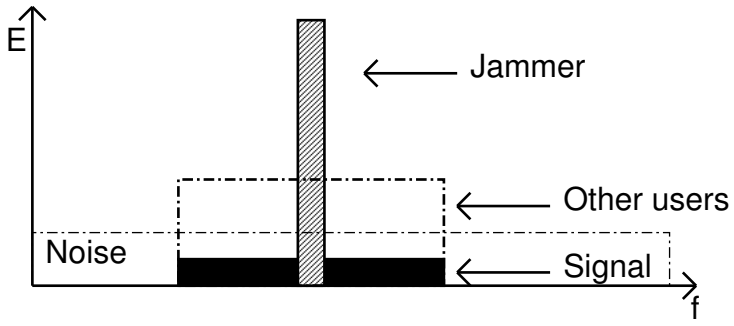

Figure 2.4 Spread signal in noisy environment with jammer signal and other users.

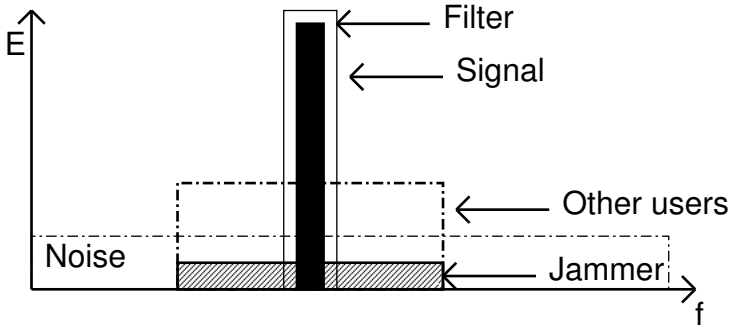

Figure 2.5 Signal in receiver after despreading.

spreading and filtering, the signal to noise ratio is increased with that factor compared to *SNR* at the receiver input. This means that the same bit error rate (which is the function of *SNR*) can be achieved with G_s times worse *SNR* at the receiver input than in the non-spreading systems.

2.3 Pulse shaping filtering

Usually it is necessary to limit the frequency band occupied by the signal in order not to disturb the signals transmitted on other frequency bands of the same or different systems. The filtering of the data sequences has to be performed in the time domain so that sequential pulses do not disturb each other. This is the reason why the channel bandwidth limiting filters are usually called pulse shaping filters when used in transmitters.

In a WCDMA system, the signal to be transmitted is data sequence spread with the spreading sequence. The spreading is performed with chip rate F_c and the filtering is

accomplished with a root-raised cosine filter [20], which has the impulse response

$$rrc(t) = \frac{\sin\left(\pi\frac{t}{T_c}(1-\alpha)\right) + 4\alpha\frac{t}{T_c}\cos\left(\pi\frac{t}{T_c}(1+\alpha)\right)}{\pi\frac{t}{T_c}\left(1-\left(4\alpha\frac{t}{T_c}\right)^2\right)}. \tag{2.13}$$

where T_c is the chip duration and α is the roll-off factor. A causal discrete time counterpart of Eq. (2.13) with N samples (N is odd) is

$$rrc(n) = \frac{\sin\left(\pi\frac{(1-\alpha)(n-N_c)}{L}\right) + \frac{4\alpha(n-N_c)}{L}\cos\left(\pi\frac{(1+\alpha)(n-N_c)}{L}\right)}{\pi\frac{(n-N_c)}{L}\left(1-\left(\frac{4\alpha(n-N_c)}{L}\right)^2\right)}, 0 \le n \le N-1. \tag{2.14}$$

where $N_c = \frac{N-1}{2}$ is the index of the center coefficient, $L = \frac{T_c}{T_s} = \frac{F_s}{F_c}$ is the oversampling ratio, T_s is the sampling interval and $F_s = \frac{1}{T_s}$ is the sampling frequency. Typically $L = \frac{T_c}{T_s}$is an integer.

When used for pulse shaping filtering in the transmitter and channel filtering at the receiver, the combination of these two filtering operations corresponds to a raised cosine (*RCOS*) filtering [21]. The ideal raised-cosine filter has two important properties. It has a bandlimited frequency response given by

$$RCOS(f) = \begin{cases} \frac{1}{F_c} & ,0 \le |f| \le (1-\alpha)\frac{F_c}{2} \\ \frac{1}{2F_c}\left(1-\sin\left(\pi\frac{1}{\alpha F_c}\left(|f|-\frac{F_c}{2}\right)\right)\right) & ,(1-\alpha)\frac{F_c}{2} \le |f| \le (1+\alpha)\frac{F_c}{2} \\ 0 & ,|f| > (1+\alpha)\frac{F_c}{2}, \end{cases} \tag{2.15}$$

and an impulse response

$$rcos(t) = \left(\frac{\sin\left(\pi\frac{t}{T_c}\right)}{\pi\frac{t}{T_c}}\right)\left(\frac{\sin\left(\alpha\pi\frac{t}{T_c}\right)}{1-\left(2\alpha\frac{t}{T_c}\right)^2}\right), \tag{2.16}$$

which has a causal discrete time equivalence of length N [21] given by

$$rcos(n) = \left(\frac{\sin\left(\frac{\pi(n-N_c)}{L}\right)}{\frac{\pi(n-N_c)}{L}}\right)\left(\frac{\cos\left(\frac{\pi\alpha(n-N_c)}{L}\right)}{1-\left(\frac{2\alpha(n-N_c)}{L}\right)^2}\right) \quad ,0 \le n \le N-1, \tag{2.17}$$

where $N_c = \frac{N-1}{2}$ is the index of the center coefficient (N is odd) and $L = \frac{T_c}{T_s}$ is the oversampling ratio. This impulse response has the property of having a zero value for any integer value of $n = \frac{N-1}{2} \pm kL, 0 < k < -\infty$. This means that there is no intersymbol interference (*ISI*) at the multiples of sampling interval T_s. The sequence of raised cosine impulses is presented in Fig. 2.6.

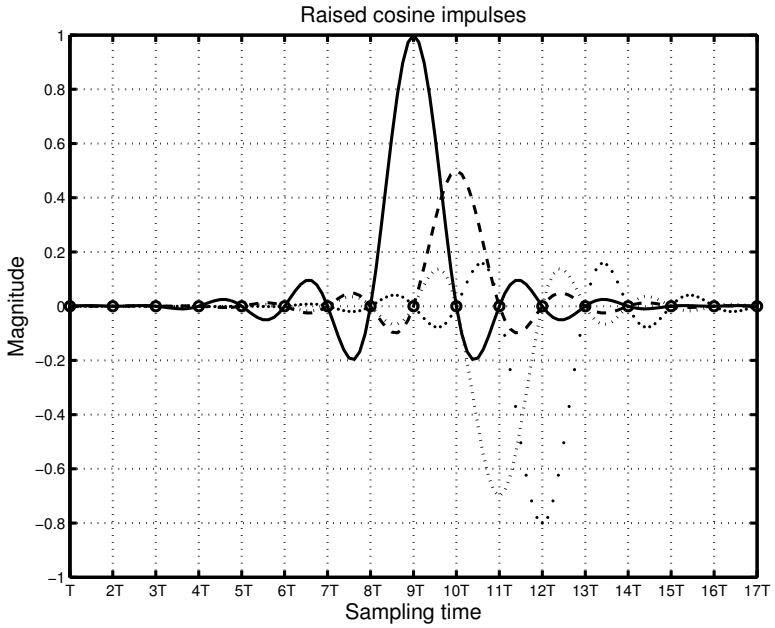

Figure 2.6 *ISI*-free impulses.

Because neither the transmit filter nor the receive filter are ideal root-raised cosine filters and neither have infinite length, the *ISI* value is not zero in practical cases. The *ISI* value is one of the performance metrics discussed in next section.

2.4 Performance metrics

Two fundamental metrics of digital communication system performance are bit error rate (*BER*) and symbol error rate (*SER*), which are the probabilities that an error occurs when receiving one transmitted bit or one transmitted symbol, respectively. Both of them are dependent of modulation type, signal-to-noise ratio (*SNR*), channel characteristics, detection type etc. [18]. However, when the base-station transmitter is designed, information on the other signal processing parts of the system is not necessarily available, so only the contribution of the subsystem under design to *SNR* and *BER* can be controlled.

In this book, *SNR* is defined via the error vector magnitude (*EVM*), which is defined to be the root mean square (RMS) deviation of the received symbol given as a percentage of the symbol magnitude [20]. Actually, this is an RMS noise amplitude given as a percentage of the symbol magnitude. *EVM* consists of all noise in the system, including *ISI* and quantization noise etc. When designing the transmitter, *EVM*

as a function of ISI and quantization noise has to be determined and minimized.

EVM_{rms} is defined as

$$EVM_{rms} = \frac{\sqrt{\sigma_s^2 + \sigma_n^2}}{|S|} = \sqrt{ISI_{rms}^2 + \frac{\sigma_n^2}{|S|^2}}, \tag{2.18}$$

where σ_s^2 is the variance of the symbol error generated by the nonidealities of the combination of the transmit and receive filters, σ_n^2 is the noise added by the system, and $|S|$ is the magnitude of the received symbol.

ISI_{rms} is defined as

$$ISI_{rms} = \frac{\sigma_s}{|S|} = \frac{\sqrt{\left(\sum_{i=-K, i\neq 0}^{K} h_{tr}\left(N_c + iL\right)^2\right)}}{h_{tr}\left(N_c\right)}, \quad K = floor\left(\frac{N_c - 1}{L}\right), \tag{2.19}$$

where $h_{tr}(n)$ is the combination of the channel filters of transmitter and the receiver. L is the oversampling ratio and $N_c = \frac{N-1}{2}$ is the index of the center coefficient of the filter $h_{tr}(n)$. N is the number of the filter coefficients (odd).

In addition to ISI_{rms}, the contribution of the current frequency channel to EVM of the adjacent frequency channels has to be minimized. This means minimization of the contribution to the additive noise σ_n^2 of the adjacent channel. The amount of this contribution is given as the adjacent channel power ratio ($ACPR_n$) in decibels

$$ACPR_n = 10\log_{10}\left(\frac{P_{nth\_adj\_chan}}{P_{cchan}}\right), \tag{2.20}$$

or as the adjacent channel leakage power ratio, which is the inverse of $ACPR_n$

$$ACLR_n = 10\log_{10}\left(\frac{P_{cchan}}{P_{nth\_adj\_chan}}\right), \tag{2.21}$$

where $P_{nth\_adj\_chan}$ is the power of nthadjacent frequency channel and P_{cchan} is the power of the current signal channel.

$ACLR_n$ and EVM_{rms} are considered to be the key performance metrics of the base-station transmitter throughout the design process described in this book.

Chapter 3

Transmitter structures

The most commonly used (and published) transmitter architectures are direct conversion [22], [23], [24], [25], [26], [27] and two-step transmitter [22], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37]. A two step transmitter can be realized either with an analog or digital intermediate frequency (IF) part.

Direct conversion and two-step transmitter structures are usually used in the amplitude modulators, whereas the frequency synthesizer based transmitters [38], [39], [40], [41], [42], [43] are used in narrow band phase and frequency modulators.

Two transmitter types used with the power amplifier (PA) linearization techniques [44], [45] are also described, because the linearization techniques require that the signals are presented in a different format at the IF frequency compared to the feed-forward or look-up table methods[46], [47] in which the signal at the IF is still basically the sinusoidal carrier multiplied by the filtered data.

3.1 Direct conversion transmitter

The principle of the direct conversion transmitter is presented in Fig. 3.1. In direct

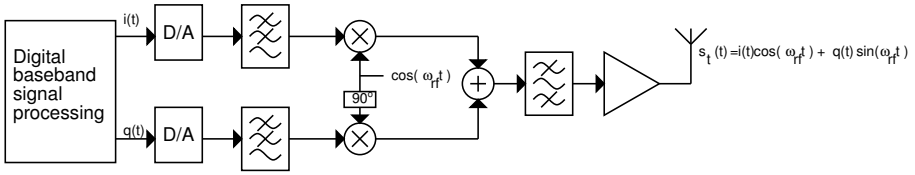

Figure 3.1 Direct conversion transmitter.

conversion transmitters, the bandlimited baseband signals are converted directly up to the radio frequency ω_{rf} with in-phase and quadrature carriers. The band-pass filter

after the signal summation is used to suppress the out-of-band signals generated by the harmonic distortion of the carrier. The structure of this kind of transmitter is quite simple; however, it suffers from the following drawback. The strong signal at the output of the power amplifier may couple to the local oscillator (LO), which is usually a voltage controlled oscillator, causing the phenomenon that is known as injection pulling [22], [48]. This means that the frequency of the local oscillator is pulled away from the desired value. The severity of the injection pulling is proportional to the difference between the frequency of the local oscillator and the frequencies at the output of the PA. By taking advantage of that, the problem of injection pulling can be alleviated by using an offset LO direct-conversion structure (Fig. 3.2). In this structure the carrier

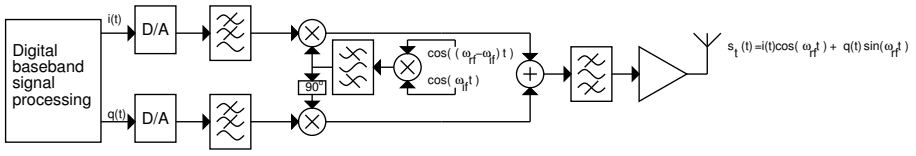

Figure 3.2 Offset LO direct-conversion transmitter.

signal is formed by mixing two lower frequency signals. An additional band-pass filter is needed to filter away the undesired carrier at the frequency $\omega_{rf} - 2\omega_{if}$.

3.2 Two-step transmitter

The injection pulling can also be avoided by using the two-step transmitter presented in Fig. 3.3. The two- (or multiple-) step transmitter can be considered as a dual of the super-heterodyne receiver [22]. In this structure, the baseband data is first upconverted

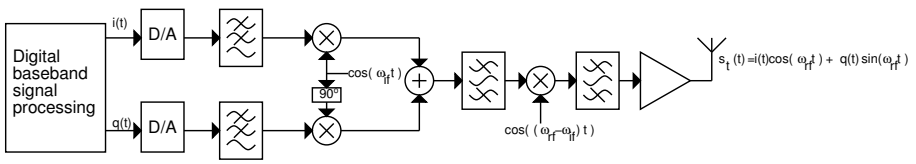

Figure 3.3 Two-step transmitter.

to the intermediate frequency ω_{if} and then to the desired radio frequency ω_{rf}. The two-step transmitter has two advantages. First, the quadrature modulation is performed at a fixed lower frequency resulting in better matching between in-phase and quadrature (I and Q) carriers, which in turn diminishes the crosstalk between I and Q data streams. Second, the additional attenuation of the adjacent channel spurs and noise may be achieved by using a band-pass filter at the IF. The drawback is that the stop band

attenuation at the RF frequency has to be larger than in a direct conversion transmitter because the signal component at the frequency $\omega_{rf} - 2\omega_{if}$ has the same power as the desired sideband.

3.3 Phase modulating synthesizers

When the narrow band phase or frequency modulation is performed, the phase modulating synthesizer is often used [38], [39], [40], [41]. The synthesizer can be either an analog one, such as a phase locked loop (PLL) or a direct digital synthesizer (DDS), often also called a numerically controlled oscillator (NCO) . The PLL-based modulator is presented in Fig. 3.4. This kind of transmitter generates the modulated carrier

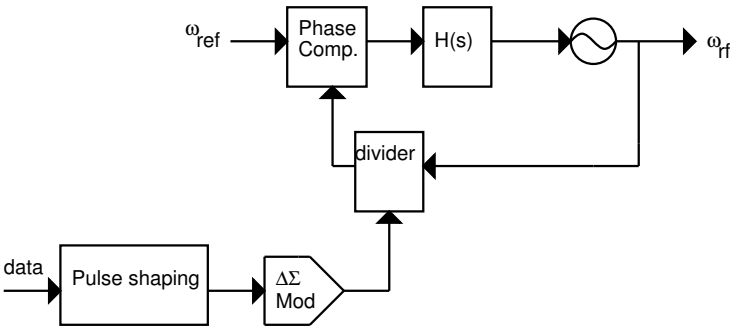

Figure 3.4 PLL based modulator.

by controlling the frequency divider of the PLL with a pulse shaped ΔΣ-modulated data stream, thus modulating either phase or frequency. The phase comparator detects the phase difference between the reference oscillator and the output of the divider and tunes the voltage-controlled oscillator (VCO) to minimize the average phase difference, thus producing a phase modulated carrier at radio frequency (RF). The usable bandwidth of the data is limited by the bandwidth of the loop filter $H(s)$, preventing the usage of this kind of transmitter in wide-band systems. This kind of transmitter is used in systems such as GSM [39], [40] or Bluetooth [49].

Instead of using an analog frequency synthesizer, the phase modulator can also be realized with a digital frequency synthesizer [50], [43], [42]. The basic principle of the digital phase modulator is presented in Fig. 3.5. With this kind of structure, both frequency and phase synthesizers can be realized (the data could also be added to the ”frequency control” shown in Fig. 3.5). They are often used in the same kind of applications as their analog counterparts, but they do not suffer from loop filter bandwidth limitations like the PLL based synthesizers do. Because of this, it has become increasingly interesting to implement multi-mode base-station transmitters, where the

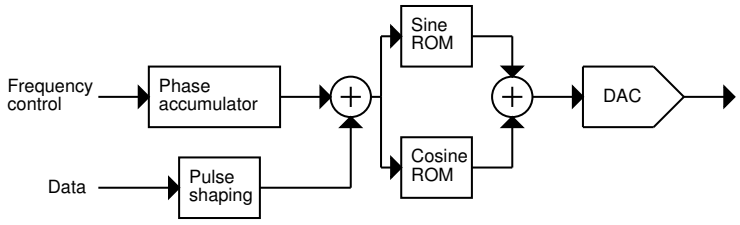

Figure 3.5 Digital phase modulation.

phase or frequency modulation is added to the digital frequency synthesizer used for the digital QAM modulator enabling the same transmitter to be used in multiple communications standards such as GSM, EDGE, and WCDMA. [17].

3.4 Constant envelope transmitters for power amplifier linearization

The PA linearization techniques described in this section require special signal decomposition at the baseband or at the IF frequencies. They can therefore be considered as separate transmitter architectures, whereas techniques such as feed-forward- [46] and look-up-table-based methods [47] are basically conventional QAM transmitters with linearizing predistortion.

With constant envelope signals, it is possible to use nonlinear power-efficient (class C, D, E or F) amplifiers because the constant envelope signals produce less distortion in the signal frequency band than the signals with a varying envelope [22]. Constant envelope signals are, for example, phase or frequency modulated signals, which have, however, a poor spectral efficiency. In order to increase the spectral efficiency, modulation techniques in which both the amplitude and phase are varying (such as the multilevel QAM) have to be used. Difficulties arise when the average transmitted power is much lower than the maximum peak power.

The ratio of the peak power to average power is called the crest factor. A high crest factor causes the following problems. First, it is not possible to use a nonlinear (power efficient) PA, because the distortion destroys the signal integrity and detection of the information that is coded to the amplitude becomes more difficult or impossible. Also, the distortion of the non-constant envelope signal causes spectral re-growth (widens the signal spectrum) and therefore disturbs the adjacent signal bands [51]. Even if the distortion problem is solved by using a more linear (class A or AB) amplifier, the efficiency is even further reduced because the amplifier has to be biased according to the maximum signal amplitude.

Both of the methods described in this section are based on the decomposition of the IF signal. The IF signal is decomposed in the constant envelope components, which allows the usage of a nonlinear power amplifier.

3.4.1 Envelope elimination and restoration

The envelope elimination and restoration (EER) [44], [52] transmitter is presented in Fig. 3.6. The relationship between the general amplitude modulation and the decom-

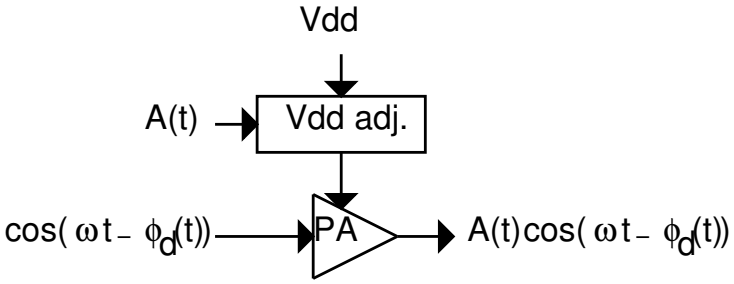

Figure 3.6 Envelope elimination and restoration.

posed presentation of Fig. 3.6 is

$$\begin{aligned} S_t(t) &= i(t)\cos(\omega t) + q(t)\sin(\omega t) \\ &= A(t)\cos(\omega t - \phi_d(t)), \end{aligned} \tag{3.1}$$

where

$$\begin{aligned} A(t) &= \sqrt{i(t)^2 + q(t)^2} \\ \phi_d(t) &= \arctan\left(\frac{q(t)}{i(t)}\right). \end{aligned} \tag{3.2}$$

The $\cos(\omega t - \phi_d)$ term in Eq. (3.1) has a constant envelope and can be amplified with a nonlinear PA. The amplitude information $A(t)$ is added to the signal by controlling the supply voltage of the amplifier.

The main weakness of this method is that it requires good matching between the amplitude branch and the carrier branch.

3.4.2 LINC transmitter

LINC is an abbreviation meaning linear amplification with nonlinear components [45], [53] (a.k.a. as an outphasing method [54]). The LINC transmitter is presented in Fig. 3.7.

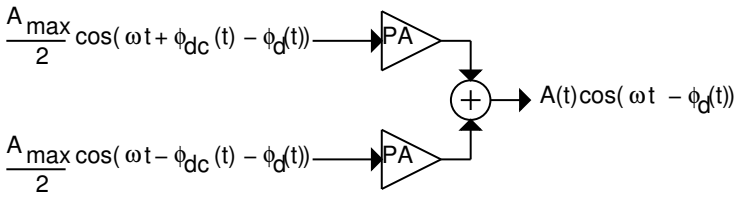

Figure 3.7 LINC power amplification.

The input signals of the power amplifiers are formed by further expanding Eq. (3.1), resulting in

$$\begin{aligned} s(t) &= A(t)\cos(\omega t - \phi_d(t)) \\ &= A_{max}\cos(\phi_{dc})\cos(\omega t - \phi_d(t)) \\ &= \frac{A_{max}}{2}\cos(\omega t + \phi_{dc} - \phi_d(t)) + \frac{A_{max}}{2}\cos(\omega t - \phi_{dc} - \phi_d(t)), \end{aligned} \tag{3.3}$$

in which

$$\begin{aligned} A(t) &= \sqrt{i(t)^2 + q(t)^2} \\ A_{max} &= max(abs(A(t))) \\ \phi_{dc}(t) &= \arccos\left(\frac{A(t)}{A_{max}}\right) \\ &= \arccos\left(\frac{\sqrt{i(t)^2 + q(t)^2}}{A_{max}}\right). \end{aligned} \tag{3.4}$$

The main weakness in the LINC realizations, as in the EER method, is the matching between the signal paths in the analog domain. If the matching is not perfect, the quality of the signal is degraded. Also, the bandwidth of the signal components is increased when compared to the original signal, and therefore a high sampling rate is usually required for the DSP. In addition, the lossless summation of high-power signals is very difficult, resulting in efficiency degradation.

3.5 Digital QAM transmitter

The digital QAM transmitter is presented in Fig. 3.8. The digital signal processing part of the transmitter usually consists of digital pulse shaping filters and a digital frequency synthesizer, which are discussed in Chapters 4 and 5. In order to make digital modulation possible, the sampling rate conversion (interpolation) between the data input sampling rate and the sampling rate of the frequency synthesizers is needed.

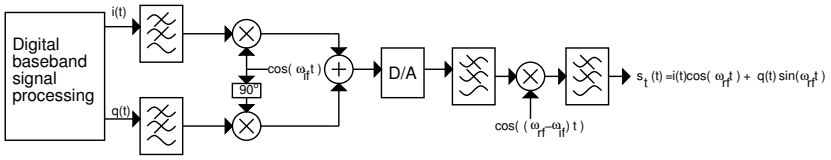

Figure 3.8 Digital QAM transmitter.

This also requires filtering, which is discussed in more detail in Chapter 4.

The main benefit of the digital QAM transmitter is that the modulation does not suffer from any kind of physical accuracy limitations or distortion that may occur when an analog multiplication is used to produce the QAM signal. Another benefit is that there is only one signal branch in the analog domain, eliminating the gain and phase imbalance problems between I and Q branches. When used in multi-carrier transmitters, the digital QAM has the benefit of lossless signal addition and mismatch-free performance. The digital QAM also has all the benefits that DSP provides. For example, the frequency resolution of the digital frequency synthesizer can be selected arbitrarily. This allows the frequencies of the carriers to be freely selectable within the frequency band allocated to the system. It is also possible to implement several types of signal-enhancement techniques with digital signal processing, including, but not limited to, amplitude and phase-distortion compensation and power-amplifier linearization with predistortion techniques. The main limitations of the usage of the digital QAM are the high power dissipation, which makes it unusable in portable devices, and the accuracy of the D/A converter. Digital transmitters are mainly used in systems like cable modems and base-station transmitters, in which the large power dissipation is not a problem [37], [36], [35].

Chapter 4

Resource-efficient digital filter design

Resource-efficient DSP generally becomes a topic when the system size integrated on a single chip increases. The main resources that we are dealing with are power, speed and area. It can be shown that each of these can be traded off against each other, hence it is only a question of which one of these properties is the most important one in a specific design case. For the mobile applications, it is almost always power, then area. For the non-portable application, the power optimization is less important, although not meaningless. Less power means the possibility of integrating more on a single chip without melting-up the package or without needing to install a cooling fan.

Speed of computation is often considered a figure of merit when speaking about DSP chips. The fact is that speed can be increased by using such techniques as pipelining and parallelism, that is, by increasing the chip area. The same techniques may also be applied in order to achieve low-power performance. However, increasing the area means increasing the price of the chip.

In this chapter, some techniques for a resource-efficient filter design are discussed. It should be kept in mind that, in the design described in Section 7.1, the order of priorization of the resources is first area, then power. The computation speed is fixed by the system specification, so the special methods for high-speed designs are not considered. The top-down method is used to describe the techniques of the area-efficient filter design.

4.1 Filter design algorithms

4.1.1 Pulse shaping filter design algorithm

The pulse shaping filter design has two main objectives: minimization of the inter symbol interference (*ISI*) and maximization of the adjacent channel leakage power ratio (*ACLR*). Most of the algorithms used for the finite impulse response (FIR) filter design, such as the methods introduced in [55], [56], [57] and [58], are suitable for only the stop-band attenuation maximization (or equiripple filter design) and does not take *ISI* into account.

One way to design filters that have at least some kind of *ISI* properties is to use the sampled impulse response of the root-raised cosine filter (Eq. (2.17)) as filter coefficients. However, the performance is far from ideal, and the stop-band attenuation is usually poor with a small number of coefficients. This problem may be alleviated by using some window function, such as Kaiser, but this worsens *ISI* performance. Windowing may also widen the pass band, which is not desirable. With the windowing method, the stop-band attenuation and *ISI* may be traded off against each other, but usually the number of coefficients for the practical values of *ACLR* and *ISI* becomes quite high. A better method for finding the pulse shaping filter coefficients was described in [6], which is based on Lagrange optimization. The Lagrange optimization method is also used in [59] to design filters or multi-rate systems.

The method goes as follows. An ideal root-raised receive filter is approximated with $h_r(n)$ with length I (odd). We try to design a transmit filter $h_t(n)$ of length K in such a way that ISI_{rms} is minimized and *ACLR* is maximized. Both filters have the same oversampling ratio L. The combination of these two filters is the time domain convolution of their impulse responses

$$\begin{aligned} h_{tr}(n) &= \sum_{i=0}^{K-1} h_t(i)h_r(n-i) \quad n = 0 \ldots N-1, N = K+I-1 \\ &= h_t^T S_n h_r, \end{aligned} \tag{4.1}$$

in which S_n is a $I \times K$ convolution matrix with elements

$$\begin{aligned} sn_{a,b} &= 1, b = n+2-a, \quad 1 \leq b \leq K \quad 1 \leq a \leq I, \\ sn_{a,b} &= 0 \qquad \text{otherwise.} \end{aligned} \tag{4.2}$$

Next ISI_{rms} has to be defined as a function of $h_t(n)$. ISI_{rms} can be presented in

matrix form

$$ISI_{rms}^2 = \frac{\sum_{i=-M,i\neq 0}^{M} h_t^T \left(S_{N_c+iL} h_r\right)\left(S_{N_c+iL} h_r\right)^T h_t}{h_t^T \left(S_{N_c} h_r\right)\left(S_{N_c} h_r\right)^T h_r}, \quad M = floor(\frac{N_c-1}{L}), N_c = \frac{N-1}{2}$$
$$= \frac{h_t^T A A^T h_t}{h_t^T A_c A_c^T h_t}, \tag{4.3}$$

where

$$A = \sum_{i=-M,i\neq 0}^{M} S_{N_c+iL} h_r \tag{4.4}$$

and

$$A_c = S_{N_c} h_r, \tag{4.5}$$

in which $N_c = \frac{N-1}{2}$ (N odd) is the index of the center coefficient of the combination of transmit and receive filters. Now, we have a matrix presentation for the normalized ISI.

Next the matrix equations for the pass-band and stop-band power of the transmit filter $h_t(n)$ are derived. The amplitude frequency response of the filter may be written as

$$H_t(f) = \sum_{n=0}^{K-1} h_t(n) e^{-j\frac{2\pi fn}{F_s}}$$
$$= \sum_{n=0}^{K-1} h_t(n) \left(\cos\left(\frac{2\pi fn}{F_s}\right) - j\sin\left(\frac{2\pi fn}{F_s}\right)\right), \tag{4.6}$$

where F_s is the sampling frequency. The power transfer function may then be written as

$$P(f) = |H_t(f)|^2$$
$$= \sum_{i=0}^{K-1}\sum_{k=0}^{K-1} h_t(i) h_t(k) e^{-j\frac{2\pi fi}{F_s}} e^{j\frac{2\pi fk}{F_s}} \tag{4.7}$$
$$= \sum_{i=0}^{K-1}\sum_{j=0}^{K-1} h_t(i) h_t(j) \cos\left(\frac{2\pi f(i-k)}{F_s}\right) - j\sum_{i=0}^{K-1}\sum_{k=0}^{K-1} h_t(i) h_t(k) \sin\left(\frac{2\pi f(i-k)}{F_s}\right).$$

Because $\sin(-x) = -\sin(x)$, the term

$$j\sum_{i=0}^{K-1}\sum_{k=0}^{K-1} h_t(i) h_t(k) sin\left(\frac{2\pi f(i-k)}{F_s}\right) \tag{4.8}$$

in Eq. (4.8) equals zero, and the power transfer function of the transmit filter $h_t(n)$

may be written as

$$\begin{aligned} P(f) &= |H_t(f)|^2 \\ &= \sum_{i=0}^{K-1}\sum_{k=0}^{K-1} h_t(i)\,h_t(k)\cos\left(\frac{2\pi f(i-k)}{F_s}\right). \end{aligned} \tag{4.9}$$

Now we may discover the integrated power on some pass-band from $-F_{pb}$ to F_{pb}.

$$\begin{aligned} E_p &= \int_{-F_{pb}}^{F_{pb}} P(f)\mathrm{d}f \\ &= \sum_{i=0}^{K-1}\sum_{k=0}^{K-1} h_t(i)h_t(k)\frac{F_s}{\pi(i-k)}\sin\left(\frac{2\pi F_{pb}(i-k)}{F_s}\right), \end{aligned} \tag{4.10}$$

and for stop bands from $\frac{-F_s}{2}$ to F_{sb} and F_{sb} to $\frac{F_s}{2}$

$$\begin{aligned} E_s &= \int_{-\frac{F_s}{2}}^{-F_{sb}} P(f)df + \int_{F_{sb}}^{\frac{F_s}{2}} P(f)\mathrm{d}f \\ &= \sum_{i=0}^{K-1}\sum_{k=0}^{K-1} h_t(i)h_t(k)\frac{F_s}{\pi(i-k)}\left(\sin(\pi(i-k)) - \sin\left(\frac{2\pi F_{sb}(i-k)}{F_s}\right)\right). \end{aligned} \tag{4.11}$$

We may write the passband power in the matrix form

$$E_p = h_t^T P_p h_t \tag{4.12}$$

in which P_p is a $K \times K$ square matrix with elements

$$pp_{i,k} = \begin{cases} 2F_{pb}, & \text{when} \quad i-k=0 \\ \frac{F_s}{\pi(i-k)}\sin\left(\frac{2\pi F_{pb}(i-k)}{F_s}\right) & \text{otherwise.} \end{cases} \tag{4.13}$$

Similarly, for the stop-band power we get

$$E_s = h_t^T P_s h_t, \tag{4.14}$$

in which P_s is a $K \times K$ matrix with elements

$$ps_{i,k} = \begin{cases} F_s - 2F_{sb}, & \text{when} \quad i-k=0 \\ -\frac{F_s}{\pi(i-k)}\sin\left(\frac{2\pi F_{sb}(i-k)}{F_s}\right) & \text{otherwise.} \end{cases} \tag{4.15}$$

Now, when we have the matrix equations for ISI_{rms}^2, E_p and E_s, we may write the

Lagrangian cost function to be maximized

$$L(h_t,\lambda) = h_t^T P_p h_t - a h_t^T P_s h_t - b h_t^T A A^T h_t + \lambda \left(h_t^T A_c - 1 \right), \tag{4.16}$$

in which the a and b are the weight factors for the stop band-power and the ISI_{rms}^2. λ rule sets the value of the center coefficient of $h_{tr}(n)$ to be one. By taking the partial derivatives and setting them to zero we get

$$\frac{\partial L(h_t,\lambda)}{\partial h_t} = 2P_p h_t - 2aP_s h_t - 2bAA^T h_t + \lambda A_c = 0 \tag{4.17}$$

$$\frac{\partial L(h_t,\lambda)}{\partial \lambda} = h_t^T A_c - 1 = 0. \tag{4.18}$$

By setting

$$Q = 2P_p - 2aP_s - 2bAA^T \tag{4.19}$$

and solving Eqs. (4.17) and (4.18) we obtain

$$h_t = \frac{Q^{-1}A_c}{(Q^{-1}A_c)^T A_c}, \tag{4.20}$$

which is the Lagrange optimized transmitter filter.

It should be noted that the filter that was used for the receiver may also include the combination of all filters in the transmitter after the pulse shaping filter. With this method, the effect of the filters after the pulse shaping filter in the transmitter may be compensated.

The main shortcoming of this algorithm is that the effect of the weighting factors a and b has to discovered by trial and error. Results of different filter design methods are compared in Table 4.1. The number of the filter coefficients is 37 for each filter.

Table 4.1 Comparison of filter design methods.

Method	*ACLR*	*ISI*
Truncation	45.30dB	-59.21dB
Window with Kaiser, $\beta = 4$	36.15dB	-40.07dB
Lagrange	73.38dB	-45.08dB
Root-raised cosine with 1001 coefficients	71.22dB	-106.10dB

In simulations, the oversampling ratio is 2 and the sample frequency is normalized to that. The pass band is defined to be from 0 to 0.61Hz and the stop band (adjacent channel) from 0.61Hz to 1Hz. It can be seen that the *ACLR* value of the filter designed with the window method suffers from the increased width of the pass band. The frequency responses of the filters are presented in Fig. 4.1.

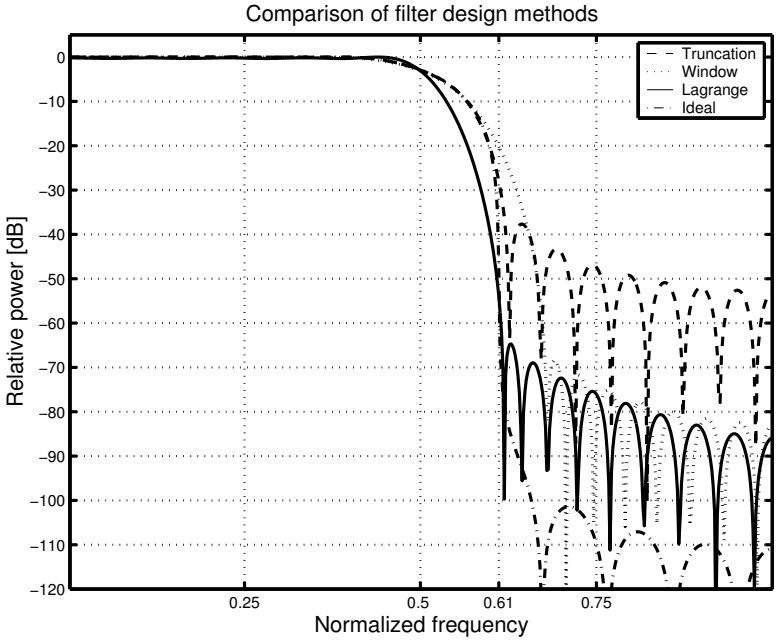

Figure 4.1 Frequency responses of the filters designed with different design methods.

4.1.2 Half-band filters for interpolation

When sampling rate conversion takes place, digital filters are needed. Filters used in sampling rate conversion must fulfill two criteria. The first and most important is that they have to filter out the image band in the case of interpolation, or the aliasing band in the case of decimation. The second is that they should be as simple as possible. When we have to interpolate or decimate signals with a reasonably wide bandwidth compared to sampling frequency, the usage of comb filters [60], [61] is not preferable because they introduce droop on their pass band. The second best approach compared to comb filters is to interpolate in steps of two if possible. This is because the number FIR filter coefficients needed for the filtering has an approximate dependency [58]

$$N \cong K\frac{F_s}{\Delta F}, \tag{4.21}$$

where K is a factor depending on the stop-band and pass-band ripple characteristics of the filter, F_s is the sample frequency and ΔF is the width of the transition band. In addition, when interpolating by two, half-band filters may be used. Half-band filters have the center of their transition band at the quarter of the sampling frequency. They can be designed with any of the filter design algorithms described in, for example, [58]. However a shortcut for their design has been introduced in [62]. Half-band filters

have the property that every other of their coefficients except the center coefficient (odd N) has zero value. This is expandable to s.c. Φ-band filters [63], in which every L^{th} coefficient is zero except the center one [62]. Φ-band filters are filters that have $L-1$ don't care bands. The half-band filters that are used in the transmitter of Chapter 7 are designed with the Least Squares error minimization algorithm [58] and by using the "trick" described in [62].

4.2 Mapping the floating point filter coefficients to canonic signed digit format

In the previous section, the method for efficient filter design for floating point presentation of the filter coefficients was presented. Because the floating point computation is quite tricky to perform on the silicon, it is preferable to use fixed point presentations of the filter coefficients. Moving from floating point to the fixed point presentation degrades the accuracy of the presentation and introduces error into the filtering operation.

Multiplication of two fixed point number consumes a lot more power and area on silicon when compared to summation. In the case when the filter has constant coefficients it is preferable to use a signed digit (SD) presentation for the filter coefficients [64]. In SD presentation, the filter coefficient is presented as sums and differences of powers of two

$$h(n) = \sum_{i=-\infty}^{\infty} c_{ni} 2^i, c_{ni} \in \{-1,0,1\}. \tag{4.22}$$

In digital filters, it is convenient to normalize the maximum value of the filter coefficient so that the maximum power of two is zero. It is also possible to present the filter coefficient only with some limited accuracy by fixing the minimum value of i. This leads to the approximation of $h(n)$

$$h_{sd}(n) = \sum_{i=-p}^{0} c_{ni} 2^i, c_{ni} \in \{-1,0,1\}. \tag{4.23}$$

When the number is presented with a minimum number of non-zero digits, the presentation is said to be a minimum signed digit (MSD) presentation. There can be multiple MSD representations for a single number, but there is only one MSD presentation in which there is no non-zero digits in parallel. This representation is called the canonic signed digit presentation (CSD). For example 0.75, in decimal notation can be presented as "1 0 -1" or "0 1 1" in signed digit presentation, of which "1 0 -1" is CSD presentation. The benefit of the SD representations is that the multiplication operation can be realized by using only adders/subtracters and shift operations that can be realized with hardwired shifts.

Several algorithms have been presented for mapping the floating point or regular two's complement presentation to the CSD presentation [65], [66], [67], [68], [69]. In the design described in Chapter 7, the modification of the method presented in [67] was used. The method was modified in order to trade off *ACLR* and *ISI* rather than the peak amplitude ripple and *ISI*. This is because the parameter to be optimized is the adjacent channel power relative to the channel power, not the ripple of the power transfer function. The equiripple design algorithms usually give poorer power attenuation on the stop band than the filters that have been designed in the sense of the least squares stop-band. Almost same kind of method was used in [70].

4.3 Efficient FIR filter structures

Once the filters have been converted to the SD representation, the further area and power reductions may be achieved by rearranging the computation on the silicon chip. In this chapter, a couple of well -known methods are described.

4.3.1 Polyphase FIR filters in sampling rate converters

The basic operations in sampling rate conversion are converting the sampling rate upwards (i.e. interpolation) and downwards (i.e. decimation). The interpolation consists of upsampling followed by filtering, while the decimation consists of filtering followed by downsampling. The decimation and interpolation and their efficient realizations are described in the following paragraphs.

The upsampling operation for the data sequence $x(n)$ may be described with

$$x_i(n) = \begin{cases} x\left(\frac{n}{L}\right), & \frac{n}{L} \in \mathbb{Z} \\ 0 \quad \text{otherwise}, \end{cases} \tag{4.24}$$

which means that $L-1$ zeros are inserted between the samples. Equation (4.24) can also be presented with discrete Fourier series as

$$x_i(n) = \frac{1}{L} \sum_{l=0}^{L-1} x\left(\frac{n}{L}\right) e^{j\frac{2\pi nl}{L}}, \tag{4.25}$$

which is a suitable format when, for example, analyzing upsampling in the frequency domain.

Downsampling may be described with

$$x_d(n) = x(nM), \tag{4.26}$$

which means that only every M^{th} sample of $x(n)$ is included in $x_d(n)$.

The effects of the upsampling in the frequency domain can be discovered by calculating the Z-transform of $x_i(n)$ and by evaluating it on the unit circle (i.e. by substituting $Z = e^{j\frac{2\pi f}{F_s}}$), resulting in

$$\begin{aligned} X_i(f) &= \frac{1}{L}\sum_{l=0}^{L-1}\sum_{n=0}^{L(N-1)} x\left(\frac{n}{L}\right) e^{j\frac{2\pi nl}{L}} e^{-j\frac{2\pi fn}{LF_s}} \\ &= \frac{1}{L}\sum_{l=0}^{L-1}\sum_{b=0}^{N-1} x(b)\, e^{-j\frac{2\pi b}{\frac{F_{si}}{L}}\left(f-l\frac{F_{si}}{L}\right)} \\ &= \frac{1}{L}\sum_{l=0}^{L-1} X(f-l\frac{F_{si}}{L}) \\ &= \frac{1}{L}\sum_{l=0}^{L-1} X(f-lF_s) = X(f) \end{aligned} \tag{4.27}$$

where F_s is the sampling frequency before interpolation, $\frac{n}{L} = b$ and $LF_s = F_{si}$. This means that the spectrum is the same, although the sampling frequency has been changed, meaning that there are unwanted images between the new sampling frequency and the original signal band. The effects in the frequency domain are presented in Fig. 4.2.

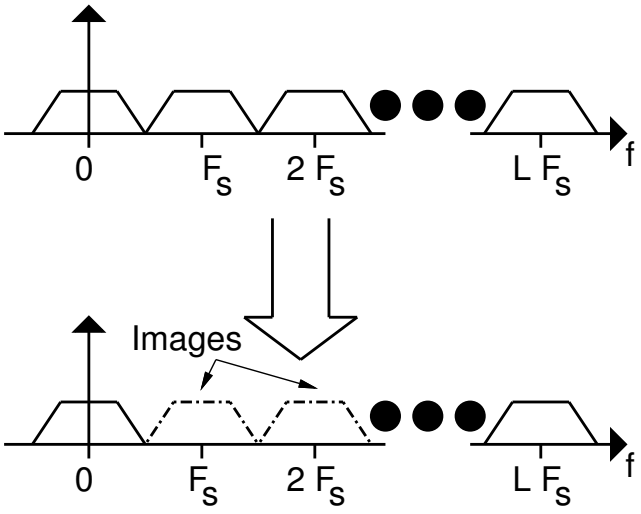

Figure 4.2 Effects of interpolation in the frequency domain.

The images should be filtered out after upsampling because they contain redundant information, which may, for example contaminate signals on the adjacent frequency bands.

For downsampling, $x_d(n)$ in the frequency domain may be written as

$$\begin{aligned} X_d(f) &= \sum_{n=0}^{\frac{N-1}{M}} x(nM) e^{-j\frac{2\pi n f}{F_{sb}}} \\ &= \sum_{b=0}^{N-1} \left(\frac{1}{M} \sum_{l=0}^{M-1} x(b) e^{j\frac{2\pi b l}{M}} \right) e^{-j\frac{2\pi b f}{F_s}} \\ &= \frac{1}{M} \sum_{l=0}^{M-1} \sum_{b=0}^{N-1} x(b) e^{-j\frac{2\pi b}{F_s}\left(f - l\frac{F_s}{M}\right)} \\ &= \frac{1}{M} \sum_{l=0}^{M-1} X(f - lF_{sd}), \end{aligned} \tag{4.28}$$

in which $b = nM$ and $F_{sd} = \frac{F_s}{M}$. This means that the spectrum after decimation contains aliased components from the frequency bands centered $l\frac{F_s}{M}$, $l = 1 \ldots M-1$. In order to avoid aliasing (Fig. 4.3), the signal should be bandlimited before decimation (i.e. the signal power of the aliasing bands should be zero).

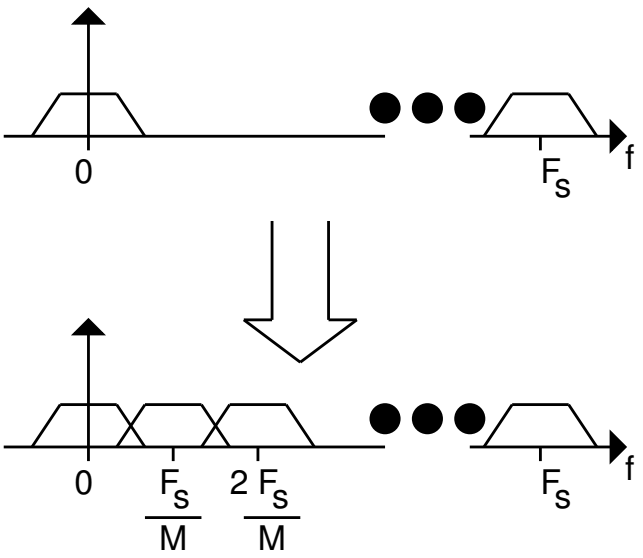

Figure 4.3 Effects of decimation in the frequency domain.

Interpolation and decimation filters can be realized efficiently by using so-called polyphase decomposition. Polyphase decomposition is based on identities presented in Fig. 4.4 and holds for every L and M [58]. The polyphase decomposition of the interpolation filter and one possible realization of it are presented in Figs. 4.5 and 4.6. The decomposed filter for decimation and one possible implementation of it are presented in Figs. 4.7 and 4.8.

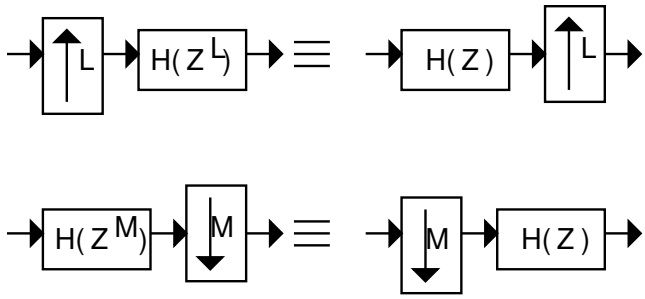

Figure 4.4 Identities for the order of filtering and up/down sampling.

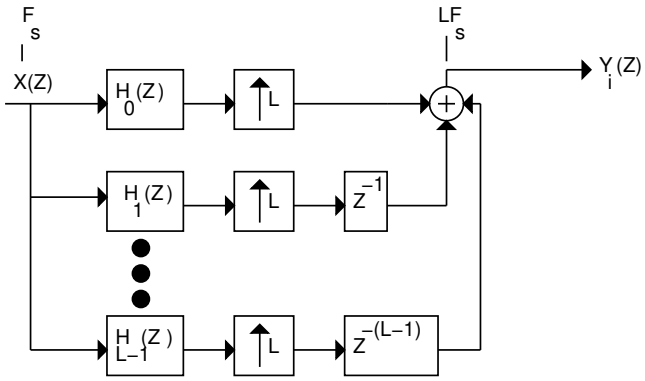

Figure 4.5 Polyphase decomposition of the interpolation filter.

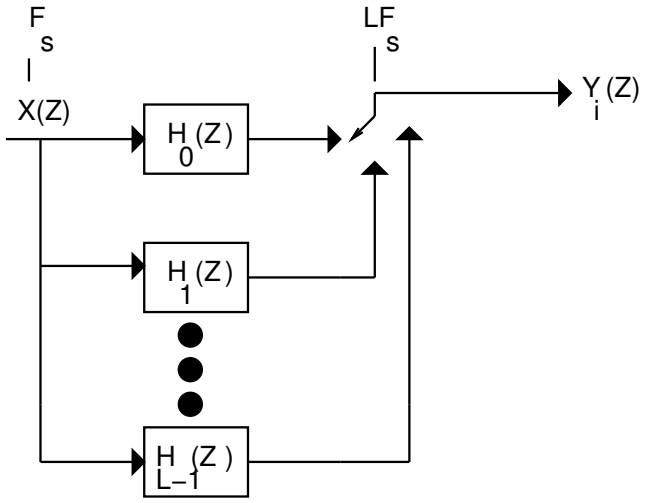

Figure 4.6 One possible realization of polyphase decomposed interpolation filter.

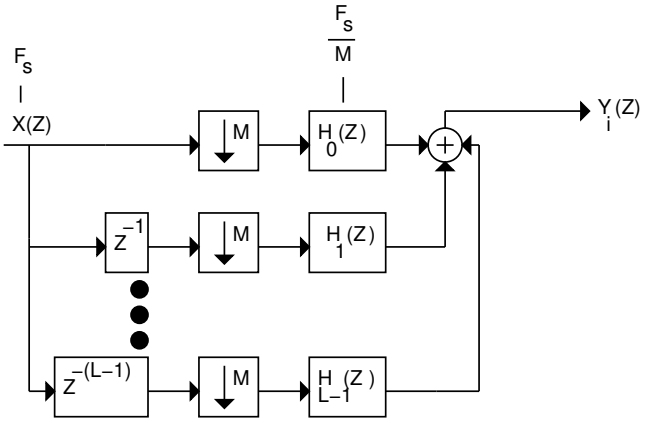

Figure 4.7 Polyphase decomposition of the decimation filter.

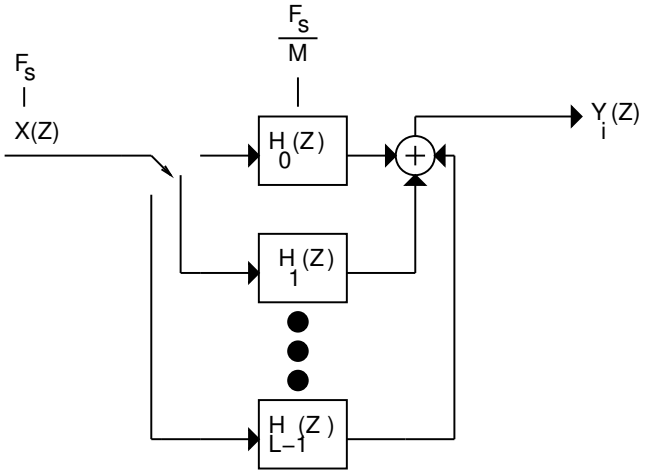

Figure 4.8 One possible realization of the polyphase decomposed decimation filter

The advantage of the polyphase decompositions is that the computation can always be performed with the lower clock frequency, resulting in either the possibility of reducing supply voltage in order to minimize power dissipation or the use of the pipelining/interleaving (P/I) technique in order to minimize the area [71].

In the case of FIR filters with symmetrical or anti symmetrical coefficients (linear phase FIR filters), the symmetry of the coefficients can be exploited in order to further reduce the area. The maximum number of symmetrical sub-filters in sampling rate conversion with different filter tap and sampling-rate combinations are listed in Table 4.2 [72].

Table 4.2 Maximum number of symmetrical sub-filters.

	Odd number of coefficients	Even number of coefficients
Odd L or M	1	1
Even L or M	2	0

4.3.2 Efficient realizations of FIR filters

In this section, the two main structures of digital FIR filters, namely the direct form and the transposed direct form, are presented. The pros and cons of both of them are considered and some methods for reducing the amount of hardware in filter realizations are also discussed.

4.3.2.1 Direct form structure

The folded direct form FIR filter structure is presented in Fig. 4.9. Folding is only

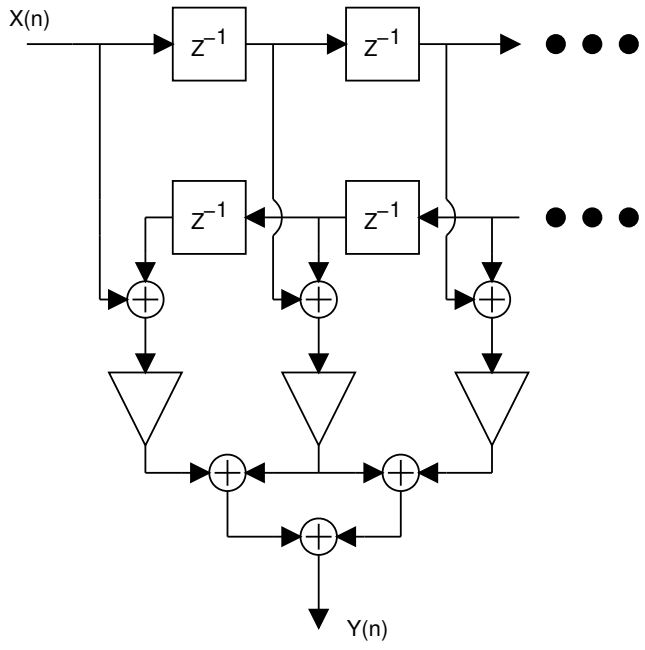

Figure 4.9 Folded regular direct form FIR filter structure.

applicable when the FIR filter has a linear phase, i.e. the coefficients are either symmetrical or anti-symmetrical relative to the center coefficient. Applications where the coefficients may not be symmetrical are, for example, sub-filters in polyphase decompositions and in the predistortion filters for phase error correction.

In the folded regular direct form, the number of bits needed in the delay elements

are defined by the number of input bits rather than the required word length of the filter, which may lead to a reduced amount of hardware. Another advantage of the transposed direct form is that, if the filter coefficients are updated (for example, in programmable filters), the effect is seen immediately at the filter output. On the other hand, the delay path from the registers to the output can be straightforwardly pipelined if the latency is allowed [72].

4.3.2.2 Transposed direct form structure

The transposed direct form structure is presented in Fig. 4.10. Basically the amount of

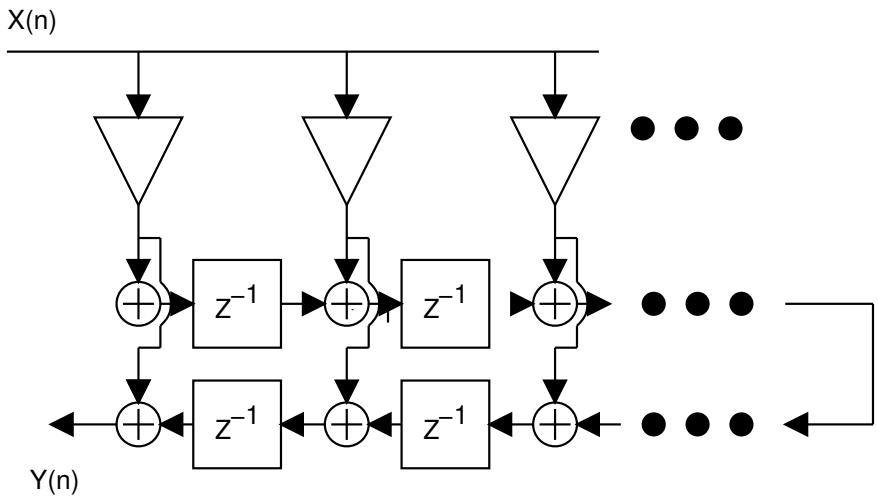

Figure 4.10 Folded transposed direct form FIR filter structure.

hardware is almost the same in direct form and transposed direct form FIR filters. The benefit of the transposed direct form is that the maximum delay path in this structure is shorter compared to the regular direct form, resulting in faster performance. Also, the redundant arithmetic addition, such as carry-save addition, may be applied to the adders between the register stages resulting in increased speed. This, however, requires a doubling of the register elements, because the signal is divided into carry and sum signals and an additional adder before the output to sum up the carry and sum signals. Transposed direct form filters can also be pipelined, but it may be more difficult than in a regular direct form structure, at least in the case of pipelining a single adder stage.

The subexpression sharing method can be applied to both of the structures, reduc-

ing the amount of hardware effectively [73],[74], [75].

4.3.2.3 Pipelining/interleaving technique

The pipelining/interleaving (P/I) technique [71] is applicable when, for example, the same filtering operation is to be performed for several independent data streams. Instead of using K times the same hardware, parallel data streams are interleaved in time, and the computation is made with a K times higher clock frequency. Finally, the data stream is deinterleaved either in K or some other amount of parallel data streams. When having K parallel data streams to be filtered with the same filter, the same hardware for filter coefficients may be used. However, register stages has to be added to the filter. Instead of one delay element we have to have K delay elements. The P/I structure of the polyphase interpolation filter is presented in Fig. 4.11. The main drawback of this

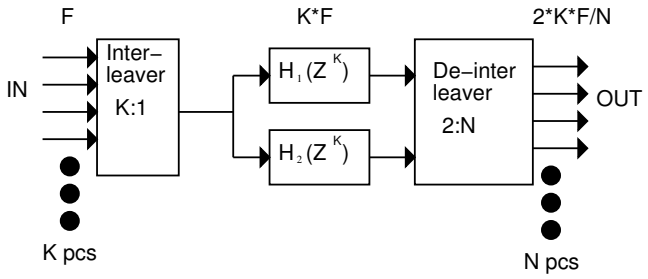

Figure 4.11 Pipelined/Interleaved polyphase filter.

method is that the additional register stages increase the power consumption, because the amount of the registers is the same as in parallel realization, but the clock frequency is higher.

4.3.2.4 Reduction of the sign bit load

In the filters that are realized with the SD coefficients, the sign bit loading due to the sign bit extension in shift and add operations may become a problem. This is because the load at the sign bit position may slow down the circuit or it may require a large amount of buffering. The loading is avoided by the usage of the constant vector addition method proposed in [72]. In this method, the sum of sign bit extension vector is computed a priori and added to the output of the filter. In this case, the sign bit extension does not have to be performed inside the filter, and the extensive loading of the sign bit is avoided.

4.3.2.5 Word length effects and scaling

In order to get the best possible trade off between the signal-to-noise ratio and *ACLR* at the output of the digital filter and the amount of hardware, the minimum accuracy needed has to be found. Basically there are three sources of noise due to the finite precision arithmetic in the FIR filter. The first source is the quantization of the result of the multiplication. The second is the noise due to the truncation at the filter output, and the third is the noise due to the overflows.

The variance of the quantization noise due to the finite internal word length is

$$\sigma_n^2 = \frac{\Delta^2}{12} = \frac{2^{-2B}}{3}, \tag{4.29}$$

where Δ is 2^{-B+1}, and B is the internal word length of the filter. It is assumed here that the computation is made in a 2's complement format and that the most significant bit (sign bit) has the weight of $2^0 = 1$, so the absolute value of the number is less than, or equal to, 1. Assuming an N tap transposed form filter, the noise variance at the filter output due to the internal word length quantization is

$$\sigma_{iwl}^2 = N\sigma_n^2. \tag{4.30}$$

If we want to keep the noise added by the filtering operation below the desired noise level indicated by B_o output bits, the internal word length of the filter should be

$$B \geq B_o + \frac{ln(N)}{2ln(2)} \approx B_o + 0.721ln(N). \tag{4.31}$$

The noise variance added due to the quantization at the filter output is

$$\sigma_{op}^2 = \frac{\Delta_o^2}{3} = \frac{2^{-2B_o}}{3}, \tag{4.32}$$

in which $\Delta_o = 2^{-B_o+1}$ and B_o is the number of bits at the filter output after quantization. This means that quantization at the filter output has less effect on the signal-to-noise ratio than shortening the internal word length of the filter. In the case of cascaded filter stages, only minor hardware savings can be achieved by truncation of the filter output.

The third source of noise is the noise generated by overflows in the filter. The overflows of the filter may be avoided if the internal word length of the filter is selected so that the maximum presentable number in the filter is

$$M_{output} = M_{input} \sum_{n=0}^{N-1} |h(n)|, \tag{4.33}$$

where M_{input} is the maximum value at the filter input [76]. However this usually gives over-pessimistic values for the bandlimited signals. Other methods for discovering the dynamic range of the signal are presented in [76],[77] and [78], but they do not guarantee an overflow-free performance. The noise due to the overflows may be minimized by using saturating logic in the adders [78]. This leads to clipping instead of overflows, which reduces remarkably the generated noise. Because the maximum output value is dependent on the statistics of the signal, the number of internal bits needed can also be approximated with simulations, while the saturation logic may be applied in order to minimize the effect of occasional overflows.

Chapter 5

Methods for direct digital frequency synthesis and modulation

The core of the digital QAM modulator is the direct digital synthesizer (DDS). The most often used methods for the direct digital frequency synthesis are the look-up table (LUT) based frequency synthesis methods and the methods based on the CORDIC vector rotation algorithm. These methods are described in the following sections.

5.1 Direct digital frequency synthesizers using the look-up table method.

The most straightforward approach for digital modulation is the LUT-based method used in, for example, [79],[80], [35]. In this method, the carrier is formed by addressing the memory with the phase value and mapping the current phase to the value of the sinusoidal signal. Then the generated carrier is multiplied with the modulating data by using digital multipliers. Advantages of this method are the good frequency resolution and relative simplicity. Drawbacks are that the multipliers and memories may require a large area, especially with high resolutions, and memories may become the speed bottleneck at high sampling rates.

With a look-up table method, it is possible to realize every function $y = f(x)$ of variable x by mapping the value x to the output y with some mapping element, usually a memory block. The accuracy of the mapping depends on the resolution of the input value x and the area (=accuracy) of the mapping element. When this is applied to the

frequency synthesis, the variable to be mapped is usually the phase of the sinusoid and is mapped to the corresponding amplitude value [81]. The principle of the LUT-based frequency synthesis is presented in Fig. 5.1. $\Delta\phi$ is the phase increment, which is

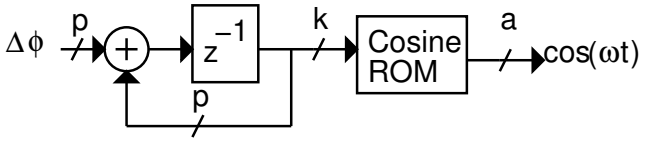

Figure 5.1 LUT-based DDS

integrated over time with a digital integrator also called a phase accumulator. The phase value obtained by integration is then used as the address to the ROM memory, which maps the phase value to the desired amplitude value. The accuracy of the synthesis is controlled by the values of p, k and a (Fig. 5.1) [79], [80].

The enhancement of the performance of the synthesizer is usually achieved by increasing the values of k and a. The value p affects mostly the frequency resolution of the synthesizer that is not the speed bottleneck. The phase accumulator can be made almost arbitrarily fast with pipelining techniques, for example [82]. The increase of the values a and k results the increase of the area of the ROM memory. The increase of this area may slow down the operation of the memory block and limit the achievable frequency band. Therefore, several techniques of memory size reduction have been developed.

The compression techniques can be roughly divided into three categories, namely the symmetry reduction techniques [83], [84], [85], segmentation techniques [86], and subtractive techniques [87], [88].

The symmetry reduction techniques are based on the fact that the shape of the sinusoid is the same in each quadrant, so only the first quadrant should be coded into the memory. The values in other three quadrants are obtained by controlling the sign of the phase and magnitude (Fig. 5.2). In quadrature synthesizers, additional symmetry

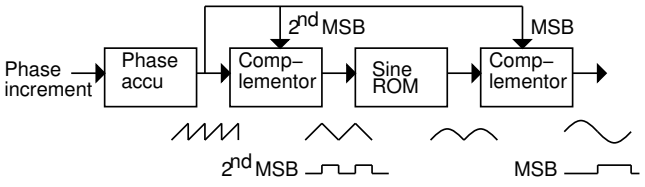

Figure 5.2 Symmetry reduction technique.

reductions may be achieved by taking into account that [85]

$$\cos(\phi) = \sin\left(\frac{\pi}{2} - \phi\right), \quad 0 \leq \phi \leq \frac{\pi}{2}. \tag{5.1}$$

In the subtractive reductions, the number of output bits of memory is reduced by subtracting some simple function out of the sine function and by saving only the difference into the memory. The simplest subtractive method, called sine-phase difference algorithm, is to subtract the phase out of the sine and add it to the output of the memory [87], but also more complex functions such as double trigonometric [88], slope optimized linear function [89], quadratic [90] and quad-line approximation [91] have been used.

With segmentation techniques, the large memory is divided into smaller units by exploiting the trigonometric identities, for example

$$\begin{aligned} \sin(\phi(n)) &= sin\left(\frac{\pi}{2}(A(n)+B(n)+C(n))\right) \\ &= \sin\left(\frac{\pi}{2}(A(n)+B(n))\right)\cos\left(\frac{\pi}{2}C(n)\right) \\ &\quad +\cos\left(\frac{\pi}{2}(A(n)+B(n))\right)\sin\left(\frac{\pi}{2}C(n)\right), \end{aligned} \tag{5.2}$$

where $A(n)$, $B(n)$, and $C(n)$ are sub-buses of the phase value bus $\phi(n)$. For symmetry reasons $\phi(n)$ is limited to be $0 \leq \phi(n) \leq \frac{\pi}{2}$. If $C(n)$ is chosen to be small (i.e. few least significant bits, *LSBs*), Eq. (5.2) may be approximated with

$$\sin(\phi(n)) \approx \sin\left(\frac{\pi}{2}(A(n)+B(n))\right) + \cos\left(\frac{\pi}{2}(A(n)+\langle B(n)\rangle)\right)\sin\left(\frac{\pi}{2}C(n)\right), \tag{5.3}$$

where $\langle B(n)\rangle$ is the mean value of $B(n)$. Now the first term of Eq. (5.3) can be stored in a coarse ROM and the second term in a fine ROM (Fig. 5.3).

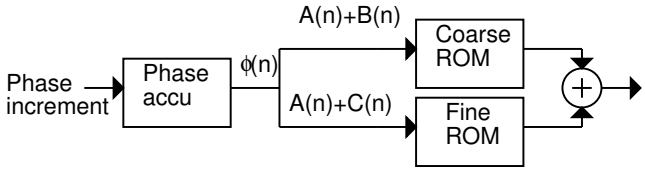

Figure 5.3 Memory compression by trigonometric approximation.

A wide variety of segmentation techniques have been presented during the past few years using trigonometric identities [83], [86], [92], interpolating [93], [94], and interpolating with nonlinear addressing [95]. With these methods, symmetry and subtractive reductions are usually also applied. The effectiveness of the compression method is a trade-off between the memory size and complexity of the additional hardware re-

quired to form the actual sinusoidal output. A comparison of memory compression ratios obtained with various compression methods is presented in [95]; however, the comparison of the memory compression ratios does not tell the whole truth, since the highest compression ratios may be achieved with the methods that require an extensive amount of additional hardware. A comparison of the respective areas required for implementation of various DDS circuits is presented in Table 5.1 in Section 5.3

It is preferable to use the combination of the different compression methods, for example, subtraction reduction, symmetry reduction, and segmentation [87], [96]. The speed of the synthesizer can be increased with, for example, the parallel structures [97].

5.2 CORDIC vector rotation algorithm based frequency synthesizer and modulator

CORDIC is an abbreviation for the coordinate rotation digital computer, which was introduced by J. Volder in 1959 [98].

CORDIC type algorithms are suitable for calculating trigonometric functions [98],[99], [100], hyperbolic trigonometric functions [101], logarithmic and exponent functions [102], sine wave generation [103], [104], linear transformations [105],[106], digital filters and matrix based DSP algorithms [105] and inverse trigonometric functions [107]. The CORDIC algorithm is also suitable for digital modulator/demodulator applications [108], [103],[6], [109] [110]. When applied to frequency synthesis and modulation, the algorithm provides as good a frequency resolution as the LUT-based method, but the modulation can be performed without any multipliers as in [103].

In hybrid realizations of the CORDIC algorithm [111], [112], [109], [110], a combination of LUT and the CORDIC rotator is used in order to further reduce the area of the synthesizer.

The CORDIC based modulation can be presented as follows. Discrete-time QAM modulation may be presented as a matrix multiplication as

$$\begin{aligned} M_r &= \begin{bmatrix} i(n)\cos(\phi(n)) + q(n)\sin(\phi(n)) \\ -i(n)\sin(\phi(n)) + q(n)\cos(\phi(n)) \end{bmatrix} \\ &= \begin{bmatrix} \cos(\phi(n)) & \sin(\phi(n)) \\ -\sin(\phi(n)) & \cos(\phi(n)) \end{bmatrix} \begin{bmatrix} i(n) \\ q(n) \end{bmatrix}, \end{aligned} \tag{5.4}$$

where $\phi(n) = \omega n T_s$, T_s is the sampling interval and ω is the desired angular frequency. Eq. (5.4) gives the desired modulation result as the first element of the resulting column

vector. By notifying that

$$\begin{bmatrix} \cos(\phi_1 \pm \phi_2) & \sin(\phi_1 \pm \phi_2) \\ -\sin(\phi_1 \pm \phi_2) & \cos(\phi_1 \pm \phi_2) \end{bmatrix} = \begin{bmatrix} \cos(\phi_2) & \pm\sin(\phi_2) \\ \mp\sin(\phi_2) & \cos(\phi_2) \end{bmatrix} \times \begin{bmatrix} \cos(\phi_1) & \sin(\phi_2) \\ -\sin(\phi_1) & \cos(\phi_1) \end{bmatrix}, \tag{5.5}$$

it may be stated that the rotation $\phi(n)$ in Eq. (5.4) can be achieved by the chain of multiplications

$$M_r \simeq M_{ra} = \prod_{k=0}^{N-1} \left(\begin{bmatrix} \cos(\phi_k) & \sigma_k(n)\sin(\phi_k) \\ -\sigma_k(n)\sin(\phi_k) & \cos(\phi_k) \end{bmatrix} \right) \begin{bmatrix} i(n) \\ q(n) \end{bmatrix}, \tag{5.6}$$

where $\sigma_k(n) = \{-1, 1\}$ and it is determined by

$$\begin{aligned} Z_0(n) &= \phi(n) \\ \sigma_k(n) &= \text{sign}(Z_k(n)) \\ Z_{k+1}(n) &= Z_k(n) - \sigma_k(n)\phi_k. \end{aligned} \tag{5.7}$$

Z_{k+1} in Eq. (5.7) is the residual angle to be rotated after k^{th} stage. To operate properly, Eq. (5.6) has two terms of convergence [98]. First, the angle $\phi(n)$ should be bounded and presentable by the sum of ϕ_k's

$$\sum_{k=0}^{N-1} \phi_k + \phi_{N-1} \geq |\phi(n)|, \quad -\pi \leq \phi(n) \leq \pi.$$

$$\phi_k \leq \phi_{N-1} + \sum_{i=k+1}^{N-1} \phi_i \tag{5.8}$$

Second, the residual rotation angle should converge to zero

$$\lim_{k \to \infty} |Z_k(n)| = 0, \tag{5.9}$$

which is always satisfied when

$$|Z_{k+1}(n)| \leq \phi_k. \tag{5.10}$$

Furthermore Eq. (5.6) can be simplified by taking $\cos(\phi_k)$ as a common subexpression. This results in

$$M_{ra} = \begin{bmatrix} \cos(\phi_0) & \sigma_0(n)\sin(\phi_0) \\ -\sigma_0(n)\sin(\phi_0) & \cos(\phi_0) \end{bmatrix} \times$$

$$\prod_{k=1}^{N-1}\cos(\phi_k)\left(\begin{bmatrix}1 & \sigma_k(n)\tan(\phi_k)\\ -\sigma_k(n)\tan(\phi_k) & 1\end{bmatrix}\right)\begin{bmatrix}i(n)\\ q(n)\end{bmatrix}. \quad (5.11)$$

By selecting

$$\phi_k = \arctan\left(2^{-k+1}\right) \quad (5.12)$$

and $\phi_0 = \frac{\pi}{2}$ to satisfy the Eq. (5.8), we may write the Eq. (5.11) in a form

$$M_{ra} = K\begin{bmatrix}0 & \sigma_0(n)\\ -\sigma_0(n) & 0\end{bmatrix}\times \prod_{k=1}^{N-1}\left(\begin{bmatrix}1 & \sigma_k(n)2^{-k+1}\\ -\sigma_k(n)2^{-k+1} & 1\end{bmatrix}\right)\begin{bmatrix}i(n)\\ q(n)\end{bmatrix}, \quad (5.13)$$

where

$$K = \prod_{k=1}^{N-1}\cos(\phi_k). \quad (5.14)$$

The values of ϕ_k and K can be computed beforehand because the number of stages N is known and because the sign of $cos(\phi_k) = cos(-\phi_k)$ is always positive, when $0 \leq \phi_k < \frac{\pi}{2}$. The value of K is only dependent on the number of stages, approaching 1.6468 with large N. If the constant scaling factor of the modulated signal is allowed, the effect of K can be discarded resulting in the modulation equation

$$M_b = \begin{bmatrix}0 & \sigma_0(n)\\ -\sigma_0(n) & 0\end{bmatrix}\times \prod_{k=1}^{N-1}\left(\begin{bmatrix}1 & \sigma_k(n)2^{-k+1}\\ -\sigma_k(n)2^{-k+1} & 1\end{bmatrix}\right)\begin{bmatrix}i(n)\\ q(n)\end{bmatrix}. \quad (5.15)$$

Eq. (5.15) can be computed with the series of N CORDIC rotation blocks, which may be realized by using only adders/subtracters and negators in the zero stage. The input-output relation of the stages are

$$\begin{aligned} X_{k+1}(n) &= X_k(n) + \text{sign}(Z_k(n))\,Y_k(n)\,2^{-k+1}\\ Y_{k+1}(n) &= -\text{sign}(Z_k(n))\,X_k(n)\,2^{-k+1} + Y_k(n)\\ Z_{k+1}(n) &= Z_k(n) - \text{sign}(Z_k(n))\,\phi_k \end{aligned} \quad (5.16)$$

for k^{th} stage and

$$\begin{aligned} X_1(n) &= \text{sign}(Z_0(n))\,Y_0(n)\\ &= \text{sign}(\phi(n))\,q(n)\\ Y_1(n) &= -\text{sign}(Z_0(n))\,X_0(n) \end{aligned}$$

$$
\begin{aligned}
&= -\text{sign}\left(\phi\left(n\right)\right) i\left(n\right) \\
Z_1\left(n\right) &= Z_0\left(n\right) - \text{sign}\left(Z_0\left(n\right)\right) \phi_0 \\
&= \phi\left(n\right) - \text{sign}\left(\phi\left(n\right)\right) \frac{\pi}{2}
\end{aligned} \tag{5.17}
$$

for the zero stage. $\phi\left(n\right)$ can be generated with the same kind of phase accumulator as is presented in Fig. 5.1. It should be noted that the values of ϕ_k are presented with the same notation as $\phi\left(n\right)$, usually in the two's complement format.

One possible architecture for the CORDIC rotator is presented in Fig. 5.4. It takes

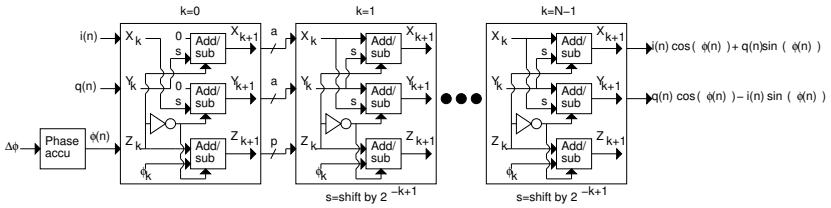

Figure 5.4 CORDIC vector modulator.

the data values $i\left(n\right)$ and $q\left(n\right)$ as the input and the modulated carrier is X output of the last stage. The accuracy of the modulation performed with the CORDIC rotator is dependent on the number of stages N, number of bits p (Fig. 5.4) on the phase computation path, and the number of bits a on the X/Y path. The frequency resolution is selected by the number of bits in the phase accumulator. The effects of N, a and p (Fig. 5.4) are analyzed in [113], [114] and [115]. The CORDIC may also be used as a frequency synthesizer if $q\left(n\right)$ and $i\left(n\right)$ are constants.

With the CORDIC rotator, no digital multipliers are needed for the modulation of the carrier, as is with the LUT-based digital frequency synthesizer. Also no memories are required, thus enabling pipelining. If only the frequency synthesis is performed, there is no or little advantage of the multiplier-free realization and the LUT-based or LUT/CORDIC hybrid may result in a smaller area. Very efficient hybrid mixers have also been presented [109], [110].

5.3 Survey of digital frequency synthesizers

Area, speed and resolution of various LUT- and CORDIC-based frequency synthesizer implementations are presented in Table 5.1. It can be noticed that, with a hybrid realization based on the combination of LUT and CORDIC vector rotation algorithm, effective savings of hardware may be achieved. It should be noted that if the "Mixer" feature column in Table 5.1 contains "No", at least three multipliers are required in addition to the reported hardware in order to enable quadrature modulation.

Area-speed-resolution trade-off between LUT and CORDIC realizations is analyzed in detail in [112].

Table 5.1 Comparison of DDS implementations.

Method	Type	Process	(P)hase, (A)mpl. resol.	Sampl. freq. [MHz]	Area [mm^2]	Mixer	Comments
Sunderland [83]	segm. LUT	3.5μm CMOS	P=14 A=12	7.5	48.1	No	External DAC
Nicholas [86]	segm. LUT	1.25μm CMOS	P=15 A=12	150	24.55	No	8-b DAC
Bellaouar [93]	interp. LUT	0.8μm CMOS	P=12 A=9	30	0.9	No	Core area, no DAC, IQ
Langlois1 [94]	interp. LUT	0.35μm CMOS	P=18 A=14	100	0.110	No	Core area, no DAC, IQ
Langlois2 [94]	interp. LUT	0.35μm CMOS	P=16 A=12	320	0.282	No	Core area, no DAC, IQ
Langlois3 [94]	interp. LUT	0.35μm CMOS	P=16 A=12	100	0.079	No	Core area, no DAC, IQ
Curticăpean [92]	seqm. LUT	0.35μm CMOS	P=16 A=16	30	0.23	No	Core area, no DAC, IQ
De Caro [116]	poly-nom.	0.35μm CMOS	P=14 A=12	80.4	0.31	No	Core area, no DAC, IQ
Yang [91]	segm. LUT	0.35μm CMOS	P=11 A=9	800	1.3	No	Core area, 9-b DAC
Gielis [103]	COR.	1.0μm bipolar	P=12 A=10	540	24.96	Yes	No DAC, IQ
Madisetti [117]	hybr. COR.	1.0μm CMOS	P=22 A=16	100	12.00	No	Core area, no DAC, IQ
Janiszewski [112]	hybr. COR.	0.35μm CMOS	P=20 A=16	310	3.72	No	Core area, no DAC
Curticăpean [118]	hybr. COR.	0.35μm CMOS	P=19 A=16	100	0.46	No	Core area, no DAC, IQ
Torosyan1 [109]	hybr. COR.	0.25μm CMOS	P=19 A=13	300	0.36	Yes	Core area, no DAC, IQ
Torosyan2 [109]	hybr. COR.	0.25μm CMOS	P=19 A=13	600	0.72	Yes	Core area, no DAC, IQ
Song1 [119]	hybr. COR.	0.35μm CMOS	P=18 A=16	150	1.4	No	Core area, no DAC, IQ
Song2 [110]	hybr. COR.	0.25μm CMOS	P=18 A=15	330	0.51	Yes	Core area, no DAC, IQ

5.4 Other digital modulation methods

5.4.1 Modulation to quarter of the sampling rate

Similarly to CORDIC modulation, multipliers are not needed with this method [36],[37] [120]. The upconversion is made to a quarter of the system clock frequency by multiplying the data values repeatedly by sequence $[1, 0, -1, 0]$. If the sequence $[+1, -1]$ is used, the data is upconverted to half of the sampling frequency. The only hardware required for this modulation is a negator and a multiplexer. The sign control can also be included in, for example, coefficients of a digital filter [37]. A shortcoming of the method is the fixed carrier frequency. This method can be expanded in order to generate carrier frequencies other than the quarter of a single sampling frequency. In this case, the method is called multistage modulation and is described in, for example, [121].

5.4.2 Frequency synthesis with nonlinear D/A converter

A memory-free method for frequency synthesis is presented in [122]. In this method, the phase-to-sinusoid mapping is achieved by a nonlinear D/A converter. While the usage of memories is avoided, the implementation of a good-quality nonlinear D/A converter is not trivial. Also, the amplitude modulation has to be performed in the analog domain; the advantage of DSP in modulation is lost.

Chapter 6

Current-steering digital-to-analog converter design

Current-steering digital-to-analog converters have become the mainstream architecture of D/A converters since the late 80's. This is due to following properties of current-steering architecture. With this it is possible to provide relatively large currents (10 to 20mA) to 50Ω load without buffering. The operation speed of a current-steering converter is determined by the ability to drive the gates of the switches, instead of the gain bandwidth product (GBW) of the buffer circuitry, as in resistor-string and switched-capacitor D/A converters. Sample rates of several hundreds of millions of samples per second can therefore be achieved, which makes the current-steering D/A converter the most suitable architecture for, for example, digital IF transmitters and direct digital synthesizers. In addition, only transistors are used to provide the output current; this enables the usage of the standard CMOS process, whereas high-accuracy resistors and capacitors are required in resistor-string or switched-capacitor converters.

This chapter describes the theory of current-steering D/A converters applied in the design of the prototype circuits presented in Chapter 7. As an introduction to the subject, some general issues considering the current-steering D/A converters and their performance metrics are discussed in Sections 6.1 and 6.2. In Section 6.3, the static linearity as a function of transistor mismatch is analyzed. The previously published linearity yield models are compared with simulations, and the yield model developed and published by the author [13] is presented. Section 6.4 discusses the previously published calibration methods that are used to improve the static linearity of the converters above 12 bits. Also the digital calibration method developed and published by

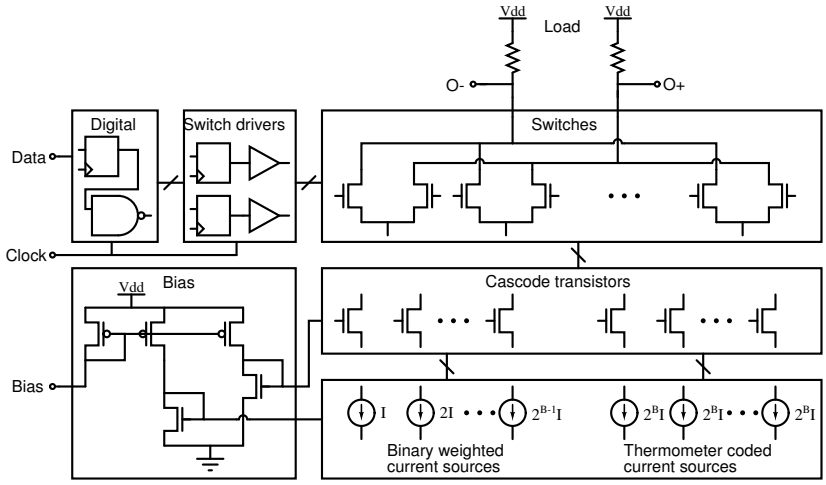

Figure 6.1 Block diagram of a typical current-steering D/A converter.

the author and his team [14] is presented. Section 6.5 considers the effects due to the output impedance variation. In Section 6.6, the signal conversion from discrete time digital to continuous time analog is discussed and mathematically modeled, while distortion sources such as transition asymmetry are analyzed. In Section 6.7, the timing nonlinearities are added to the model presented in Section 6.6. The effect of the jitter on the spectral performance is analyzed by simulations, and the performance degradation due to code-dependent clock load and power supply interference is demonstrated. The jitter analysis is also partially published by the author in [15]. Section 6.8 is about the layout techniques used to reduce the effect of the process gradients on the current source mismatch. The previously published methods are discussed, and methods used in the prototype circuit presented in Section 7.2 are described.

6.1 General description of the current steering D/A converter

A block diagram of a typical current-steering D/A converter is presented in Fig. 6.1 A current-steering D/A converter consists of current sources and cascode transistors that are often used to increase the impedance of the current source. Switches are used to combine the current of the current sources and form an output signal. Gates of the switches are driven by control signals decoded from input data with digital circuitry. Clock buffers and some analog bias circuitry are also required. These circuit blocks together form a complex mixed-mode circuit entity.

In an N-bit converter, there are 2^N input codes, of which one is zero, resulting

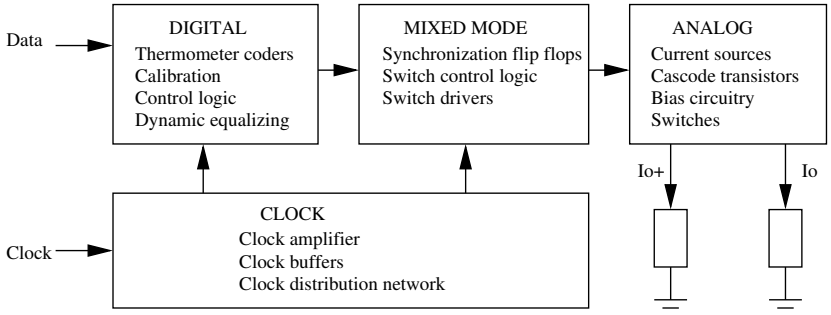

Figure 6.2 Domains of the D/A converter.

in a total current requirement of $2^N - 1$ times the current corresponding to the LSB (I_{lsb}). Current sources can be weighted (usually binary weighted), thermometer coded, (i.e. each current source has the same weight), or segmented (i.e. combination of binary weighted and thermometer coded), which is usually the case. The usage of binary weighted current sources reduces the number of switches, but introduces difficulties in switch-driver synchronization, since larger currents require larger switches, thus providing a larger capacitive load. The timing differences in switch control signals result in glitches, i.e. spikes in the output current. For example, MSB transition $01111 -> 1000$ in a binary-weighted D/A converter may cause a glitch, since the transition of the MSB bit is slower than the three LSB bits. It is therefore preferable to use as many as possible thermometer-coded bits at the MSB end in order to produce good synchronization of the most significant bit transitions, and use binary weighting for the LSB part. The practical limit for thermometer-coded bits is around 7, due to the increased area required for the switch drivers and the decoder logic.

A current-steering D/A converter can be divided into four types of circuitry domains each of them having a unique contribution to the performance of the converter. The domains of the D/A converter are presented in Fig. 6.2.

The digital domain of a D/A converter includes all of the pure digital circuitry of the converter. The digital domain is less sensitive to noise and interference. It is more like a source of interference, whose propagation to any of the other domains should be eliminated, or, at least, minimized. The digital circuitry usually consists of at least a thermometer decoder, since segmented architecture is often used. Calibration, which is often needed for accuracies of 12-bit or more, requires also some kind of digital circuitry. Switching activity equalization, which is used to decorrelate the switching activity from the input code, is also mostly performed in the digital domain.

In the mixed-mode domain, the control signals for the switches of the converter are formed and the switch control signals are synchronized with flip-flops. The synchronization flip-flops/latches usually convert the digital data signal from a single-ended

digital bit to an fully differential analog signal, which can then be transformed to an actual control signal with some simple logic circuitry and buffering. The power supplies of the mixed-mode domain are usually separated from the digital domain in order to avoid coupling and interference after synchronization. The transition activity in the digital domain is related to the input data, thus generating input data dependent interference, whose coupling to the mixed-mode domain can result in code-dependent timing jitter, and thus harmonic distortion at the converter output. In addition, even though the switch control signals in the mixed-mode domain are sensitive to interference, they are digital like signals switching from rail-to-rail, and therefore generate interference that can couple to the analog domain.

In the analog domain, the unit currents are generated and combined in order to form the analog output signal. Usually one or multiple cascode transistors are used to increase the output impedances of the current sources in order to reduce the distortion generated by the code dependent output impedance variation. The analog domain is sensitive to interference, but, whereas the sensitivity in the mixed-mode domain is mainly related to the jitter of the sampling instance, the sensitivity in the analog domain is continuous. All nodes in the analog domain (except the output nodes and gates of the switch transistors) are biased to a certain operating point, and they should remain stable even when switching occurs. Unfortunately this kind of stability is unachievable. Interference can, and will, couple to every possible node in the analog domain. Effects are more severe on one node than on another, and they can be reduced by various means, such as shielding with ground planes; however, coupling to the analog domain cannot be totally prevented.

The sensitivity of the clock domain is related to sampling. Interference in the clock domain will cause jitter. Because the clock takes care of all synchronization in the converter, the interference in the clock domain will be reflected in the performance of the whole converter. The higher the sampling frequency, the more severe the effect of the jitter on the performance of the converter. Unfortunately, the clock net, and thus the clock domain, has to be distributed around the converter because it is connected to both digital and mixed-mode domains. Therefore it is subjected to interference originated in both of these domains. It is also sensitive to the coupling of the analog output to the clock signal, but because the clock domain is not directly connected to the analog domain, this kind of coupling can be more easily avoided.

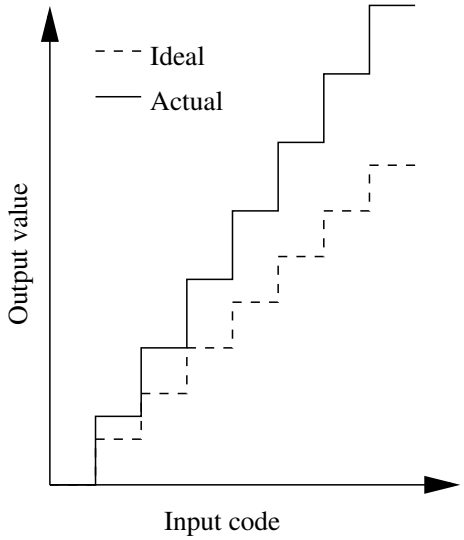

Figure 6.3 Gain error of the D/A converter.

6.2 Performance metrics

6.2.1 Static linearity: Gain error, DNL and INL

In the ideal case, the output current of the converter is linearly dependent on the input code, i.e. the input code "7" causes $7I_{lsb}$ current to flow to the output load. If the average LSB step is smaller or larger than was originally intended, there is gain error (Fig. 6.3). The gain error is usually not a problem, since the relative accuracy of the conversion is much more important than the absolute accuracy, i.e. it is more important than that with input code "8", the output is twice as large as with input code "4" regardless of the actual output values corresponding to "4" and "8". However, in some applications, such as I/Q transmitters, where two converters have to have equal gain, the gain error is an important parameter, since the imbalance between I and Q signal branches results in reduced sideband rejection in I/Q mixers and inter-channel interference in reception. Gain calibration between converters is therefore usually needed in that kind of application.

Within a single converter, the gain error is usually not a problem, since the error does not introduce nonlinearity, which would distort the output signal. More severe nonidealities are the Differential Nonlinearity (*DNL*) and Integral Nonlinearity (*INL*). They are nonlinearities mainly caused by a mismatch of the current source transistors and limited output impedance of the current branch. The *INL* of a D/A converter is defined as the difference between the analog output value and the straight line drawn between output values corresponding to the smallest and the largest input code, divided

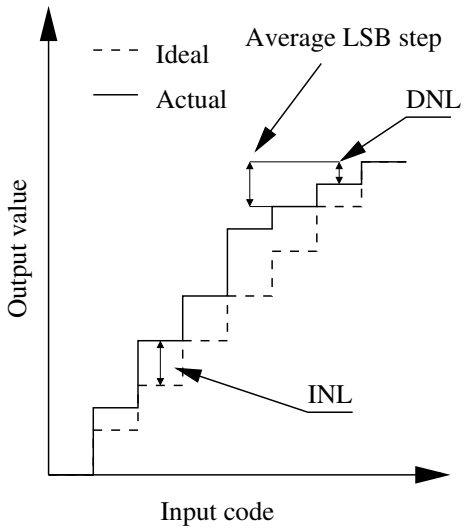

Figure 6.4 Differential Nonlinearity and Integral Nonlinearity.

by the average LSB step. *INL* is sometimes also defined as the difference of the analog output and best-fit straight line throughout the output values [123]. In this book, the former definition is used. *DNL* is defined as the variation of the step size relative to average LSB step. It should be noted that, because *DNL* and *INL* are obtained by referring the actual output to the linear output obtained by using the average LSB step, *INL* values corresponding to the smallest and the largest input codes are always zero. Definitions of *DNL* and *INL* presented in Fig. 6.4.

The information included in *DNL* can be further analyzed as follows. *DNL* tells us how much the step at code transition from k to $k+1$ actually differs from the average step size. If the value of *DNL* is less than -1, the converter is no longer monotonic, i.e. the output value decreases as the input code increases. Fully thermometer-coded converters are always monotonic. Monotonicity can only be lost with binary weighted converters. A converter is considered to have N bit accuracy if *DNL* does not exceed ± 0.5LSB, i.e. the output value is within the bounds of quantization error.

INL is a curve that tells us how much the actual output corresponding to input code k differs from linear (not ideal) output. *INL* is often reported as a single figure (maximum $INL = x$) or as a range ($INL = \pm x$), but these figures will not tell us much about a single converter. More important is the shape of *INL* curve, because it includes all information about the low-frequency nonlinearity of the converter.

DNL and *INL* are important performance metrics in the sense that they define the limits of the linearity of the converter. Linearity is further degraded by dynamic non-idealities as the signal frequency and sampling rate are increased. The role of *DNL*

and *INL* of the converters as a performance metric is two-fold. On the one hand, they define the quality of the low frequency operation, on the other, with modern Nyquist-rate converters with sample rates from 100MHz to 1GHz, the dynamic nonlinearities are clearly limiting the performance, and therefore the static linearity of the converter is quite meaningless.

6.2.2 Linearity and noise: SNR, SFDR, THD, SINAD and ENOB

In addition to *DNL* and *INL*, various figures of merit are commonly used to indicate the quality of a D/A converter.

A signal-to-noise ratio (*SNR*) is the ratio of the signal power (usually the power of a sinusoidal signal) and the noise power integrated over certain signal band in decibels. When Nyquist-rate converters are considered, *SNR* is usually defined for the signal band from 0 to half of the sample rate. The upper limit of the *SNR* is defined by quantization noise,

$$\sigma_n^2 = \frac{\Delta^2}{12}, \tag{6.1}$$

in which Δ is the size of the quantization step. Presenting the sinusoidal signal with an amplitude A with N-bits results in

$$\Delta = \frac{2A}{2^N}, \tag{6.2}$$

and *SNR* of

$$SNR = 6.02N + 1.76. \tag{6.3}$$

SNR will be further degraded by other noise sources, such as thermal noise and 1/f-noise.

Very often interference spurs exist in the output spectrum of the converter. These spurs are most often due to harmonic distortion, i.e nonlinearity, of the converter, but they can also occur for some other reason, such as coupling. The figure of merit that tells us the level of highest spurious tone relative to the signal power is the spurious free dynamic range (*SFDR*), which is given in decibels relative to (given) signal power.

Total harmonic distortion (THD) is the sum of the power of the all harmonic distortion components, relative to signal power. THD is also usually given in decibels.

The signal-to-noise and distortion ratio *SINAD* is the sum of noise and distortion power relative to signal power. With *SINAD*, it is possible to calculate the effective number of bits (*ENOB*) of the converter. Assuming the SINAD is determined, *ENOB* can be defined as

$$ENOB = \frac{SINAD - 1.76}{6.02}. \tag{6.4}$$

In current steering D/A converters, the figures of merit that are generally announced

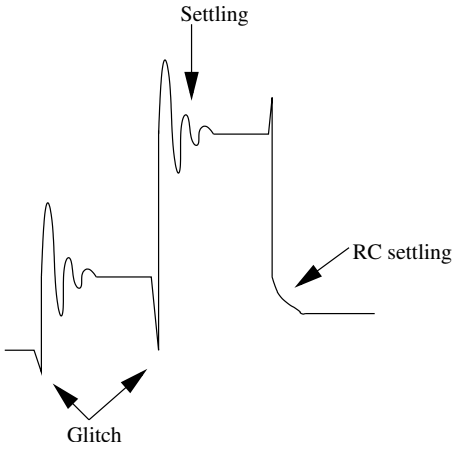

Figure 6.5 Settling and glitches.

are *DNL*, *INL* and *SFDR* since they give a comparable figure for the linearity of the device. However, very often the *SFDR* (and therefore also the *SINAD*) of the device is so low that the *ENOB* of the device is quite far from the value defined by *INL* or *DNL*.

6.2.3 Time domain performance: Settling and glitches

In Fig. 6.5, a typical time-domain waveform of a current steering D/A converter is presented. There are two major time-domain properties that are often referred to, namely settling and glitches.

Settling is the time required by the output signal to settle to its current value after transition with certain accuracy, e.g. within a 0.5 LSB range of the "ideal value". There are several types of settling, depending on the load of the converter. With RC-loads, the settling is low-pass type RC-settling, indicating filtering of the higher-frequency components. With RLC type of loads, the settling takes place as damped oscillations. Because the settling is a filtering operation for the output signal, no serious harm results from it as long as the filtering does not affect the desired signal band (from 0 to half of the sampling frequency in general).

"Glitch" is a spike on the output signal. Glitches can be generated in various ways, and often they are referred as a form of signal degradation. Glitches can be problematic when they are generated by the timing differences in synchronization. Also unequal rise and fall times in binary weighted converters may generate glitches even with differential output, whereas in thermometer coded converters the asymmetry in transitions is usually canceled by differentiality. It is essential to understand what the phenomena behind the glitch is, because most of the glitches or spikes at the output do not introduce any kind of nonlinearity to the signal. As long as the glitch is linearly dependent

on the signal or its discrete time derivative, it only carries part of the signal energy, or indicates that some signal energy is missing, as later demonstrated in Section 6.6.

6.3 Models and relations of transistor mismatch and static linearity

In this section, the analysis of the statistical behavior of *INL* yield as a function of the current source mismatch is presented. It is demonstrated how the segmentation of current sources affects the statistical behavior of *INL* and *DNL*. Various yield models published during the past decade are analyzed and compared by simulations. In addition, regression models for *DNL* and *INL* yields are presented. These models are applicable in cases of 10 to 16 bits with 1 to 6 and 1 to 3 thermometer coded MSB bits for *INL* and *DNL*, respectively. The regression model presented is published by the author in [13].

6.3.1 Current source mismatch and yield

DNL and *INL* are generated by the mismatch of the drain currents of the current source transistors. The mismatch of the drain currents is due to global and local variations during the manufacturing process of a silicon chip. The relative variance of the difference between the two transistors, having the same dimensions, orientation and operation point, can be expressed as [124] [125]

$$\begin{aligned}\frac{\sigma_{Id}^2}{I_d^2} &= \frac{4\sigma_{vt}^2}{(V_{gs}-V_T)^2}+\frac{\sigma_\beta^2}{\beta^2}\\ &= \frac{4A_{vt}^2}{WL(V_{gs}-V_T)^2}+\frac{4S_{vt}^2D^2}{(V_{gs}-V_T)^2}+\frac{A_\beta^2}{WL}+S_\beta^2D^2,\end{aligned} \tag{6.5}$$

where A_{vt}, A_β, S_{vt}, and S_β are process related constants, V_{gs} is the gate-source voltage of the transistor, V_T is the threshold voltage of the transistor, W is the channel width and L is the channel length.

The magnitudes of *INL* and *DNL* are dependent on the mismatch of the drain currents of the current sources and the structure of the D/A converter. The following figures represent *DNL*s of a 14-bit thermometer-coded (Fig. 6.6) and binary-weighted (Fig. 6.7) D/A converter. It can be seen that the maximum value of *DNL* is strongly dependent on the structure of the D/A converter, i.e. segmentation has an effect on *DNL*. Obviously, it is dependent on the number of current sources switched when moving from one input code to another.

In Fig. 6.8 and Fig. 6.9 the standard deviations of *DNL* values of the thermometer

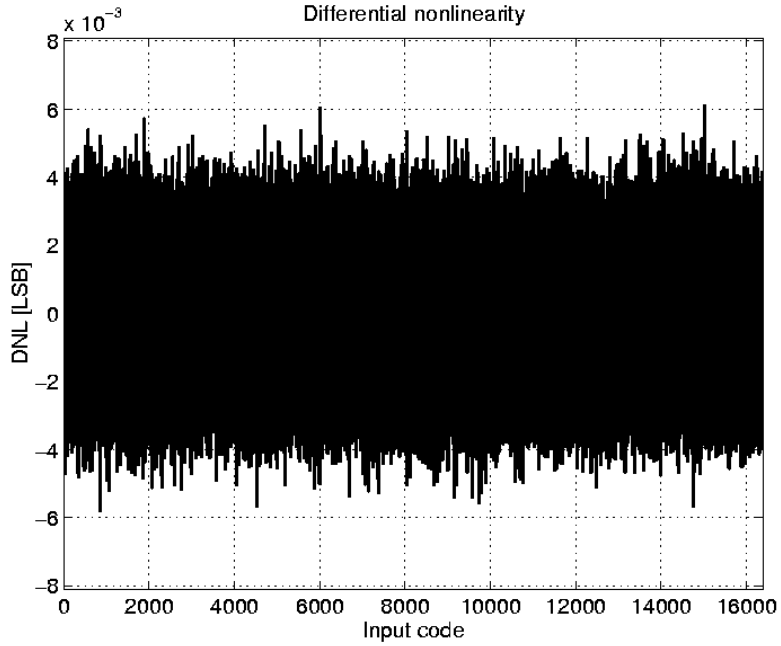

Figure 6.6 20 *DNL*s of the thermometer coded D/A converter.

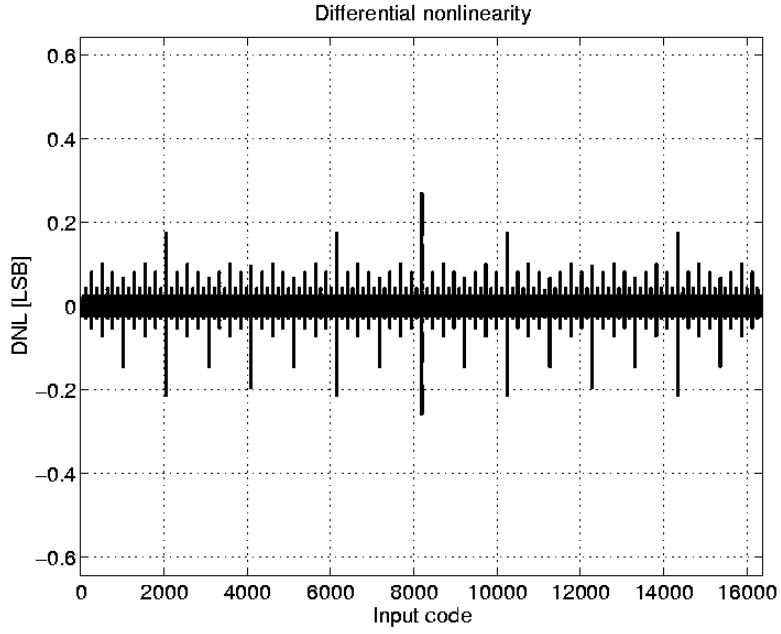

Figure 6.7 20 *DNL*s of the binary weighted D/A converter.

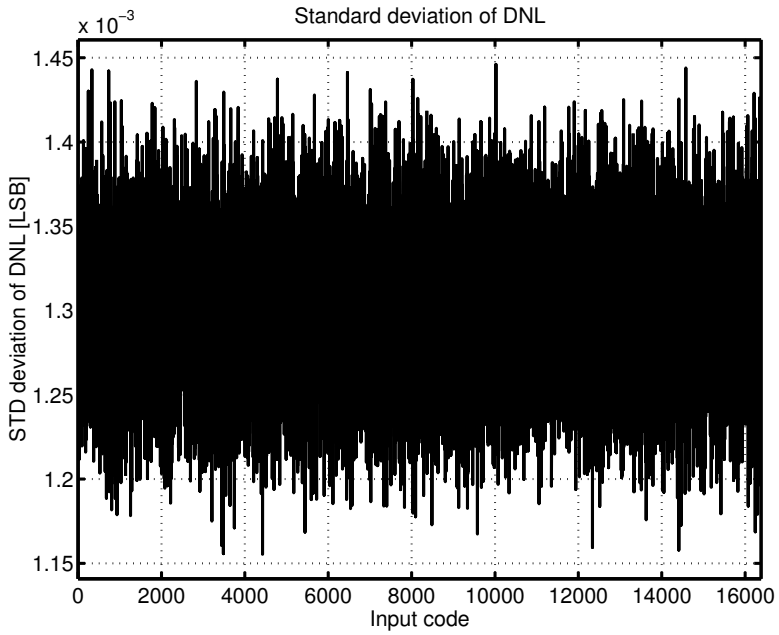

Figure 6.8 Standard deviation of 500 *DNL*s of the thermometer coded D/A converter.

coded and binary weighted D/A converters are presented. It can be observed, that the variance of the maximum value of *DNL* is

$$\sigma^2_{DNLmax} = B\sigma^2_{lsb}, \tag{6.6}$$

[126], where B is the maximum number of unit current sources switched in the single code transition and σ_{lsb} is the relative standard deviation of the error current of a unit current source.

The same kind of simulations were carried out for the *INL* of the thermometer-coded and binary-weighted D/A converters. *INL* curves for these converters are presented in Fig. 6.10 and Fig. 6.11 The dependency of *INL* on the converter structure is analyzed in the following sections.

6.3.2 Statistical model of the static linearity

In order to analyze the static behavior of *INL*, equations for the variances and covariances of *INL* as function of the input code are derived for the thermometer-coded and binary-weighted D/A converter.

The single-ended output current of the thermometer coded D/A converter that cor-

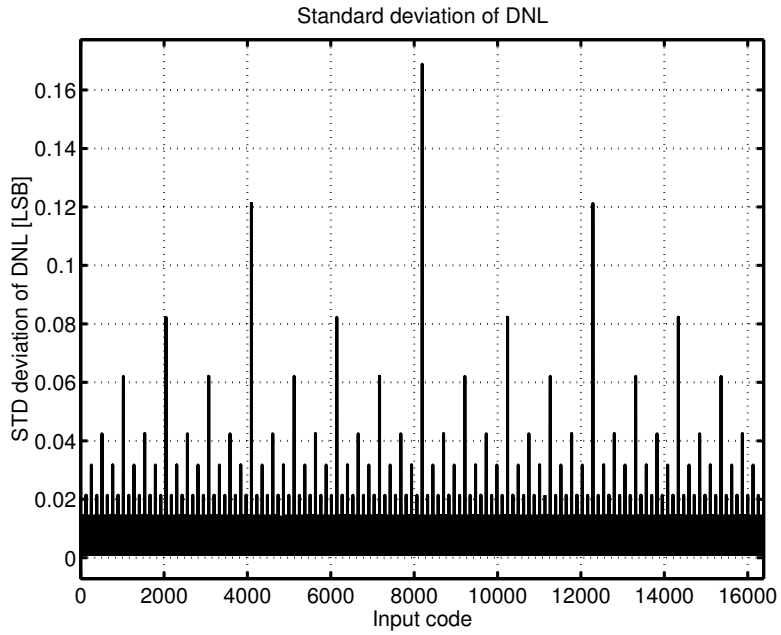

Figure 6.9 Standard deviation of 500 *DNL*s of the binary weighted D/A converter.

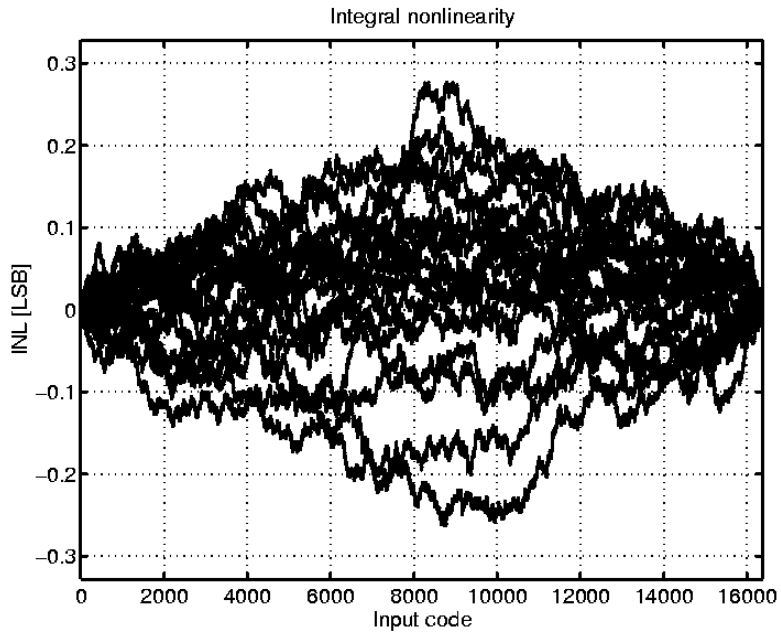

Figure 6.10 20 *INL*s of the thermometer coded D/A converter.

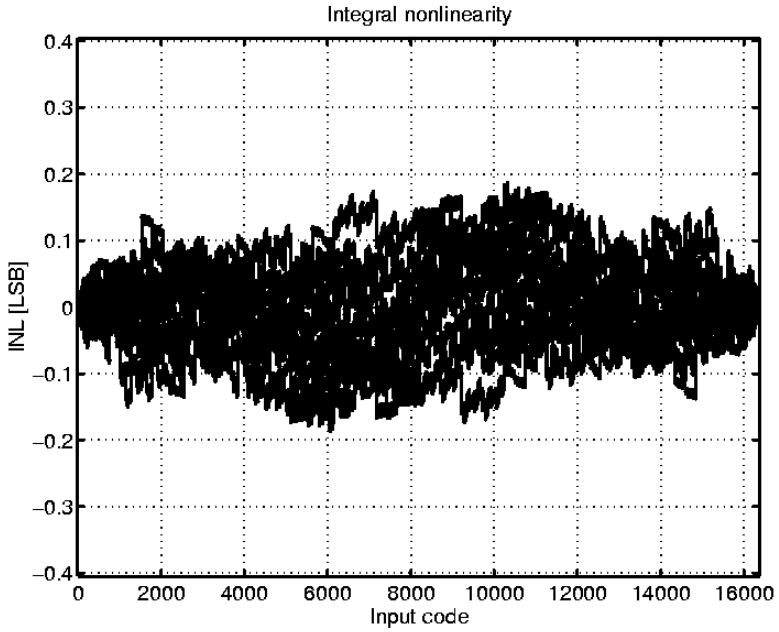

Figure 6.11 20 *INL*s of the binary weighted D/A converter.

responds to the offset binary type input code k can be written as

$$I_{out}(k) = \sum_{i=0}^{k} I_{lsb} + \sum_{i=0}^{k} \varepsilon_i, \quad I_{out}(0) = 0, \varepsilon_0 = 0 \tag{6.7}$$

where I_{lsb} is the ideal LSB current and ε_i is the error current of i^{th} current source. ε_i is assumed to have a normal distribution with zero mean and variance $\sigma^2_{I_{lsb}}$. Because the shape of the envelope of the *INL* is the same for both the single-ended and differential case, only the single-ended case is examined here for simplicity. The actual LSB step can be found by fitting the straight line between the end points of the curve obtained by sweeping the value of k from 0 to $2^N - 1$, where N is the number of bits resulting in

$$I_{lsba} = I_{lsb} + \sum_{i=1}^{2^{N-1}} \frac{\varepsilon_i}{2^N - 1}, \tag{6.8}$$

where the summation term corresponds to the gain error of the converter. Now it is

possible to define INL relative to I_{lsba} as the function of k as

$$\begin{aligned} INL_t(k) &= \frac{I_{out} - kI_{lsba}}{I_{lsba}} \\ &= \frac{I_{lsb}\left(\sum_{i=1}^{k}\frac{\varepsilon_i}{I_{lsb}} - k\sum_{i=1}^{2^N-1}\frac{\varepsilon_i}{I_{lsb}\left(2^N-1\right)}\right)}{I_{lsb}\left(1+\sum_{i=1}^{2^N-1}\frac{\varepsilon_i}{2^N-1}\right)}. \end{aligned} \tag{6.9}$$

If it is assumed that the denominator of the right side of Eq. (6.9) is approximately I_{lsb}, Eq. (6.9) can be approximated as

$$INL_t(k) \approx \sum_{i=1}^{k}\varepsilon_{ri} - k\sum_{i=1}^{2^N-1}\frac{\varepsilon_{ri}}{2^N-1}, \tag{6.10}$$

where $\varepsilon_{ri} = \frac{\varepsilon_i}{I_{lsb}}$ is error of the i^{th} current source relative to LSB current I_{lsb}. The subscript t in $I_t(k)$ stands for the thermometer coded D/A converter.

The INL of the converter can also be presented with the matrix notation

$$INL_t = K^T E - \frac{K^T I_{vt} I_{vt}^T E}{2^N-1} = K^T\left(I_t - \frac{I_{vt}I_{vt}^T}{2^N-1}\right)E = K^T A_t E_t, \tag{6.11}$$

where K is a $2^N-1\times 2^N$ matrix with each column containing k 1's starting from the topmost element and 2^N-k zeros, I_{vt} is $2^N-1\times 1$ unity vector, I_t is a unity matrix, and E_t is $2^N-1\times 1$ vector containing the relative error values. From Eq. (6.11), the covariance matrix of INL values at code values k_1 and k_2 can be calculated, and result in

$$\begin{aligned} C_t &= E\left[INL_t \times INL_t^T\right] \\ &= E\left[K^T A_t E_t E_t^T A_t^T K\right] \\ &= \left(K^T A_t A_t^T K\right)\sigma_{lsb}^2, \end{aligned} \tag{6.12}$$

where $\sigma_{lsb}^2 = \mathrm{VAR}[\varepsilon_{ri}]$ is the relative error variance of the unit current source. The elements of C_t are

$$c_t(k_1,k_2) = \left(MIN(k_1,k_2) - \frac{k_1k_2}{2^N-1}\right)\sigma_{lsb}^2 \tag{6.13}$$

and they represent the covariances between $INL_t(k_1)$ and $INL_t(k_2)$. By setting $k_1 = k_2 = k$, the variance of the INL at the code value k is obtained.

$$\sigma_{INL_t}^2(k) = \left(k - \frac{k^2}{2^N-1}\right)\sigma_{lsb}^2 \tag{6.14}$$

which are the values on the diagonal of C_t. The correlation coefficient between the two *INL* values is

$$\begin{aligned}\rho_t\,(k1,k2) &= \frac{COV\,[INL_{t\,k1},INL_{t\,k2})]}{\sigma_{Ik1}\sigma_{Ik2}} \\ &= \frac{MIN\,(k_1,k_2)-\frac{k_1k_2}{2^N-1}}{\sqrt{k_1-\frac{k_1^2}{2^{N-1}}}\sqrt{k_2-\frac{k_2^2}{2^{N-1}}}}.\end{aligned} \tag{6.15}$$

In a similar manner, the covariance matrix for the binary weighted and segmented D/A converter can be computed. The *INL* of the binary-weighted converter can be written as

$$\begin{aligned}INL_b &\approx B^T W_{db}E - \frac{B^T W_b I_{vb}^T W_{db}E}{2^N-1} \\ &= B^T\left(I_b - \frac{W_b I_{vb}^T}{2^N-1}\right)W_{db}E = B^T A_b E_b,\end{aligned} \tag{6.16}$$

where B is an $N\times 2^N$ matrix with each column containing the binary presentation of k, W_b is an $N\times 1$ weighting vector with elements $w_i = 2^{N-1-i}, i=[0...N-1]$, $W_{db} = \sqrt{Diag(W_b)}$, and E_b is an $N\times 1$ error vector of normally distributed random variables with zero mean and variance σ_{lsb}^2.

By combining the *INL* matrix equations of the binary-weighted and thermometer coded converters, the *INL* equation for the segmented converter is obtained as

$$\begin{aligned}INL_s &\approx S^T W_{ds}E_s - \frac{S^T W_s I_{vs}^T W_{ds}E_s}{2^N-1} \\ &= S^T\left(I_s - \frac{W_s I_{vs}^T}{2^N-1}\right)W_{ds}E_s \\ &= S^T A_s E_s,\end{aligned} \tag{6.17}$$

where

$$S = \begin{bmatrix} K_s \\ B_s \end{bmatrix}, \tag{6.18}$$

$$W_s = \begin{bmatrix} 2^B \times I_{vt} \\ W_b \end{bmatrix}, \tag{6.19}$$

K_s is a matrix with columns containing $2^{N-B}-1$ thermometer coded MSB bits and B_s with columns containing B binary weighted LSB bits, $W_{ds} = \sqrt{Diag(W_s)}$, and E_s is $2^{N-B}+B-1\times 1$ error vector of normally distributed random variables with zero mean and variance σ_{lsb}^2.

The *INL* covariance matrix of the segmented D/A converter is

$$\begin{aligned} C_s &= E\left[INL_s \times INL_s^T\right] \\ &= E\left[S^T A_s E_s E_s^T A_s^T S\right] \\ &= \left(S^T A_s A_s^T S\right)\sigma_{lsb}^2 \end{aligned} \tag{6.20}$$

with elements

$$c_{sk_1,k_2} = \left(S_{k_1}^T W_{ds}^T W_{ds} S_{k_2} - \frac{k_1 k_2}{2^N - 1}\right)\sigma_{lsb}^2 \tag{6.21}$$

The variances of *INL* can be found on the diagonal of C_s resulting in

$$\sigma_{INL_{sk}}^2 = \left(k - \frac{k^2}{2^N - 1}\right)\sigma_{lsb}^2. \tag{6.22}$$

The correlation coefficients are given by

$$\begin{aligned} \rho_{sk_1,k_2} &= \frac{COV\left[INL_{sk_1}, INL_{sk_2})\right]}{\sigma_{Ik1}\sigma_{Ik2}} \\ &= \frac{S_{k_1}^T W_{ds}^T W_{ds} S - \frac{k_1 k_2}{2^N - 1}}{\sqrt{k_1 - \frac{k_1^2}{2^N - 1}}\sqrt{k_2 - \frac{k_2^2}{2^N - 1}}}. \end{aligned} \tag{6.23}$$

To give an example of the differences between the correlation matrices of a thermometer-coded and binary-weighted D/A converter, correlation matrices for 3-bit converters are presented. The correlation matrix R_t for the 3-bit thermometer-coded D/A converter is

$$R_t = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0.6455 & 0.4714 & 0.3536 & 0.2582 & 0.1667 & 0 \\ 0 & 0.6455 & 1 & 0.7303 & 0.5477 & 0.4000 & 0.2582 & 0 \\ 0 & 0.4714 & 0.7303 & 1 & 0.7500 & 0.5477 & 0.3536 & 0 \\ 0 & 0.3536 & 0.5477 & 0.7500 & 1 & 0.7303 & 0.4714 & 0 \\ 0 & 0.2582 & 0.400 & 0.5477 & 0.7303 & 1 & 0.6455 & 0 \\ 0 & 0.1667 & 0.2582 & 0.3536 & 0.4714 & 0.6455 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}. \tag{6.24}$$

Correlation matrix R_b for the 3-bit binary weighted D/A converter is

$$R_b = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -0.2582 & 0.4714 & -0.4714 & 0.2582 & -1 & 0 \\ 0 & -0.2582 & 1 & 0.7303 & -0.7303 & -1 & 0.2582 & 0 \\ 0 & 0.4714 & 0.7303 & 1 & -1 & -0.7303 & -0.4714 & 0 \\ 0 & -0.4714 & -0.7303 & -1 & 1 & 0.7303 & 0.4714 & 0 \\ 0 & 0.2582 & -1 & -0.7303 & 0.7303 & 1 & -0.2582 & 0 \\ 0 & -1 & 0.2582 & -0.4714 & 0.4714 & -0.2582 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix} . \tag{6.25}$$

Correlation matrix for the 3-bit converter with 2 segmented MSB bits

$$R_s = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -0.2582 & 0.4714 & -0.4714 & 0.2582 & -1 & 0 \\ 0 & -0.2582 & 1 & 0.7303 & 0.5477 & 0.4000 & 0.2582 & 0 \\ 0 & 0.4714 & 0.7303 & 1 & 0.1667 & 0.5477 & -0.4714 & 0 \\ 0 & -0.4714 & 0.5477 & 0.1667 & 1 & 0.7303 & 0.4714 & 0 \\ 0 & 0.2582 & 0.4 & 0.5477 & 0.7303 & 1 & -0.2582 & 0 \\ 0 & -1 & 0.2582 & -0.4714 & 0.4714 & -0.2582 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix} . \tag{6.26}$$

The following observations can be made about the covariance matrices C_t, C_b, and C_s, and correlation matrices R_t,R_b, and R_s:

1) Values on the edges of the covariance matrices are zero which means that *INL* values at $k = 0$ and $k = 2^N - 1$ are zero, and therefore they can be discarded from the statistical model of *INL*.

2) If the edge rows and columns are discarded from C_t, the resulting matrix C_{t2} is a symmetrical positive definite matrix and hence invertible. Its inverse is also positive definite. The square root of the positive definite matrix is non-singular.

3) If the edge rows and columns are discarded from C_b and C_s, the resulting matrices C_{b2} and C_{s2} are singular, their determinants are zero, and therefore they are not invertible. They are symmetrical positive semi-definite matrices.

4) The correlation factors of *INL*s of the thermometer-coded D/A converters are positive and their values are less than one.

5) The correlation factors of the segmented and binary-weighted D/A converters can be negative. The correlation factors of the binary-weighted converter be-

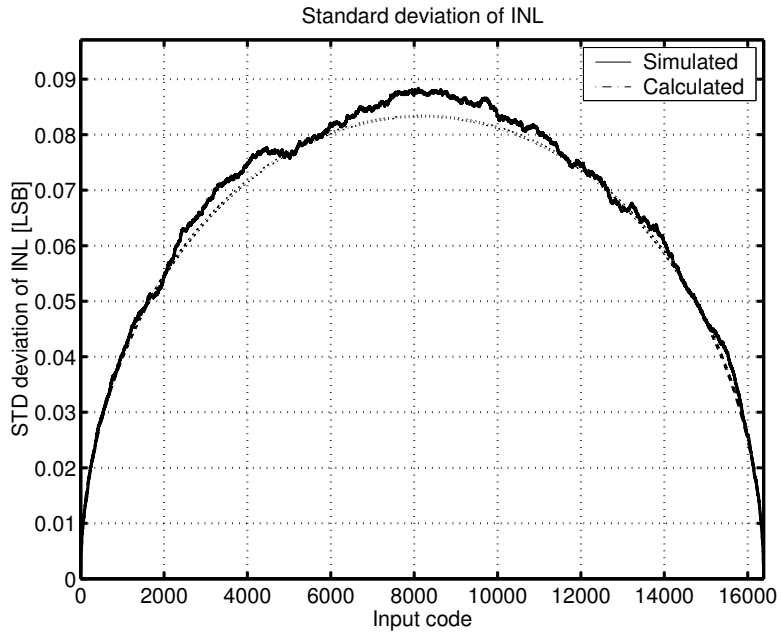

Figure 6.12 RMS of 500 *INL*s of the thermometer-coded D/A converter.

tween *INL*s positioned symmetrically relative to the midpoint of the *INL* curve have the correlation factor -1, indicating the perfect negative correlation.

Figs 6.12 and 6.13 represent the simulated and computed values of the standard deviation of *INL* for the thermometer-coded and binary-weighted D/A converter. It can be seen that the simulated values follow quite accurately the values computed with Eq. (6.14).

Once the variances, covariances and correlation factors are calculated, it is possible to try to find a statistical model for the *INL*. One good candidate for the statistical distribution model of the *INL* is the multivariate normal distribution (MVN) . The probability density function (PDF) of the MVN is

$$f_x(x) = \frac{1}{(2\pi)^{\frac{n}{2}} |C_x|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-m_x)C_x^{-1}(x-m_x)}, \tag{6.27}$$

where $x = [x_1, x_2, ...x_n]^T$ is a vector of n real valued random variables, $m_x = [m_1, m_2, ...m_n]^T$ is a vector containing the means of x, and C_x is a symmetric positive definite covariance matrix.

For INL_{t2} (*INL*, from which the first and last values are discarded because they are

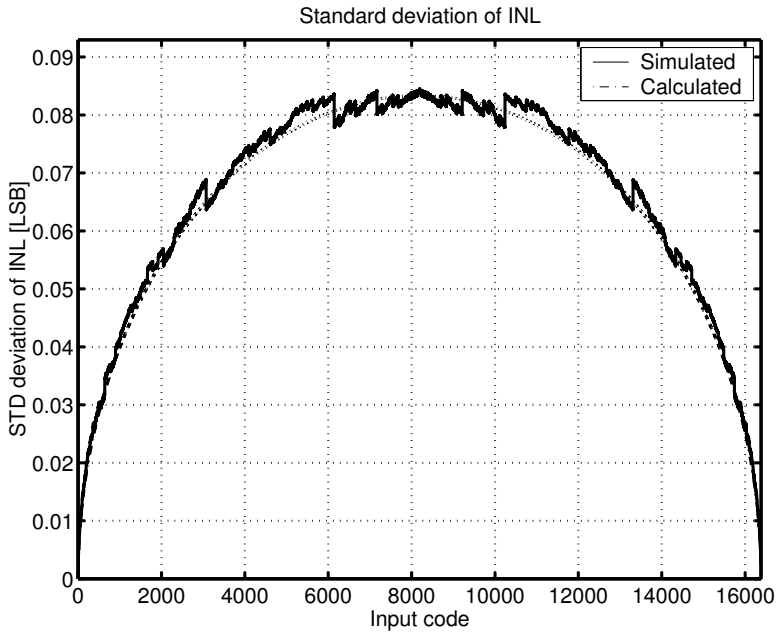

Figure 6.13 RMS of 500 *INL*s of the binary-weighted D/A converter.

always zero), the MVN density function can be written as

$$f_{INL_{t2}}(INL_{t2}) = \frac{1}{(2\pi)^{\frac{2^N-2}{2}} |C_{t2}|^{\frac{1}{2}}} e^{-\frac{1}{2} INL_{t2}^T C_{t2}^{-1} INL_{t2}}, \tag{6.28}$$

but for INL_{b2} and INL_{s2} (INL, from which the first and last values are discarded because they are always zero) this is not possible, since C_{b2} and C_{s2} are singular and therefore not invertible, and because their determinants are zero.

Once the PDF is defined, the cumulative distribution function (CDF) of INL_{t2} can be defined as

$$F_{INL_{t2}}(INL_{t2}) = \frac{1}{(2\pi)^{\frac{2^N-2}{2}} |C_{t2}|^{\frac{1}{2}}} \int_{-\infty}^{I_1} \ldots \int_{-\infty}^{I_{2^N-2}} e^{-\frac{1}{2} INL_{t2}^T C_{t2}^{-1} INL_{t2}} dI_1 \ldots dI_{2^N-2}. \tag{6.29}$$

In order to validate the assumption of the multinormal distribution, simulations were performed by generating vectors of random variables that have the desired variance and covariance properties. This is achieved by variable transformation

$$X = C_{t2}^{-\frac{1}{2}} INL_{t2}, \quad INL_{t2} = C_{t2}^{\frac{1}{2}} X, \tag{6.30}$$

where X is a vector of an independent N(0,1) distributed random variable. Simulation

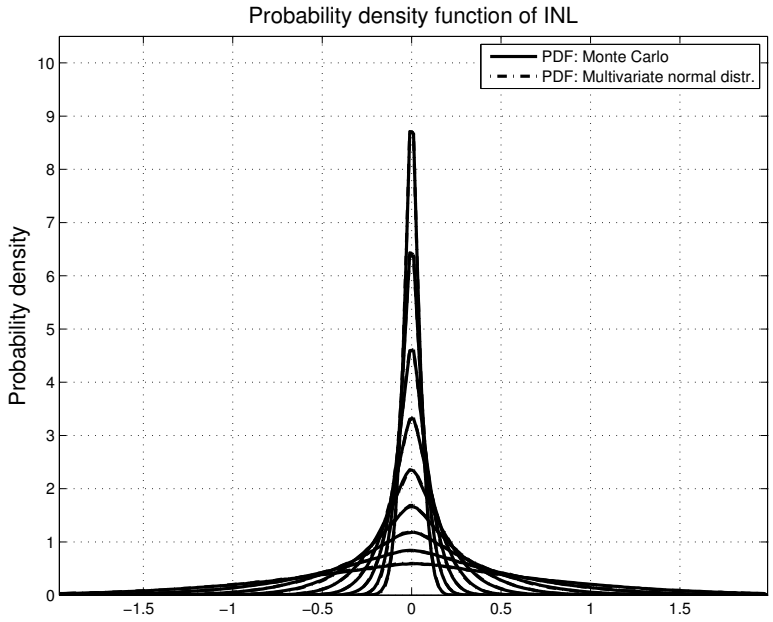

Figure 6.14 PDF of *INL* of a thermometer-coded D/A converter with various values of σ.

results for the 8-bit thermometer coded D/A converter are presented in Figs. 6.14-6.17. It can be observed, that the results obtained with Monte-Carlo simulation follows quite accurately the MVN distribution generated by variable transformation. Similarly, the comparison can be made for the segmented 8-bit converter with 4 binary-weighted bits. The results are presented in Figs. 6.18-6.21.

Simulation results indicate that the assumption of the multivariate normal distribution is valid for *INL* values of the thermometer-coded D/A converter. Due to the fact that the covariance matrices for the binary-weighted and the segmented D/A converter are not invertible, the multivariate normal distribution is not directly applicable. Singular Value Decomposition can be used in order to calculate an approximation of the inverse of covariance matrices; however, this does not solve the problem related to the zero-valued determinant of the singular covariance matrices.

Eq. (6.29) is probably solvable by numerical methods [127],[128],[129]; however, in all practical cases ($N > 12$), the number of random variables ($2^N - 2$) is much larger than the number of variables (7) that the program presented in [127] could handle. However, if we take into account the development of computer resources, the $N = 14$ case is perhaps solvable with modern computers.

Despite the difficulties solving Eq. (6.29), we may write the exact definition of *INL* yield in the case of thermometer-coded D/A converter (assuming that the distribution

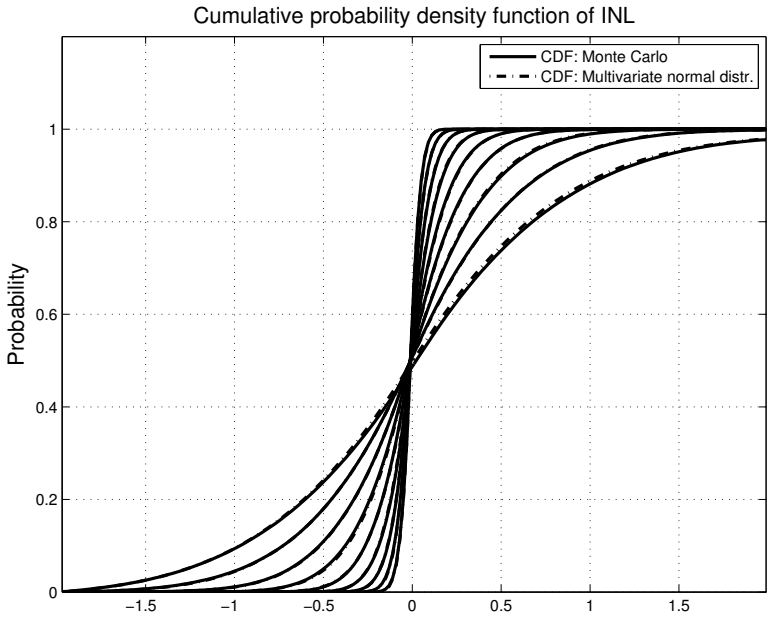

Figure 6.15 CDF of *INL* of a thermometer-coded D/A converter with various values of σ.

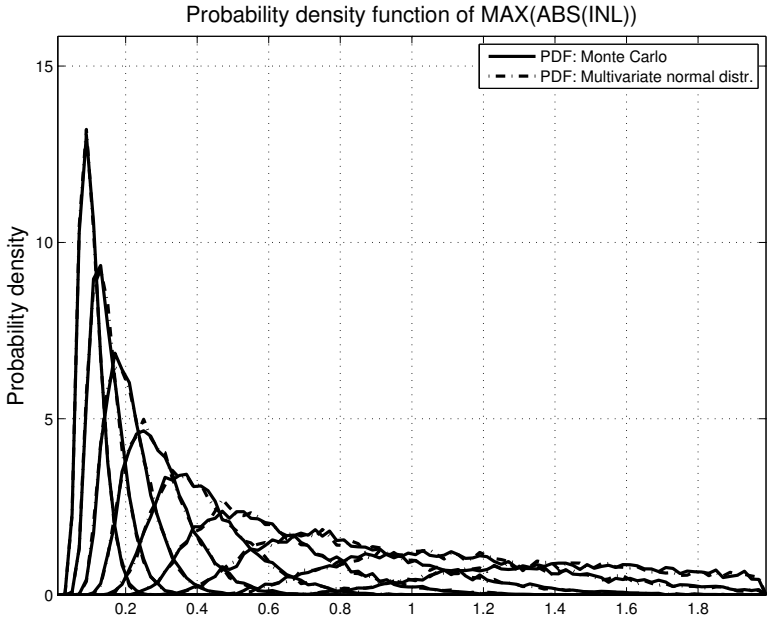

Figure 6.16 PDF of the maximum absolute value of *INL* of a thermometer-coded D/A converter with various values of σ.

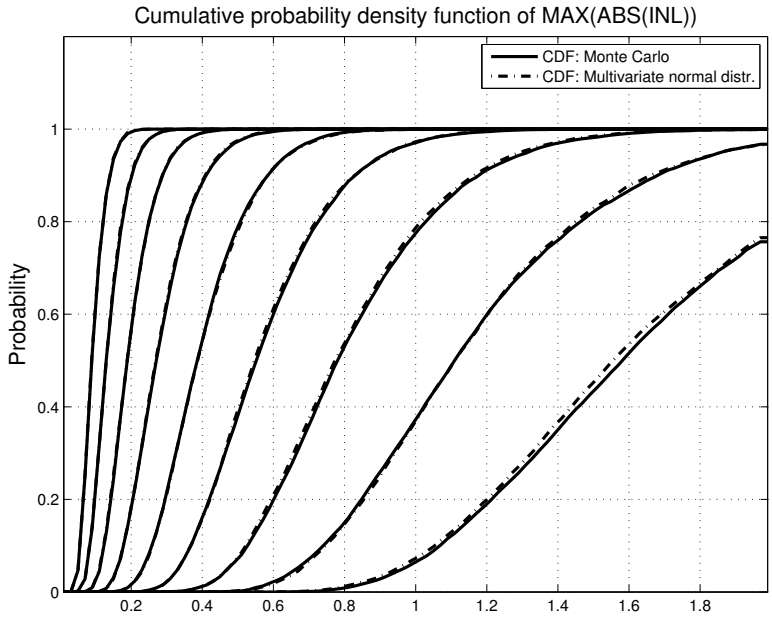

Figure 6.17 CDF of the maximum absolute value of *INL* of a thermometer-coded D/A converter with various values of σ.

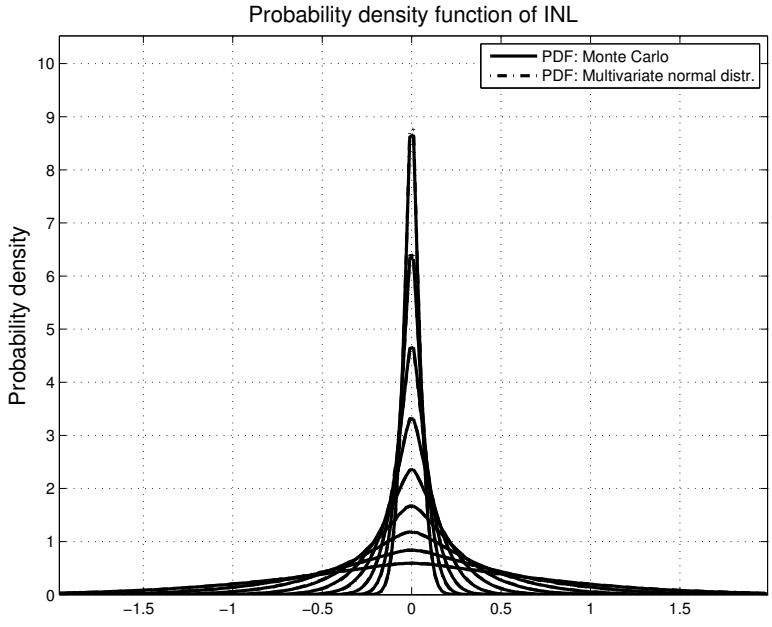

Figure 6.18 PDF of *INL* of a segmented D/A converter with various values of σ.

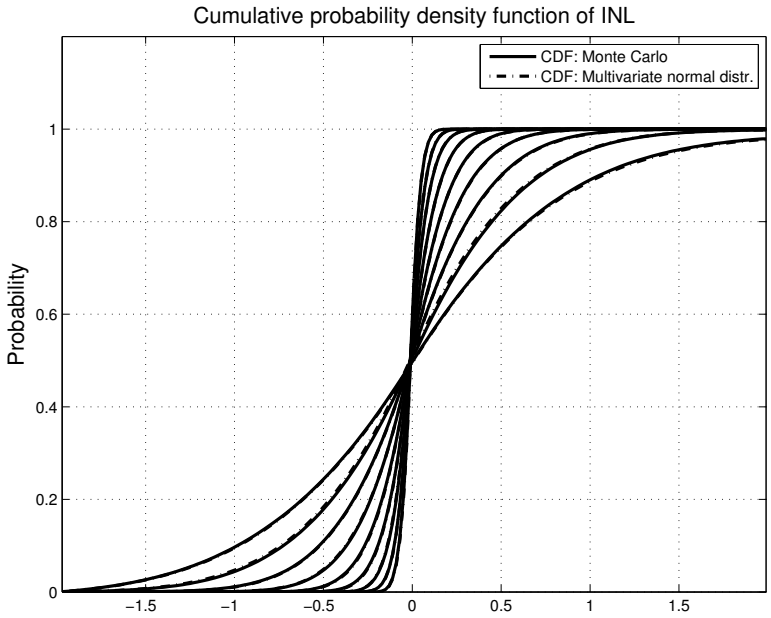

Figure 6.19 CDF of *INL* of a segmented D/A converter with various values of σ.

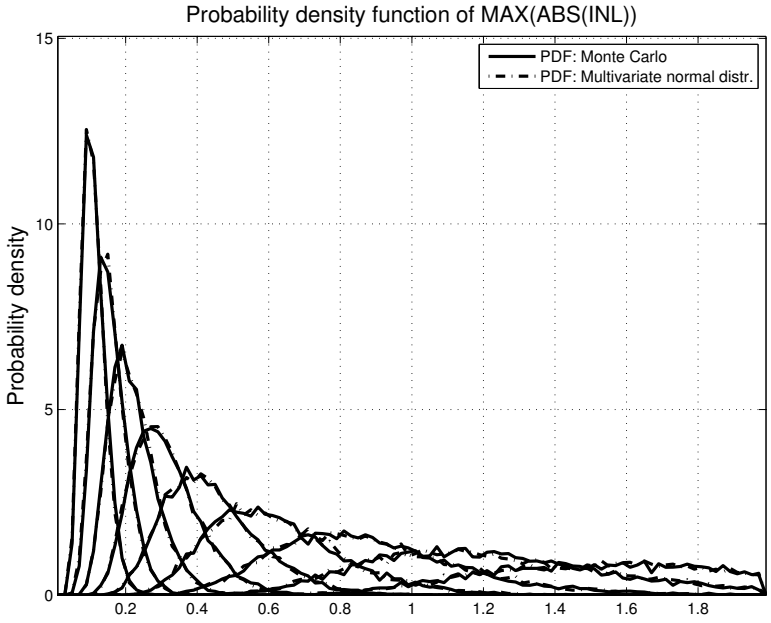

Figure 6.20 PDF of the maximum absolute value of *INL* of a segmented D/A converter with various values of σ.

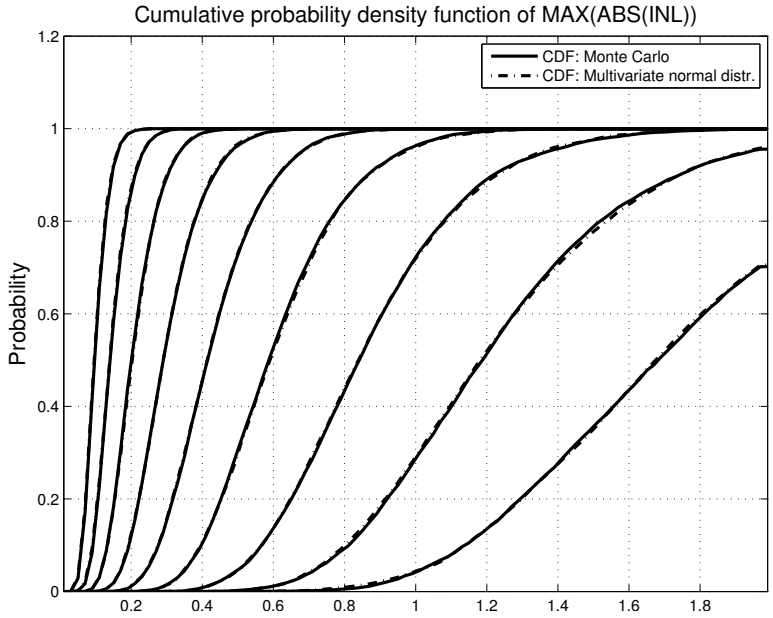

Figure 6.21 CDF of the maximum absolute value of *INL* of a segmented D/A converter with various values of σ.

follows the MVN distribution)

$$Yield_{INL_t} = P(|INL_{t2}| < 0.5) \times 100\% \tag{6.31}$$

$$= (2F_{INL_{t2}}(INL_{t2}) - 1) \times 100\%, \tag{6.32}$$

$$INL_{t2} = [0.5, 0.5...0.5]^T. \tag{6.33}$$

This equals the proportion of converters selected from a random set of converters that fulfills *INL* specification $|\text{INL}| \leq 0.5$. Thus, because the yield is a function of the relative variance of the unit current source, which is a function of transistor area and operation point [124], we could, in theory, discover the area and operating point that would give a desired yield.

The result that the *INL* of the converter follows the multivariate normal distribution is also presented by Cong an Geiger in [130]. The yield analysis presented in this book was performed by the author most likely at the same time as Cong and Geiger without knowledge of their work. The analysis is included in this book also to complete the analysis of the previously published yield models, performed in the next section.

6.3.3 INL yield models

Because the solution of Eq. (6.29) is quite hard to compute, the relation of $Yield_{INL}$ and σ_{lsb}^2 has to be defined by other means. Various methods [124], [131], [126], [132] has been suggested for evaluating the maximum current source variance allowed in order to achieve the desired accuracy of the converter (or, in other words, a high-enough yield). In [124] the assumption is made that the *INL* values are independent, and thus the probability to have a certain *INL* value can be computed as a product of normally distributed probabilities over the input range. Using this assumption yields pessimistic results. This is commented on [133] and the model is improved in [131] so that only the two most probable error values at MSB transition are taken into account; however, this model results in optimistic results as demonstrated by simulations later in this section.

In [126], an observation is made that the maximum standard deviation of the output values is achieved at the two center-most input codes; the value is $0.5\sqrt{2^N\sigma_{lsb}}$ (half of the maximum value due to fitting a line between the endpoints). This statement is verified with simulations. The result is approximately the same as the results obtained with Eq. (6.14) and Eq. (6.22).

In [132], it is assumed that, if an *INL* error occurs when passing through all the possible input codes, there is a 50% chance that the error still occurs for fictive code 2^N. resulting in

$$Yield_{inl} = \left(P\left(Y\left(2^N\right) < 0.5\right) - 0.75\right) \times 4 \times 100\%, \tag{6.34}$$

where $Y\left(2^N\right)$ is normally distributed error at the fictive input code 2^N, with variance $2^N\sigma_{lsb}^2$, and $P\left(Y\left(2^N\right) < 0.5\right)$ can be obtained from the CDF of the normal distribution. This model does not take into account the fitting of the output between the code values 0 and $2^N - 1$. Also with probability values $P\left(Y\left(2^N\right) < 0.5\right) < 0.75$ the $Yield_{inl}$ becomes negative, which is not possible in any case, since the yield is inherently positive for all finite error variance values. However, according to simulation results presented in [132],it seems to model the statistical behavior of the sum of error sources very accurately in the certain parameter range.

The model presented in [132] gives a quite accurate estimate of the statistical behavior of *INL*+gain error, and therefore it is worth of trying to further improve it to model the actual *INL* yield with line fitted between the endpoints. Assuming that if there is a *INL* error somewhere at the code positions $k = 0...2^{N-1} - 2$, there is a probability of 0.5 that it still exists at the fictive code position $k = \frac{2^N-1}{2}$. If it is assumed that the probability of an *INL* error existing at some code position $k = 2^{N-1} - 2...2^N - 1$ is equal to the probability of an *INL* error existing in the code range $k = 0...2^{N-1} - 2$, and that the *INL* error values of these two halves are uncorrelated (which is not true),

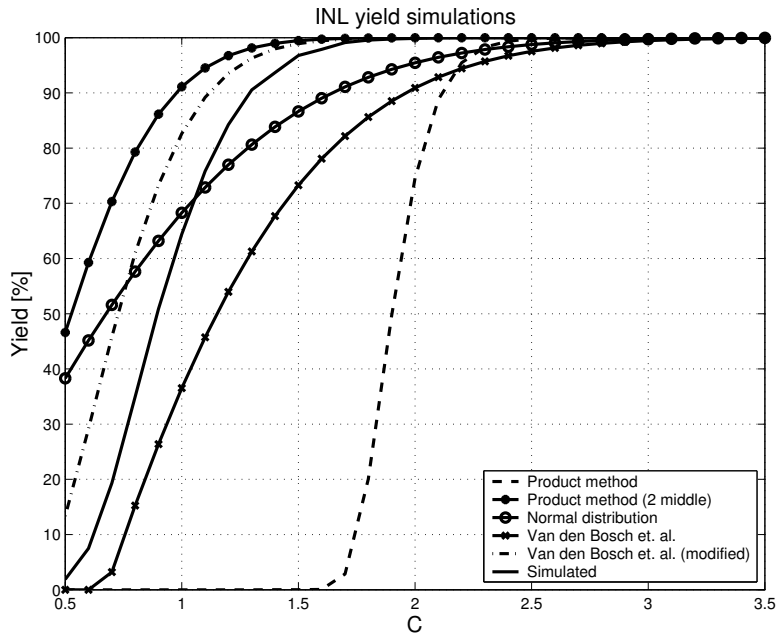

Figure 6.22 *INL* yield as a function of standard deviation divider C.

the following equation for *INL* yield can be written

$$Yield_{inl2} = \left(\left(P\left(Y_2 \left(\frac{2^N - 1}{2} \right) < 0.5 \right) - 0.75 \right) \times 4 \right)^2 \times 100\%, \tag{6.35}$$

where $Y_2\left(\frac{2^N-1}{2}\right)$ is a normally distributed random variable with variance $\sigma^2_{y_2} = \frac{2^N-1}{4}\sigma^2_{lsb}$, which is obtained from Eq. (6.14).

As a reference, it is also tested how an *INL* yield follows the normal distribution with variance $\sigma^2_{inl} = (2^N - 1)\sigma^2_{lsb}$. This equals the assumption that the maximum *INL* is the sum of the error currents.

In order to find out, which of these models gives the most reliable results for the D/A converter design, they are compared to the results obtained with Monte-Carlo simulations. In Fig. 6.22 the simulated yield and different estimation methods are compared. The simulated converter is a 14-bit segmented one with 10 binary weighted bits. One thousand converters are used in Monte-Carlo simulations. The relative standard deviation of a unit current source is assumed to be $\sigma_{lsb} = \frac{1}{2C\sqrt{2^N-1}}$.

The *INL* yield obtained with various methods as the function of unit current source standard deviation is presented in Fig. 6.23. In order to demonstrate the shortcoming of the modification of the Van den Bosch method, the simulations were also carried out for the 5 and 6-bit thermometer-coded D/A converters (Fig. 6.24 and Fig. 6.25).

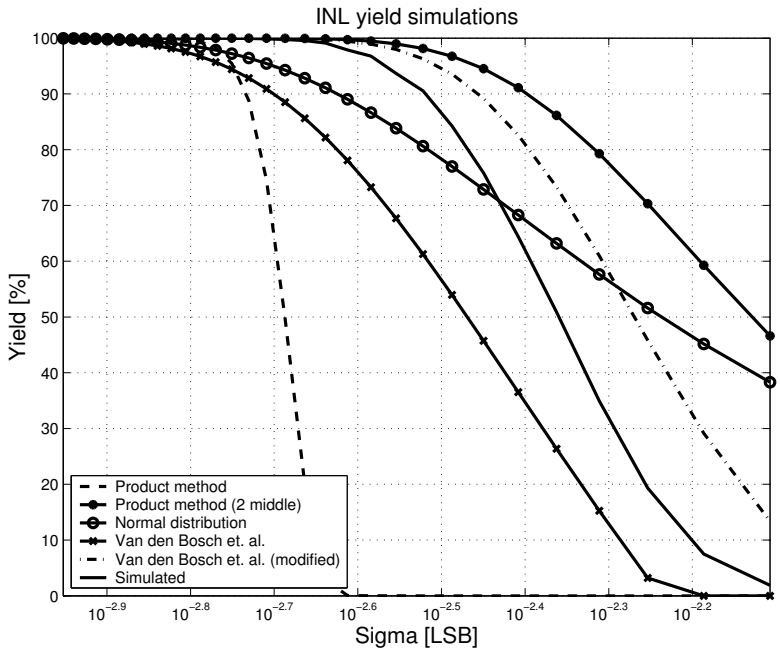

Figure 6.23 *INL* yield as a function of standard deviation of the unit current source.

It can be observed that, when a 14-bit segmented case, the results obtained from the proposed model are somewhat optimistic, in the 5-bit case, results are (almost) identical compared to Monte-Carlo simulations, and in the 6-bit case, the results are just slightly optimistic. So, it can be seen that the accuracy of the model depends on the number of bits of the converter.

Based on the simulation results, it is justified to say that none of presented *INL* yield approximations gives an accurate basis for the design of the D/A converter. For accurate information, the designer is still dependent on Monte-Carlo simulations. With the assumption of normally distributed *INL* with the standard deviation $\sigma_{inl} = \sqrt{2^N - 1}\sigma_{lsb}$, pessimistic yield estimations are obtained in the practical yield range. It can be used as a "rule of thumb"-model: "The Rule of three sigmas" results in high-enough yield.

6.3.4 Regression model for the INL and DNL yields

Due to difficulties in the evaluation of the cumulative distribution function of the *INL* yield, it is mandatory to use some kind of approximative approach if an accurate estimation of the yield is required. Instead of the sophisticated variations of guessing discussed in the previous section, one can always rely on numerical methods.

In [130], it was argued that, for certain *INL* and *DNL* yields, the required relative

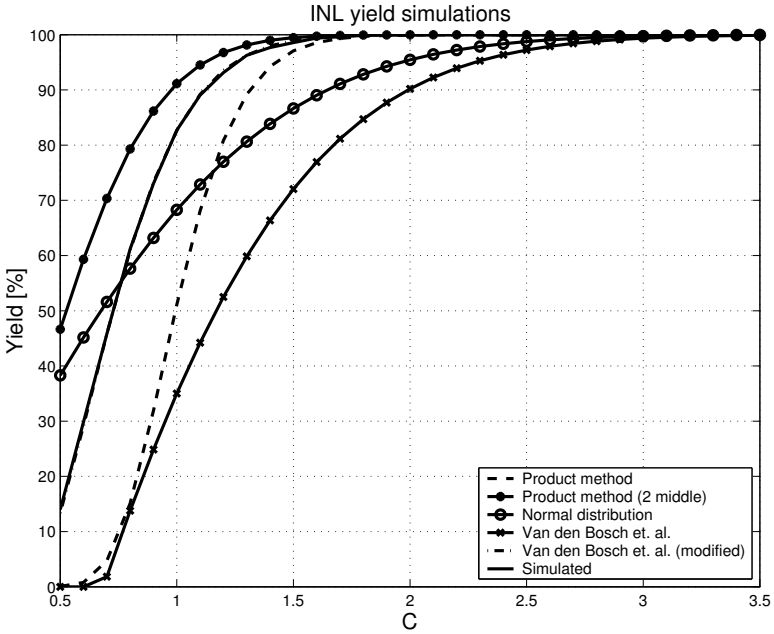

Figure 6.24 Modification of the Van den Bosch method in 5 bit case.

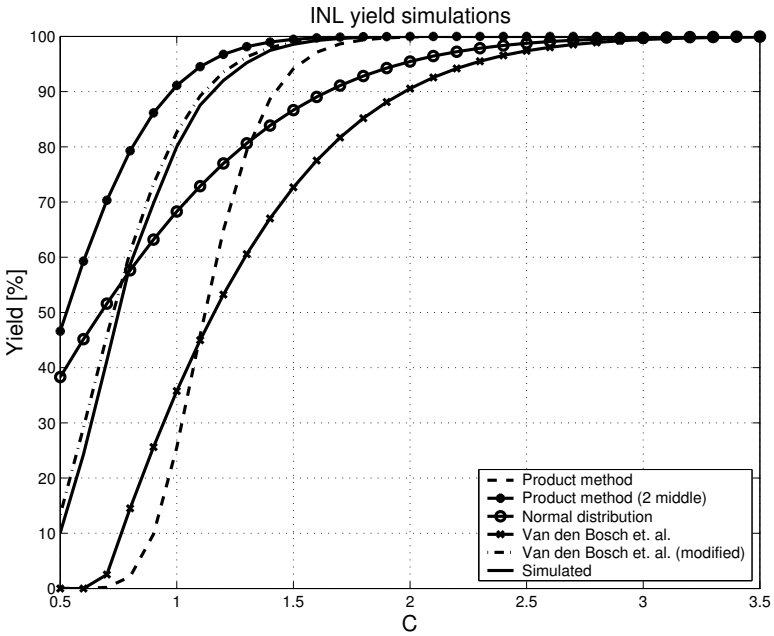

Figure 6.25 Modification of the Van den Bosch method in 6 bit case.

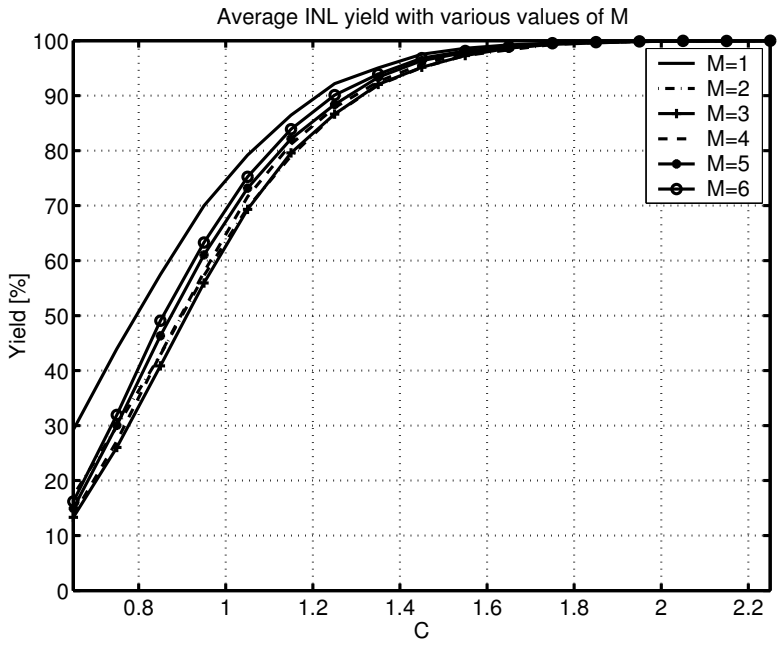

Figure 6.26 Average *INL* yield as a function of *C* for various number of segmented bits (*M*).

standard deviation of the unit current source can be written as

$$\sigma_{lsb} \approx \frac{A}{\sqrt{2^N}} Z(Yield), \tag{6.36}$$

where A is the range limit to where *INL* and *DNL* values should be, and $Z(Yield)$ is a weighting function that is independent of the number of bits (N). Because of this, the required σ_{lsb} can be discovered easily with, for example, tabulated values of $Z(Yield)$. It was demonstrated that the binary-weighted and the thermometer-coded D/A converters have different $Z(Yield)$, and that the binary-weighted D/A converters have worse differential nonlinearity (*DNL*) performance.

INL yield is dependent on the segmentation of the converter as can be seen from Fig. 6.26. C in the figure is the divider of the relative standard deviation of the LSB $\sigma_{lsb} = \frac{1}{2C\sqrt{2^N-1}}$. This is obtained from Eq. (6.36) by substitution $A = \frac{1}{2}$ and $Z(Yield) = \frac{1}{C}$. 2^N in Eq. (6.36) is substituted by $2^N - 1$ in order to define the required σ_{lsb} to be dependent on the total number of current sources in the converter. By defining σ_{lsb} this way, the effect of the number of bits on the yield is small (also in [130]) (absolute error to average yield, is $\pm 4\%$, Fig. 6.28 and Fig. 6.29) on the bit range from 10 to 16, as explained later. Therefore the average of the yields obtained with a different number of bits can be used in Fig. 6.26 and Fig. 6.27.

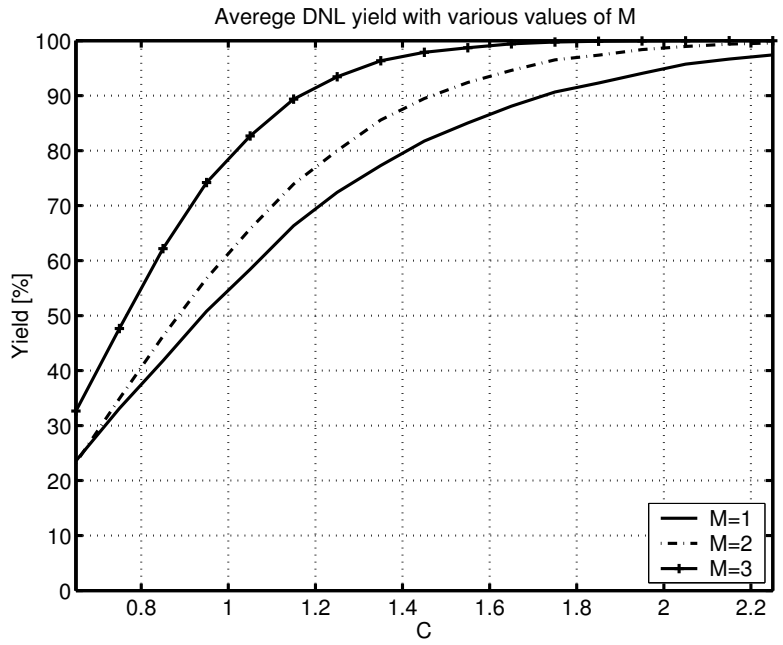

Figure 6.27 Average *DNL* yield as a function of C with various number of segmented bits (M).

INL yield is also a function of the number of segmented bits, as shown in, for example, Fig. 6.26, and the model can be developed in such a way that it also includes the dependency on the segmentation level. The yield as a function of C and segmentation could be tabulated, or curves could be given as in [130]. The solution proposed in [14] is a regression model based on the results of Monte-Carlo simulations. The number of *INL* curves simulated per each N, M, C combination is 2500. It is emphasized, that *the regression model is applicable only within the variable ranges that it was developed in*.

As a basis of the model, the following assumptions are made. The range of bits is 10-16, which is applicable for most of the design cases at the moment. The number of thermometer-coded bits is selected to vary from 1-6.

It is also assumed, that *INL* and *DNL* yields are dependent on the number of thermometer-coded MSB bits because of their larger weight. This assumption is validated by simulations by using the variable ranges under consideration; however, it does not hold when the total number of bits is small, indicating that the dependency of the required σ_{lsb} on N can not be modeled with Eq. (6.36) for all N.

With the help of Eq. (6.36), and by taking into account the number of segmented

Table 6.1 g-coefficients of $X(C,M)$ of the INL model

	1	M^1	M^2	M^3
1	-0.29711	1.75315	-0.55515	0.04734
C^{-1}	1.75115	-11.75819	3.72128	-0.31774
C^{-2}	-4.03389	29.83137	-9.46103	0.81011
C^{-3}	4.49109	-34.85655	11.14615	-0.96023
C^{-4}	-2.25836	18.06717	-5.89139	0.51356
C^{-5}	0.37269	-3.45830	1.15067	-0.10156

bits also, INL yield can be written as

$$Yield = \Phi(C,M), \tag{6.37}$$

where C is related to σ_{lsb} as $\sigma_{lsb} = \frac{1}{2C\sqrt{2^N-1}}$ and M is the number of segmented bits. The range of C is from 0.65 to 2.25. $\Phi(C,M)$ can be found by fitting some proper function to the average of the data obtained by Monte-Carlo simulations. The chosen function is

$$\begin{aligned} \Phi(C,M) &= e^{X(C,M)} \\ X(C,M) &= g_0 + \tfrac{g_1}{C} ... + \tfrac{g_i}{C^i} ... + g_{i+1}M ... + \tfrac{g_{(i\times l)}M^l}{C^i} \ , \end{aligned} \tag{6.38}$$

where i and l are the highest powers of C and M, respectively. The fitting is performed by minimizing the sum of squared errors between the Monte-Carlo simulation results and the model, resulting in

$$\sum \varepsilon_f^2 = (L_{Y_{C,M}} - XG)^T (L_{Y_{C,M}} - XG), \tag{6.39}$$

where $L_{Y_{C,M}} = \ln(AVG_N(Y_{N,C,M}))$ is a vector containing the natural logarithm of the average of the simulated yield values over the range of bits (N) from 10-16, corresponding to a certain C and M. X is a matrix with rows containing the elements of $X(C,M)$ corresponding to certain values of C and M, and $G = \left[g_0 ... g_{(i\times l)}\right]^T$. The minimum of $\sum \varepsilon_f^2$ is obtained by setting the derivative of $\sum \varepsilon_f^2$ relative to G to zero. This occurs when

$$G = (X^T X)^{-1} X^T L_{Y_{C,M}} \, . \tag{6.40}$$

In the proposed model the values of i and l are chosen to be 5 and 3, respectively, in order to achieve reasonable accuracy. The terms of $X(C,M)$ and corresponding coefficients are presented in Table 6.1.

In Fig. 6.28 the error between the Monte-Carlo simulation results and the proposed model is presented in the case $M = 4$. Error bounds between the simulated INL yields

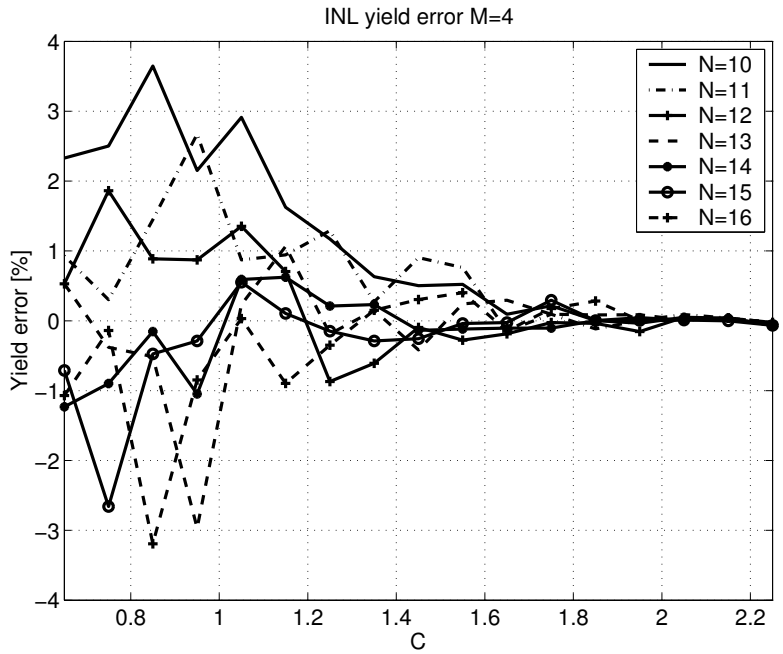

Figure 6.28 Error between simulated *INL* yields and values given by the model as a function of C. $M = 4$.

and the model are presented in Fig. 6.29.

The model for *DNL* yield is obtained with a similar method. The coefficients g for the *DNL* yield model are presented in Table 6.2.

The error between the simulated *DNL* yields and the values given by the model is presented in Fig. 6.30.

Table 6.2 g-coefficients of *DNL* model

	1	M^1	M^2
1	2.52953	-2.64437	0.64198
C^{-1}	-14.14603	14.25724	-3.47087
C^{-2}	30.28905	-27.82176	6.63958
C^{-3}	-32.86155	25.81305	-5.73440
C^{-4}	16.94454	-11.77177	2.35736
C^{-5}	-3.34097	2.08428	-0.36716

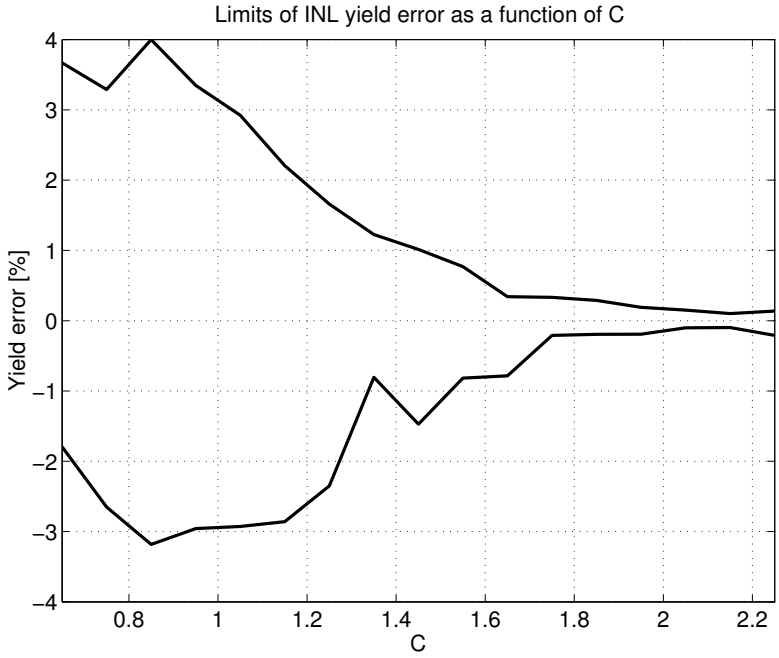

Figure 6.29 Error bounds between simulated *INL* yields and values given by the model as a function of *C*.

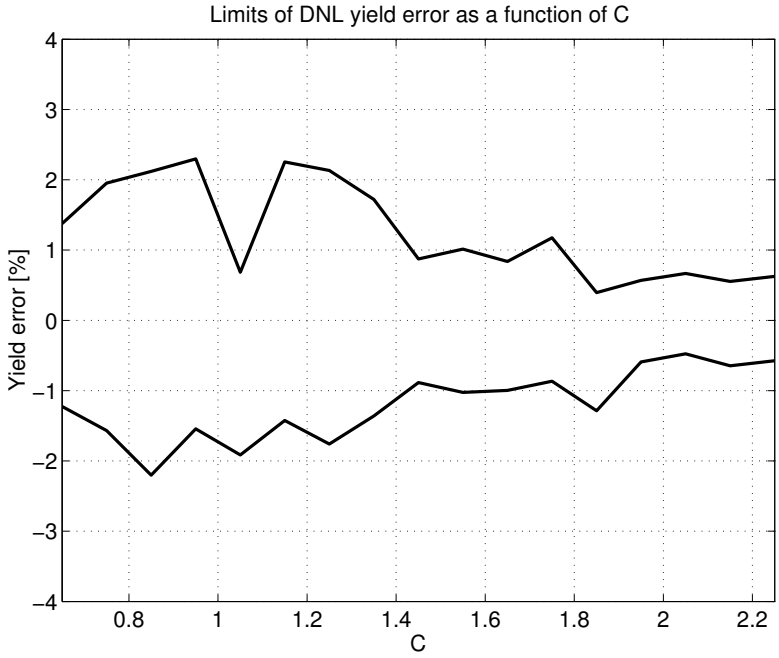

Figure 6.30 Error bounds between simulated *DNL* yields and values given by the model as a function of *C*.

6.4 Calibration techniques

The static linearity of the current-steering D/A converter is dependent on the dimensions of the current sources and their placement, as stated in Section 6.3. Various layout techniques have been developed in order to reduce the effects of the gradients on the silicon die and various algorithms have been developed in order to dimension the current sources to achieve adequate static matching. However, the fact is that static accuracy above 12 bits is hardly achievable with present silicon technologies, no matter how good the gradient cancellation might be, since the variances of *INL* and *DNL* are inversely proportional to area of the current source Eq. (6.5). There is no way to improve the matching without increasing the area, and the area required becomes quite large when linearity above 12 bit is targeted.

Because there is no practical way to measure the *INL* of the converter on-chip, the calibration algorithms usually calibrate the *DNL* of the converter; this also has an effect on *INL*.

Since in any practical case of current-steering D/A converter design the structure of the converter is either binary weighted or segmented, the *DNL* of the converter is determined by the currents with the largest weight as

$$\sigma_{dnl}^2 \approx B\sigma_{lsb}^2, \tag{6.41}$$

in which B is the number of LSB currents switched during the code transition (assuming no calibration is used). *DNL* can be reduced by reducing B by increasing the number of thermometer-coded bits. It is also possible to use multiple thermometer-coded segments, which, in practice, divides the large *DNL* values of the binary weighted part into multiple smaller *DNL*s, but does not affect *DNL* values at the transitions of the current sources with largest weight. In addition, *DNL* can be reduced by minimizing the error of the current sources of large weight by using calibration. The calibration techniques usually concentrate on tuning the erroneous current values.

Several calibration techniques have been developed to reduce the amount of error of the MSB current sources. Two main categories of calibration can be identified: continuous and quantized.

An example of continuous calibration is presented in [134]. It is continuous because the tuning process is purely analog, and no quantization of the error occurs in any phase of the calibration cycle, thereby enabling the tuning of the current to be achieved theoretically with infinite accuracy. The principle of the tuning is presented in Fig. 6.31. It can be observed, that the tuning of the current is performed by storing the V_{gs} of the diode connected transistor into the capacitance between the gate and source of the current source transistor. By using the same reference source for all of

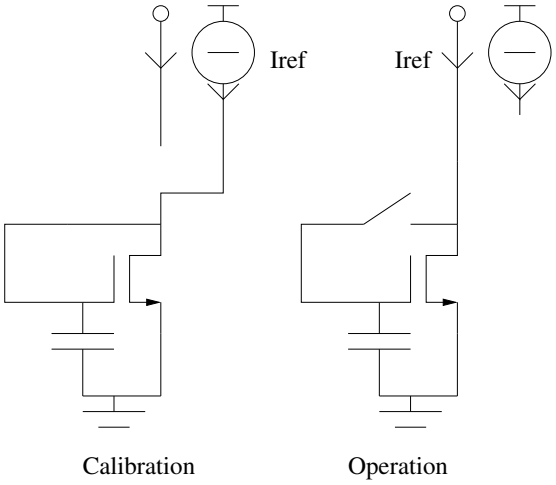

Figure 6.31 Principle of continuous tuning of an MSB current source.

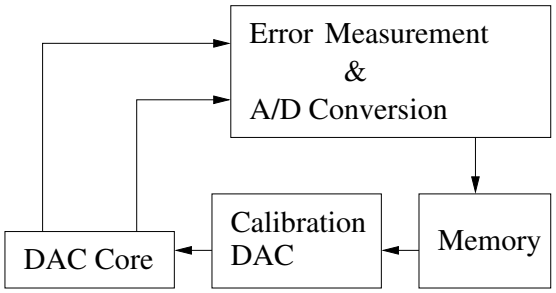

Figure 6.32 Principle of quantizing calibration of current-steering D/A converter.

the MSB current sources, it is possible to equalize the currents of the MSB sources. However, because the switches are nonideal, the calibration suffers from charge injection from the switches and leakage currents, which is the main drawback of this method. Because of leakage current the calibration of the source has to be repeated continuously in order to maintain the accuracy. This problem can be avoided by using digital memory to store the error value.

The most common calibration technique is to use digital storage elements to store the error values of the MSB current sources and use these values to tune the currents. The principle of this calibration method is presented in Fig. 6.32. This approach is used in, for example, [135], [136], [137], [138] and [139]. The common factor for these methods is the quantization of the measured error and the usage of additional digital-to-analog converters to tune the values of the MSB current sources. These additional digital to analog converters are usually called ”Calibration DACs”; there can be either one large calibration DAC, as in [138] and [136], or multiple smaller calibration DACs for each of the MSB sources, as in [135], [137] and [139].

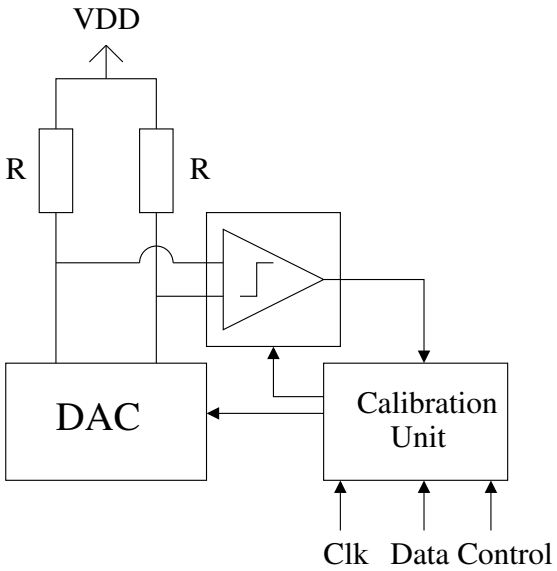

Figure 6.33 Block diagram of the D/A converter with calibration using digital predistortion.

The methods differ from each other mainly in ways of performing the measurement of the error and actual way of tuning the current, but the principle is the same; error is measured, A/D converted, stored digitally and MSB currents are tuned by D/A conversion of the stored error. The term quantized is therefore more valid in this context than the term ”digital”, since the only digital part of these calibration methods is the storage element; the tuning is made by analog means.

Also, the main drawback of these methods is related to quantization. Whereas the continuous method was purely analog and therefore accurate but sensitive to nonidealities, the quantized method is inaccurate due to quantization noise, but insensitive to interference due to the digital storage element. The quality of the calibration is determined by the accuracy of the measurement, quality of the A/D conversion, and resolution of the calibration DAC.

A slightly different technique is suggested by the author and presented in [14]. The main difference between this method and previously introduced methods is that there is no particular calibration DAC. Instead, redundancy is added to the converter by biasing the MSB current sources so that the current of a single MSB source is less than the sum of the LSB sources. Because of the redundancy, the input code space and output range of the converter are extended with four additional MSB current sources, enabling the correction of the errors of the MSB sources by digital predistortion of the input code. The principle of the proposed calibration method is presented in Fig. 6.34 and Fig. 6.34.

The calibration is performed as follows: the MSB sources, including the additional

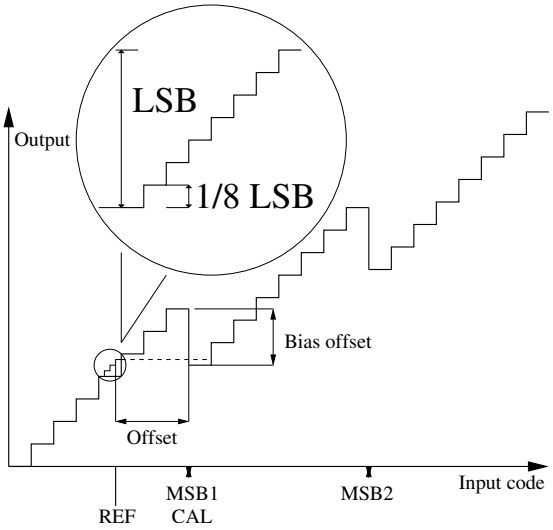

Figure 6.34 Principle of the calibration algorithm using digital predistortion.

sources, are biased so that the current of the MSB source is less than the sum of the LSB sources. This ensures, that the output value of the converter *decreases* as a new MSB current source is added to the output. Then the REF-value in Fig. 6.34 is set to CAL-1. While calibrating, the REF values are first decreased in 4 LSB steps. When the output corresponding to the REF is less than that corresponding to the CAL, the REF value is increased by steps of 1/8 LSB as long as the output corresponding to REF is again greater than the output corresponding to the CAL-value. This is simply a successive approximation register (SAR) A/D conversion of the error value. The difference between the final CAL and REF values is then stored to memory as an offset value. The CAL-value is then increased in order to add the next MSB current to output, and the SAR-cycle is repeated. After all of the offsets are measured and stored, the calibration cycle is completed and the offsets can be used in normal operation mode to predistort the input code in order to linearize the transfer function of the converter.

The functionality of the calibration algorithm was verified with simulations. Figs. 6.35 - 6.39 represent the minmax envelope curves for *DNL* and *INL* uncalibrated and calibrated 16-bit D/A converter with 6 thermometer coded MSB bits under various matching conditions for the LSB and MSB current sources. *DNL* and *INL* yield curves for various matching conditions are presented in Fig. 6.40. The number of converters in each of the yield simulations is 1000.

In each of the presented cases *DNL* and *INL* yields for uncalibrated converters is 0%. In the curves for the uncalibrated converters, the effect of the bias offset on the yield is removed, so that the presented curves indicate directly the effect of the matching on the yield.

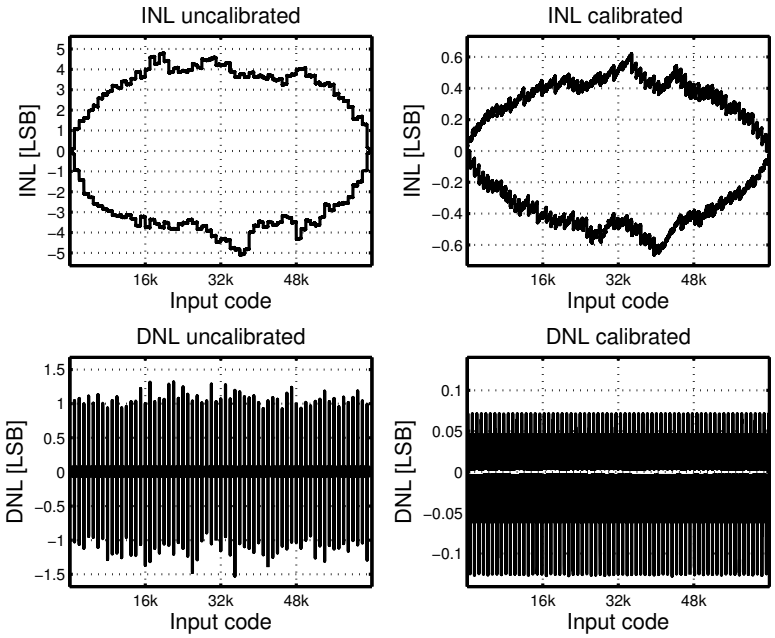

Figure 6.35 Result of calibration, $\sigma_{LSB} = \frac{1}{2\times3\sqrt{2^{16}-1}} I_{LSB}$, $\sigma_{MSB} = \frac{16\sqrt{2^{10}}}{2\times3\sqrt{2^{16}-1}} I_{LSB}$.

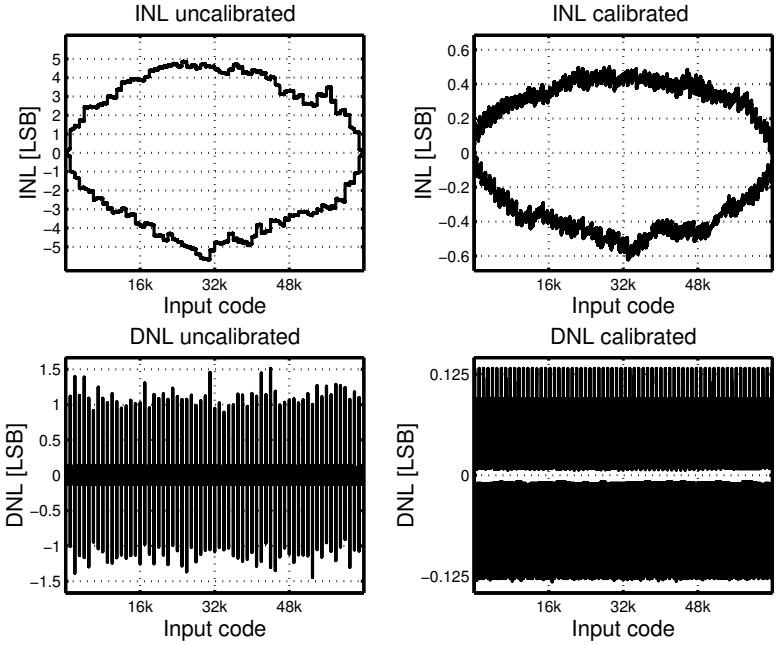

Figure 6.36 Result of calibration, $\sigma_{LSB} = \frac{2}{2\times3\sqrt{2^{16}-1}} I_{LSB}$, $\sigma_{MSB} = \frac{16\sqrt{2^{10}}}{2\times3\sqrt{2^{16}-1}} I_{LSB}$.

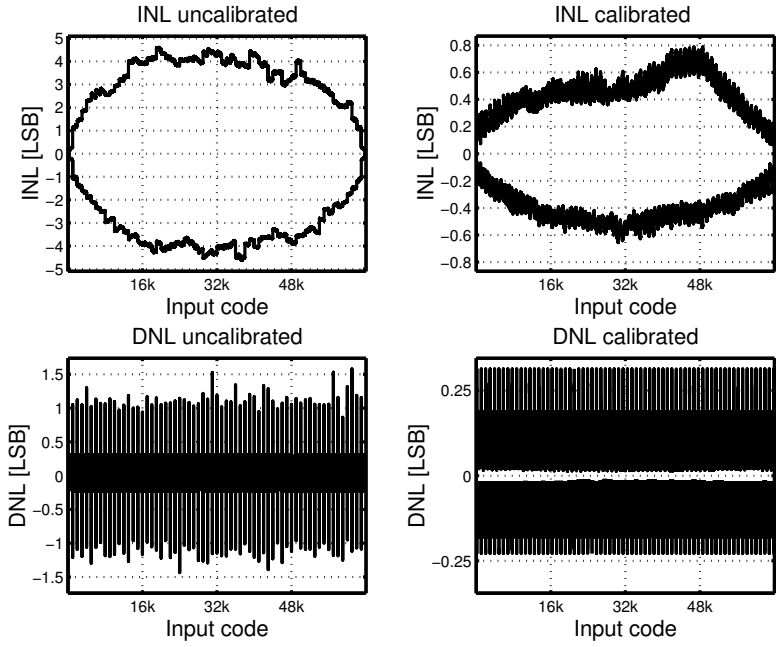

Figure 6.37 Result of calibration, $\sigma_{LSB} = \frac{4}{2\times3\sqrt{2^{16}-1}} I_{LSB}$, $\sigma_{MSB} = \frac{16\sqrt{2^{10}}}{2\times3\sqrt{2^{16}-1}} I_{LSB}$.

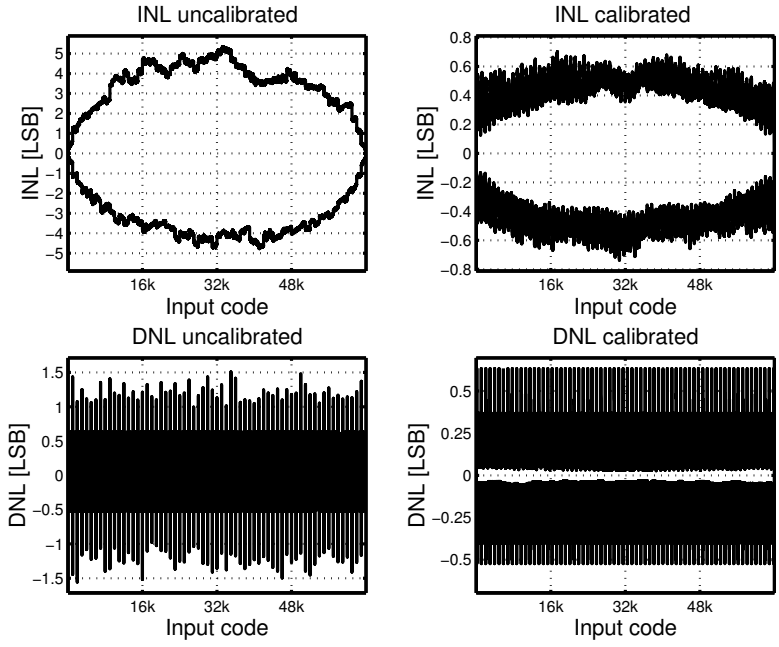

Figure 6.38 Result of calibration, $\sigma_{LSB} = \frac{8}{2\times3\sqrt{2^{16}-1}} I_{LSB}$, $\sigma_{MSB} = \frac{16\sqrt{2^{10}}}{2\times3\sqrt{2^{16}-1}} I_{LSB}$.

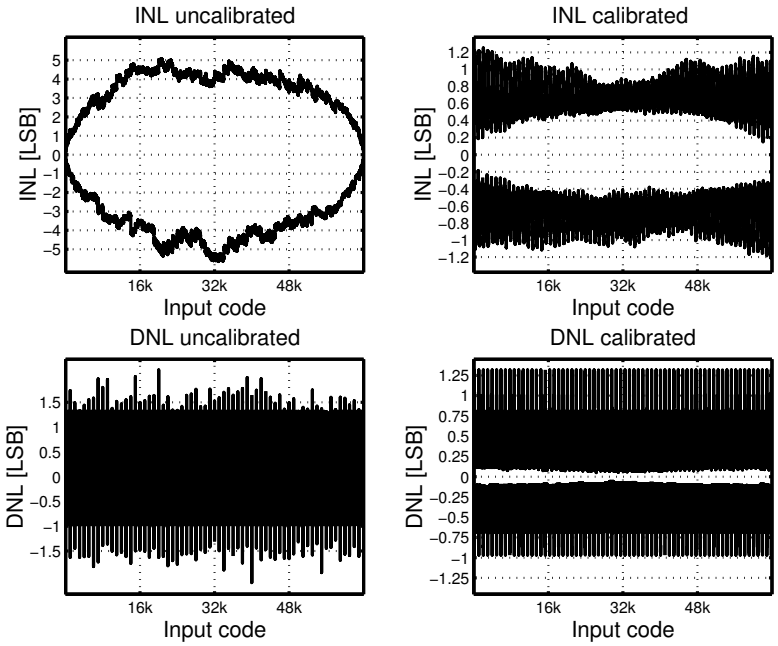

Figure 6.39 Result of calibration, $\sigma_{LSB} = \frac{16}{2\times3\sqrt{2^{16}-1}} I_{LSB}$, $\sigma_{MSB} = \frac{16\sqrt{2^{10}}}{2\times3\sqrt{2^{16}-1}} I_{LSB}$.

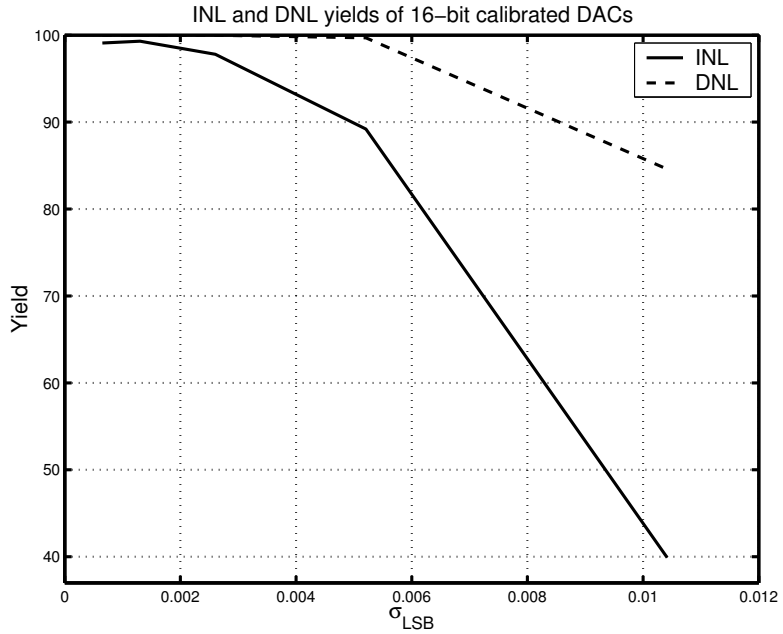

Figure 6.40 *DNL* and *INL* yields of a calibrated 16-b D/A converter as a function of σ_{LSB}.

Simulation results indicate that *DNL* and *INL* yields are strongly dependent on the matching of the LSB current sources which determines the achievable *DNL* and *INL* yield. It can be seen that, when the matching of the LSB sources is deteriorated, it begins to dominate the *DNL*, and thus the *INL* yield. It is also demonstrated that acceptable yield levels can be achieved with very poor MSB matching, thus making it possible to effectively reduce the area required for the MSB sources.

The amount of the error that can be calibrated with the algorithm is determined by the amount of additional MSB current sources. Assuming zero mean error in the sources, the maximum error that can be calibrated equals the bias offset of the MSB current sources, whereas the sum of the bias errors cannot exceed the additional code range obtained by adding the sources. In the presented case, the code range added is 4096 LSBs and the number of MSB sources is 67, resulting in a maximum correctable error of 61 LSBs.

6.5 Effects of output impedance variation

The finite impedance of the current sources causes code-dependent variation on the output impedance of the current-steering D/A converter, introducing nonlinearity to the output signal. In this section, Taylor-series approximations of the *INL* of a single-ended and differential converter are presented in order to give equations of the nonlinearities, which are then used to compute the analytical result of maximum *INL* and $SFDR$ as a function of output impedance.

Nonlinearities due to impedance variation have been previously discussed in [140], [141], [142], and [143]. The second-order nonlinearities due to impedance variation are discussed in [140], [141] and [142], third-order distortion is analyzed in [143], and the frequency dependency of the output impedance is discussed in [142].

Results presented in this section equal the results presented in [142] for a single-ended case; however, results for the $SFDR$ of differential converter differ from the results presented in [143].

The frequency dependency of the output impedance is analyzed mathematically in detail and the results are verified with simulations. Results differ slightly when compared to [142]. The limits of moving the poles and zeros of output impedance in frequency are determined.

The analysis of the output impedance analysis is linked to implementation by analyzing the effect of transistor dimensions on the frequency response of the output impedance, demonstrating the effect of various design parameters, and the boundaries of the output impedance optimization.

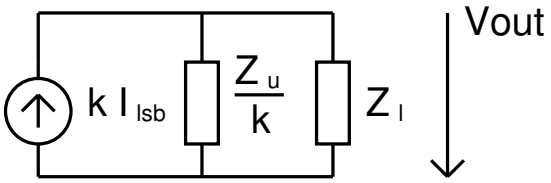

Figure 6.41 Output impedance variation of the single-ended D/A converter.

6.5.1 Distortion due to low frequency impedance variation

Let a single current source branch of a fully thermometer coded D/A converter have the impedance Z_c when the switch is conducting, and impedance Z_{op} when it is not. In this case, the change in the impedance due to switching can be modeled as an impedance in parallel with the impedance Z_u of the open (not conducting) switch. Z_u can be computed as

$$Z_u = \frac{Z_c Z_{op}}{Z_{op} - Z_c}. \tag{6.42}$$

Since Z_{op} is usually very large at low frequencies, the low frequency analysis can be performed by observing the value of the current branch in the conducting state; however, in high-frequency analysis it is important to isolate Z_{op}, since it is constant and thus does not introduce any distortion. However, Z_{op} may introduce some undesired phenomena such as frequency-dependent attenuation and, in that sense, it should be taken into account. Frequency dependency is discussed in more detail in Section 6.5.2.

The low-frequency effects of the impedance variation can be analyzed as follows. In Fig. 6.41, the simplified circuit for a single-ended D/A converter is presented. The single ended converter has the output voltage

$$V_{out}(k) = \frac{kI_{lsb}Z_l}{1 + \frac{kZ_l}{Z_u}}, \tag{6.43}$$

in which Z_l is the load impedance and Z_u is the impedance of the unit current source. *INL* caused by the finite output impedance of the current source can be written as

$$INL_{zse}(k) = \frac{\frac{kZ_l}{1+\frac{kZ_l}{Z_u}} - \frac{kZ_l}{1+\frac{Z_l\left(2^N-1\right)}{Z_u}}}{\frac{Z_l}{1+\frac{Z_l\left(2^N-1\right)}{Z_u}}}. \tag{6.44}$$

By performing third-order Taylor-series expansion for INL_{zse} with respect to k, $k \approx$

$\frac{2^N-1}{2}$, Eq. (6.44) can be written as

$$INL_{zse}(k) \approx \frac{\frac{Z_l}{Z_u}}{\left(2+(2^N-1)\frac{Z_l}{Z_u}\right)^3}\left(-8\left(1+\left(2^N-1\right)\frac{Z_l}{Z_u}\right)k^2 + \left(8\left(2^N-1\right)+6\left(2^N-1\right)^2\frac{Z_l}{Z_u}-\left(2^N-1\right)^3\frac{Z_l^2}{Z_u^2}\right)k + \left(2^N-1\right)^3\frac{Z_l}{Z_u}+\left(2^N-1\right)^4\frac{Z_l^2}{Z_u^2}\right). \quad (6.45)$$

If k is considered as a real number instead of an integer, the maximum value of INL due to finite impedance of the current source is

$$INL_{zsemax} \approx \frac{1}{32}\frac{Z_l}{Z_u}\frac{\left(2^N-1\right)^2\left(16+16\left(2^N-1\right)\frac{Z_l}{Z_u}+\left(2^N-1\right)^2\frac{Z_l^2}{Z_u^2}\right)}{(2+(2^N-1)\frac{Z_l}{Z_u})(1+(2^N-1)\frac{Z_l}{Z_u})} \quad (6.46)$$

at

$$k_{max} \approx \frac{\left(2^N-1\right)}{2}-\frac{1}{16}\frac{Z_l}{Z_u}\frac{\left(2^N-1\right)^2\left(2+\left(2^N-1\right)\frac{Z_l}{Z_u}\right)}{\left(1+(2^N-1)\frac{Z_l}{Z_u}\right)}. \quad (6.47)$$

For differential output, fourth-order Taylor-series approximation of INL with respect to k, $k \approx \frac{2^N-1}{2}$ can be written as

$$INL_{zdi}(k) \approx \frac{1}{2}\frac{Z_l^2}{Z_u^2}\frac{\left(\left(2^N-1\right)-2k\right)}{\left(2+(2^N-1)\frac{Z_l}{Z_u}\right)^4}\left(-16\left(1+\left(2^N-1\right)\frac{Z_l}{Z_u}\right)k^2 \quad (6.48)$$

$$+16\left(2^N-1\right)\left(1+\left(2^N-1\right)\frac{Z_l}{Z_u}\right)k+\frac{Z_l^2}{Z_u^2}\left(2^N-1\right)^4\Bigg) \quad (6.49)$$

with the maximum absolute value

$$INL_{zdimax} \approx \frac{Z_l^2}{Z_u^2}\frac{\left(2^N-1\right)^3}{6\sqrt{3}\sqrt{1+\frac{z_l}{Z_u}\left(2^N-1\right)}\left(2+\frac{z_l}{Z_u}\left(2^N-1\right)\right)} \quad (6.50)$$

at

$$k_{max} \approx \frac{\left(2^N-1\right)}{2}\pm\frac{\left(2^N-1\right)\left(2+\left(2^N-1\right)\frac{Z_l}{Z_u}\right)}{4\sqrt{3}\sqrt{\left(1+(2^N-1)\frac{Z_l}{Z_u}\right)}} \quad (6.51)$$

It can be observed that INL_{zsemax} is linearly dependent on $\frac{Z_l}{Z_u}$ whereas INL_{zdimax} is

dependent on $\frac{Z_l^2}{Z_u^2}$. It can therefore be assumed that the even-order distortion generated in a single-ended converter is much greater that the odd-order harmonics in a differential case. However, in real design, it is impossible to cancel all even-order distortion with differential structure, although the second-order distortion is usually attenuated about 20 dB.

INL equations, Eq. (6.45) and Eq. (6.49), include all nonlinearities due to output impedance variation normalized to LSB. Thus the harmonic distortion can be discovered by applying sinusoidal input $k = \frac{\left(2^N-1\right)(1+\sin(\omega t))}{2}$ Eq. (6.45) and Eq. (6.49), from which the amplitudes of the second and the third harmonic components can be determined.

In the single-ended case, the spurious free dynamic range (*SFDR*) is determined by the second harmonic component. *SFDR* computed from Eq. (6.45) is approximately

$$SFDR_{zse} \approx \frac{4+2\left(2^N-1\right)\frac{Z_l}{Z_u}}{\left(2^N-1\right)\frac{Z_l}{Z_u}} \tag{6.52}$$

$$SFDR_{zse} \approx 12.04-6.02N+20log_{10}\left(\frac{Z_u}{Z_l}\right)dB, \quad \frac{Z_l}{Z_u} \ll \frac{1}{2\left(2^N-1\right)}. \tag{6.53}$$

In the differential case, *SFDR* is determined by the third harmonic, being approximately

$$SFDR_{zdi} \approx \frac{16+16\left(2^N-1\right)\frac{Z_l}{Z_u}+7\left(2^N-1\right)^2\frac{Z_l^2}{Z_u^2}}{\left(2^N-1\right)^2\frac{Z_l^2}{Z_u^2}} \tag{6.54}$$

$$SFDR_{zdi} \approx 28.08-12.04N+40log_{10}\left(\frac{Z_u}{Z_l}\right)dB, \quad \frac{Z_l}{Z_u} \ll \frac{1}{16\left(2^N-1\right)}. \tag{6.55}$$

Comparison of simulated accurate distortion values and approximations from Eqs. (6.52) and (6.54) are presented in Fig. 6.42. It can be observed that the accuracy of Taylor approximations is exacerbated with large values of $\frac{Z_l}{Z_u}$.

It was stated that the even-order harmonics cannot be canceled, only attenuated. It is therefore good to use Eq. (6.52) to find the required $\frac{Z_u}{Z_l}$ ratio in order to have good enough *SFDR*. For example, with a 14-bit converter it requires approximately $\frac{Z_u}{Z_l}$ to be greater than 72.9×10^6 $(157.25\,dB)$ in order to have *SFDR* better than 85 dB. If the load impedance $Z_l = 50\Omega$, the impedance of the unit current source has to be greater than $3.6\times10^9\Omega$.

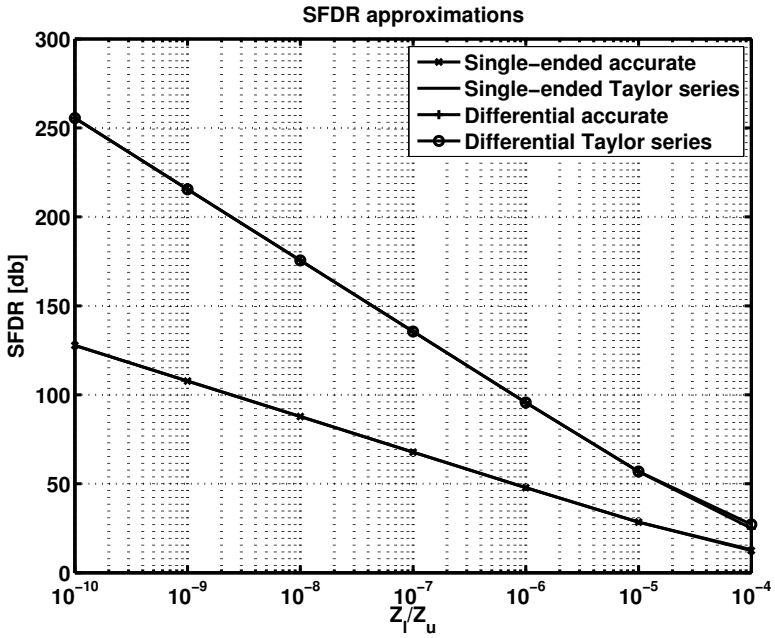

Figure 6.42 Comparison of simulated and approximated values of $SFDR$ for a single-ended and differential converter.

6.5.2 Frequency dependency of the output impedance

The frequency behavior of the output impedance also affects the distortion, since the impedance tends to decrease as the frequency increases, as presented in Fig. 6.43.

Fig. 6.44 represents a single current branch of a D/A converter with two cascode transistors above the current source transistor.

By using a small-signal equivalent circuit for a transistor and by discarding the capacitances on the drain node of M4, which are included in constant impedance Z_{op},

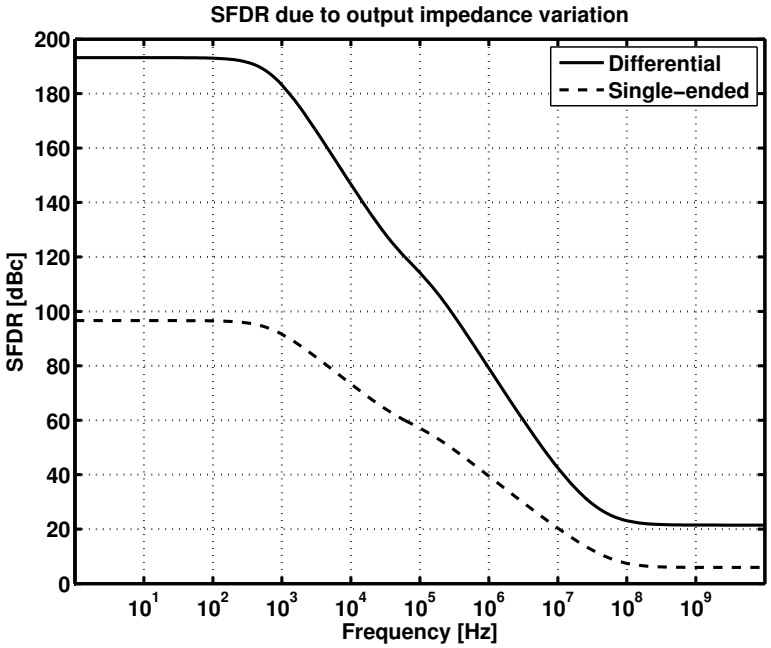

Figure 6.43 *SFDR* as a function of frequency.

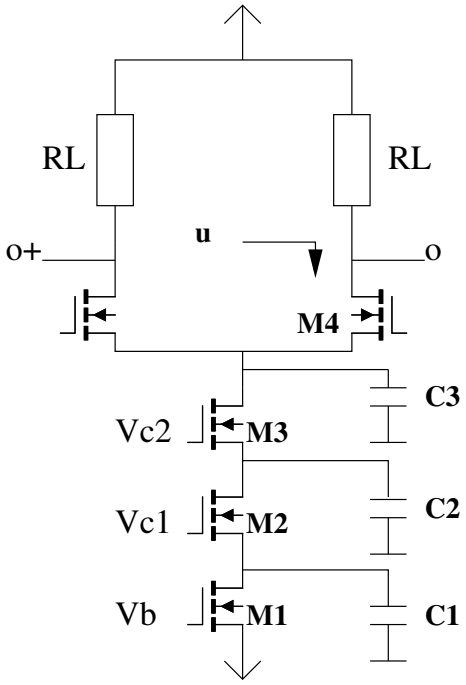

Figure 6.44 Current source with two cascode transistors and switches.

it is possible to write the equation for Z_u in S-domain as

$$Z_u(s) = \frac{N(s)}{D(s)} \tag{6.56}$$

$$\begin{aligned} N(s) = {} & s^3 C_1 C_2 C_3 \\ & + s^2 C_3 C_2 (g_{m2} + g_{ds2} + g_{ds1}) \\ & + s^2 C_3 C_1 (g_{m3} + g_{ds3} + g_{ds2}) \\ & + s^2 C_2 C_1 (g_{m4} + g_{ds4} + g_{ds3}) \\ & + s C_3 ((g_{m3} + g_{ds3})(g_{m2} + g_{ds2} + g_{ds1}) + g_{ds2} g_{ds1}) \\ & + s C_2 (g_{m2} + g_{ds2} + g_{ds1})(g_{m4} + g_{ds4} + g_{ds3}) \\ & + s C_1 ((g_{m4} + g_{ds4})(g_{m3} + g_{ds3} + g_{ds2}) + g_{ds3} g_{ds2}) \\ & + (g_{m4} + g_{ds4})(g_{m3} + g_{ds3})(g_{m2} + g_{ds2} + g_{ds1}) \\ & + (g_{m4} + g_{ds4} + g_{ds3}) g_{ds2} g_{ds1} \end{aligned} \tag{6.57}$$

$$\begin{aligned} D(s) = {} & s^3 C_1 C_2 C_3 g_{ds4} \\ & + s^2 C_3 C_2 (g_{m2} + g_{ds2} + g_{ds1}) g_{ds4} \\ & + s^2 C_3 C_1 (g_{m3} + g_{ds3} + g_{ds2}) g_{ds4} \\ & + s^2 C_2 C_1 g_{ds4} g_{ds3} \\ & + s C_3 ((g_{m3} + g_{ds3})(g_{m2} + g_{ds2} + g_{ds1}) + g_{ds2} g_{ds1}) g_{ds4} \\ & + s C_2 (g_{m2} + g_{ds2} + g_{ds1}) g_{ds3} g_{ds4} \\ & + s C_1 g_{ds4} g_{ds3} g_{ds2} \\ & + g_{ds4} g_{ds3} g_{ds2} g_{ds1}. \end{aligned} \tag{6.58}$$

Because $g_m \gg g_{ds}$, Eq. (6.56) can be approximated with

$$Z_u(s) \approx \frac{N_a(s)}{D_a(s)} \tag{6.59}$$

$$\begin{aligned} N_a(s) = {} & g_{m4} g_{m3} g_{m2} \left(s^3 \frac{C_1 C_2 C_3}{g_{m4} g_{m3} g_{m2}} + s^2 \left(\frac{C_3 C_2}{g_{m4} g_{m3}} + \frac{C_3 C_1}{g_{m4} g_{m2}} + \frac{C_2 C_1}{g_{m3} g_{m2}} \right) \right. \\ & \left. + s \left(\frac{C_3}{g_{m4}} + \frac{C_2}{g_{m3}} + \frac{C_1}{g_{m2}} \right) + 1 \right) \end{aligned} \tag{6.60}$$

$$\begin{aligned} D_a(s) = {} & g_{ds4} g_{ds3} g_{ds2} g_{ds1} \left(s^3 \frac{C_1 C_2 C_3}{g_{ds3} g_{ds2} g_{ds1}} \right. \\ & + s^2 \left(\frac{C_3 C_2 g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} + \frac{C_3 C_1 g_{m3}}{g_{ds3} g_{ds2} g_{ds1}} + \frac{C_2 C_1}{g_{ds2} g_{ds1}} \right) \\ & \left. + s \left(\frac{C_3 g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} + \frac{C_2 g_{m2}}{g_{ds2} g_{ds1}} + \frac{C_1}{g_{ds1}} \right) + 1 \right). \end{aligned} \tag{6.61}$$

It is possible to further simplify Eq. (6.61) with the following assumptions. Usually C_1 is very large compared to C_2 and C_3 because its value is determined by the routing of the current source matrix. If $\frac{C_1}{g_{ds1}} \gg \frac{C_3 g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}}$, the factorization results in

$$Z_u(s) \approx \frac{g_{m4} g_{m3} g_{m2}}{g_{ds4} g_{ds3} g_{ds2} g_{ds1}} \frac{\left(s\frac{C_1}{g_{m2}}+1\right)\left(s\frac{C_3}{g_{m4}}+1\right)}{\left(s\frac{C_1}{g_{ds1}}+1\right)\left(s\frac{C_3 g_{m3}}{g_{ds3} g_{ds2}}+1\right)}. \tag{6.62}$$

On the other hand, if $\frac{C_3 g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} \gg \frac{C_1}{g_{ds1}}$

$$Z_u(s) \approx \frac{g_{m4} g_{m3} g_{m2}}{g_{ds4} g_{ds3} g_{ds2} g_{ds1}} \frac{\left(s\frac{C_3}{g_{m4}}+1\right)}{\left(s\frac{C_3 g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}}+1\right)}. \tag{6.63}$$

Poles $p1$, $p2$, and p, and zeros $z1$ and $z2$ of Z_u in the two-cascode case are at frequencies

$$\omega_{z1} \approx \frac{g_{m2}}{C_1} \tag{6.64}$$

$$\omega_{z2} \approx \frac{g_{m4}}{C_3} \tag{6.65}$$

$$\omega_{p1} \approx \frac{g_{ds1}}{C_1} \tag{6.66}$$

$$\omega_{p2} \approx \frac{g_{ds3} g_{ds2}}{C_3 g_{m3}}, \quad \frac{C_1}{g_{ds1}} \gg \frac{C_3 g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} \tag{6.67}$$

$$\omega_{p} \approx \frac{g_{ds3} g_{ds2} g_{ds1}}{C_3 g_{m3} g_{m2}}, \quad \frac{C_3 g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} \gg \frac{C_1}{g_{ds1}} \tag{6.68}$$

Similarly, we may write the equations for the case of only one cascode and current source (Fig. 6.45) and current source only (Fig. 6.46). In the one-cascode case, the output impedance can be written as

$$Z_u(s) \approx \frac{g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} \frac{\left(s\frac{C_2}{g_{m3}}+1\right)\left(s\frac{C_1}{g_{m2}}+1\right)}{\left(s\frac{C_1}{g_{ds1}}+1\right)\left(s\frac{C_2}{g_{ds2}}+1\right)}, \qquad \frac{C_1}{g_{ds1}} \gg \frac{C_2 g_{m2}}{g_{ds2} g_{ds1}} \tag{6.69}$$

$$Z_u(s) \approx \frac{g_{m3} g_{m2}}{g_{ds3} g_{ds2} g_{ds1}} \frac{\left(s\frac{C_2}{g_{m3}}+1\right)}{\left(s\frac{C_2 g_{m2}}{g_{ds2} g_{ds1}}+1\right)}, \qquad \frac{C_2 g_{m2}}{g_{ds2} g_{ds1}} \gg \frac{C_1}{g_{ds1}}, \tag{6.70}$$

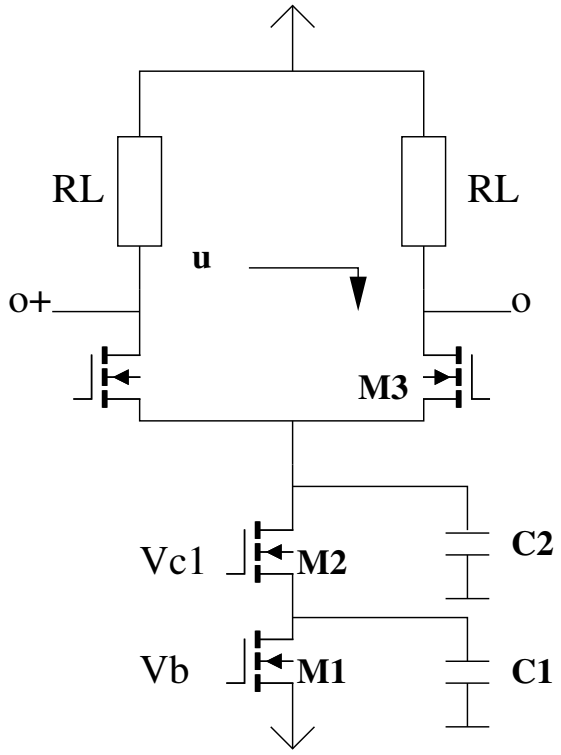

Figure 6.45 Current source with one cascode transistor.

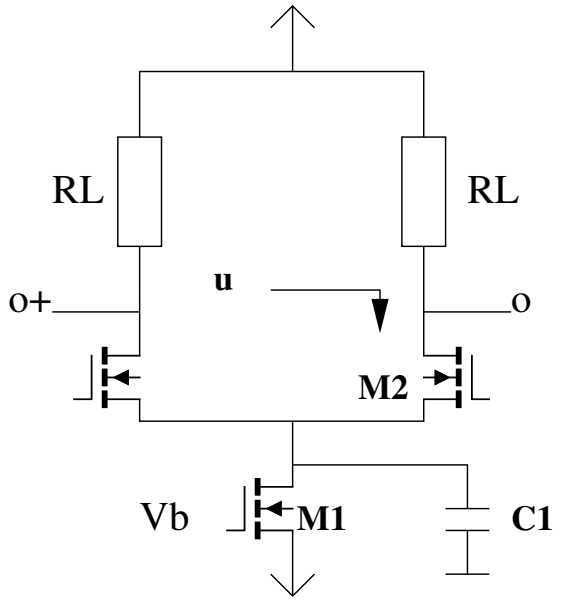

Figure 6.46 Current source without cascode transistor.

resulting in poles $p1$, $p2$, and p, and zeros $z1$ and $z2$ on frequencies

$$\omega_{z1} \approx \frac{g_{m2}}{C_1} \tag{6.71}$$

$$\omega_{z2} \approx \frac{g_{m3}}{C_2} \tag{6.72}$$

$$\omega_{p1} \approx \frac{g_{ds1}}{C_1} \tag{6.73}$$

$$\omega_{p2} \approx \frac{g_{ds2}}{C_2}, \quad \frac{C_1}{g_{ds1}} \gg \frac{C_2 g_{m2}}{g_{ds2} g_{ds1}} \tag{6.74}$$

$$\omega_{p} \approx \frac{g_{ds2} g_{ds1}}{C_2 g_{m2}}, \quad \frac{C_2 g_{m2}}{g_{ds2} g_{ds1}} \gg \frac{C_1}{g_{ds1}}. \tag{6.75}$$

Similarly for the case of current source only, the output impedance variation is

$$Z_u(s) \approx \frac{g_{m2}}{g_{ds2} g_{ds1}} \frac{\left(s\frac{C_1}{g_{m2}} + 1\right)}{\left(s\frac{C_1}{g_{ds1}} + 1\right)} \tag{6.76}$$

with pole p and zero z

$$\omega_z \approx \frac{g_{m2}}{C_1} \tag{6.77}$$

$$\omega_p \approx \frac{g_{ds1}}{C_1}. \tag{6.78}$$

As analyzed in [142], it is obvious that the best frequency performance is obtained when routing capacitances C_1, C_2, and C_3 are minimized. Next, whether it is possible to improve the frequency behavior by adjusting the dimensions (i.e. $\frac{g_m}{g_{ds}}$ ratio) of cascodes and switches will be analyzed. The assumed parameter dependencies are presented in Eqs. (6.79)-(6.83)

$$g_m \sim k_{gm}\sqrt{W/L} \tag{6.79}$$

$$g_{ds} \sim \frac{k_{gds}}{L} \tag{6.80}$$

$$C_1 \quad \text{constant} \tag{6.81}$$

$$C_2 \sim A_{cc} W_3 L_3 \tag{6.82}$$

$$C_3 \sim A_{cc} W_4 L_4. \tag{6.83}$$

Dependencies of the capacitances are made on the assumption that C_1 is determined by the large routing capacitance from current sources to cascode transistors, whereas C_2, and C_3 are determined by the channel capacitances of the transistors, and therefore they are dependent on the dimensions of the transistors. Dependencies of the g_m and g_{ds} are generally known [144].

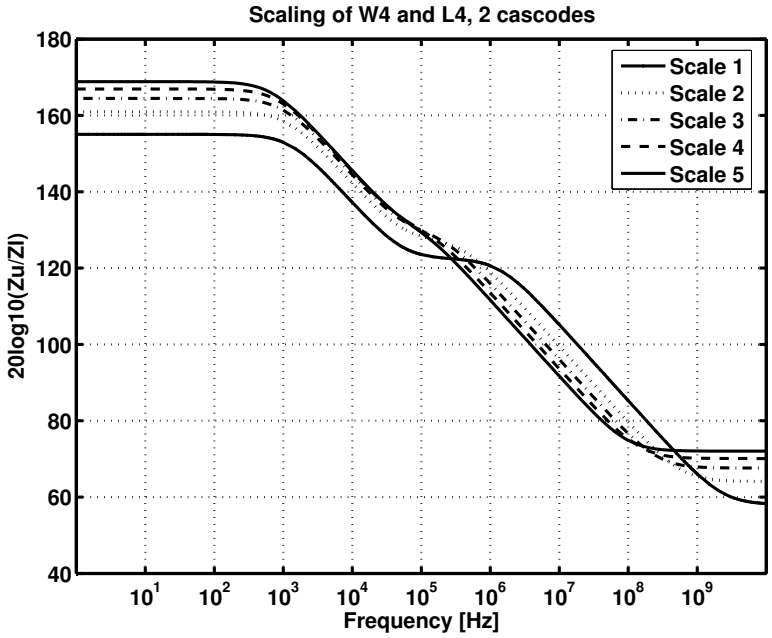

Figure 6.47 Scaling W_4 and L_4 in 2-cascode case.

Frequency dependency of the impedance Z_u can be analyzed with MATLAB. Fig. 6.47 presents the behavior of Z_u when both W_4 and L_4 are scaled by the same factor varying from 1 to 5. It can be observed that as the dimensions are increased, pole $p2$ in Eq. (6.62) is moved towards lower frequencies due to increasing C_3, until it cancels the zero z1. Further increasing the dimensions makes pole p dominant, resulting in Eq. (6.63). The zero $z2$ is moved down in frequency due to increasing C_3.

If C_3 is determined by the routing, the scaling of W_4 and L_4 does not affect its value, resulting in the behavior presented in Fig. 6.48.

The scaling of W_3 and L_3 affects Z_u, as presented in Fig. 6.49. The pole $p2$ in Eq. (6.62) is moved towards lower frequencies due to the decrease of g_{ds3}, until it cancels the zero $z1$. Further increasing the dimensions makes the pole p dominant resulting in Eq. (6.63). The zero $z2$ is unaffected. Scaling up W_3 and L_3 increases C_2, but its effect on z_u is negligible. Therefore, it really does not matter whether C_2 is constant or not, as long as $\frac{g_{m2}C_2}{g_{ds2}g_{ds1}} \ll \frac{C_1}{g_{ds1}}$.

The scaling of W_2 and L_2 affects Z_u, as presented in Fig. 6.50. The pole $p2$ in Eq. (6.62) is moved towards lower frequencies due to decreasing g_{ds2} until it cancels the zero $z1$. Further increasing the dimensions makes the pole p dominant, resulting in Eq. (6.63). The zero $z2$ is unaffected.

The scaling of W_3 and L_3 in the 1-cascode case of Fig. 6.45 affects Z_u as presented

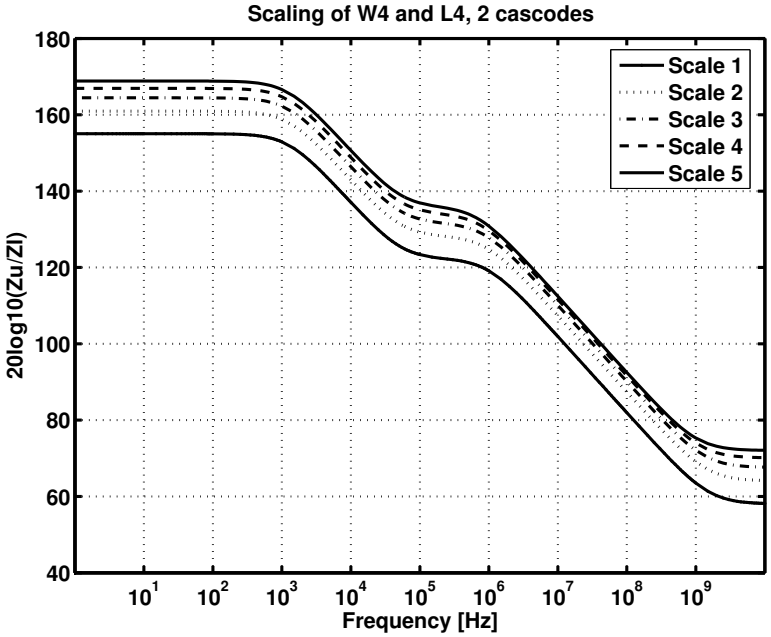

Figure 6.48 Scaling W_4 and L_4 in 2-cascode case with constant C_3.

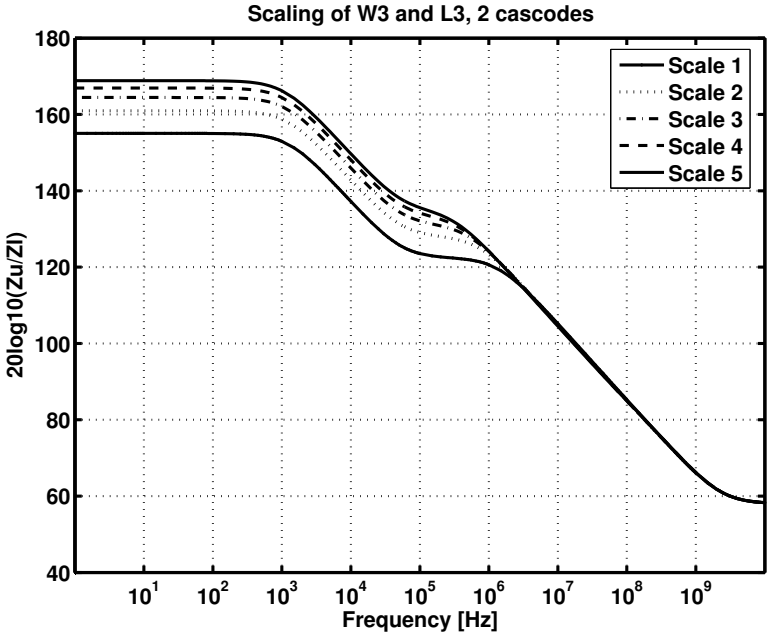

Figure 6.49 Scaling W_3 and L_3 in 2-cascode case.

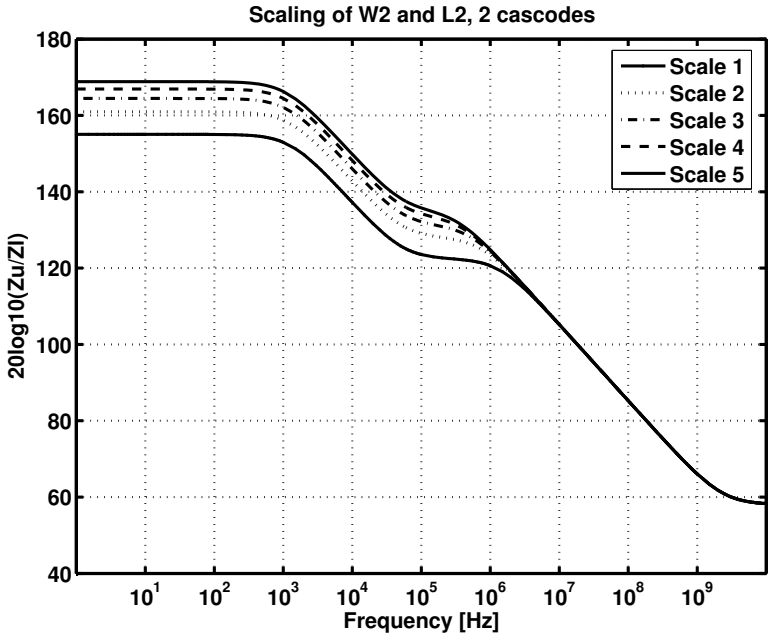

Figure 6.50 Scaling W_2 and L_2 in 2-cascode case.

in Fig. 6.51. The pole $p2$ in Eq. (6.69) is moved towards lower frequencies due to increasing C_2 until it cancels the zero $z1$. Further increasing the dimensions makes the pole p dominant, resulting in Eq. (6.70). The zero $z2$ is moved down in frequency due to increasing C_2.

If C_2 is determined by the routing, the scaling of W_3 and L_3 does not affect its value, and results in behavior similar to that shown in Fig. 6.48.

The effect of scaling W_2 and L_2 is as presented in Fig. 6.52. The pole $p1$ in Eq. (6.69) is moved to lower frequencies due to decreasing of g_{ds2}, until it cancels the zero $z1$. Further increasing the dimensions makes the pole p dominant, resulting in Eq. (6.70). The zero $z2$ remains unaffected.

The behavior of the output impedance of a current source without cascode transistors is presented in Fig. 6.53. Since C_1 is the only capacitance present and determined by routing, the pole p in Eq. (6.76) is not affected, nor is the zero z. Increasing the dimensions only scales the gain and thus increases the output impedance.

The effects of the cascode transistors and scaling of the transistor dimensions can be concluded as follows. Adding cascode transistors will, in general, increase the output impedance. Output impedance at higher frequencies is absolutely limited by Eqs. (6.63), (6.70), and (6.76), once the dominant pole is not caused by the routing capacitance C_1. In any case, it is beneficial to increase the impedance by scaling g_m

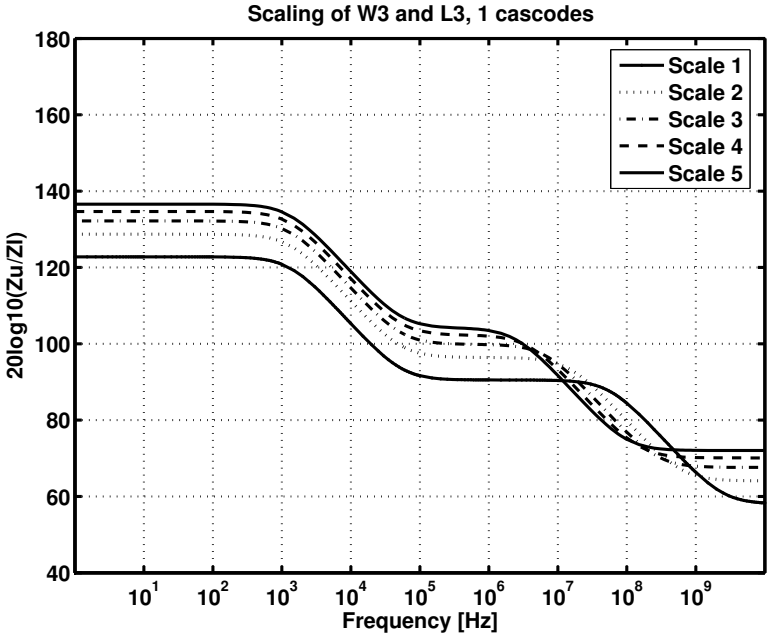

Figure 6.51 Scaling W_3 and L_3 in 1-cascode case.

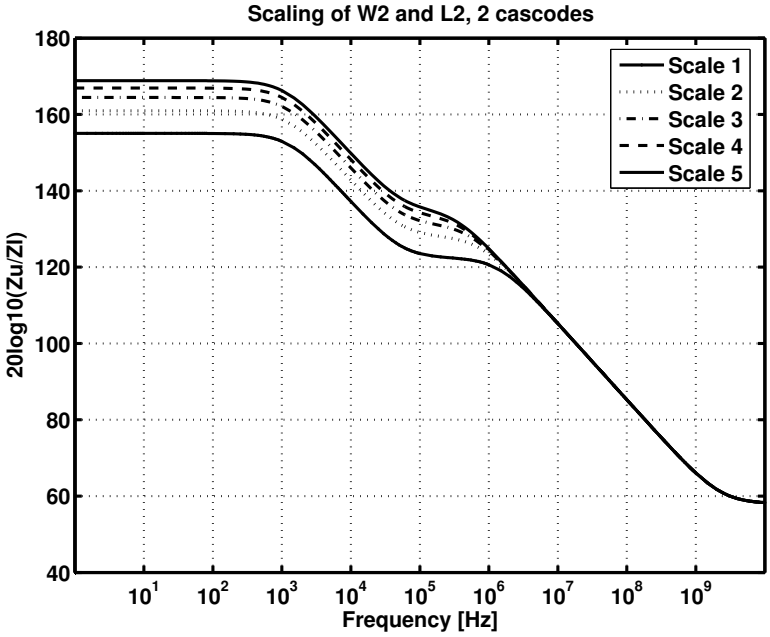

Figure 6.52 Scaling W_2 and L_2 in 1-cascode case.

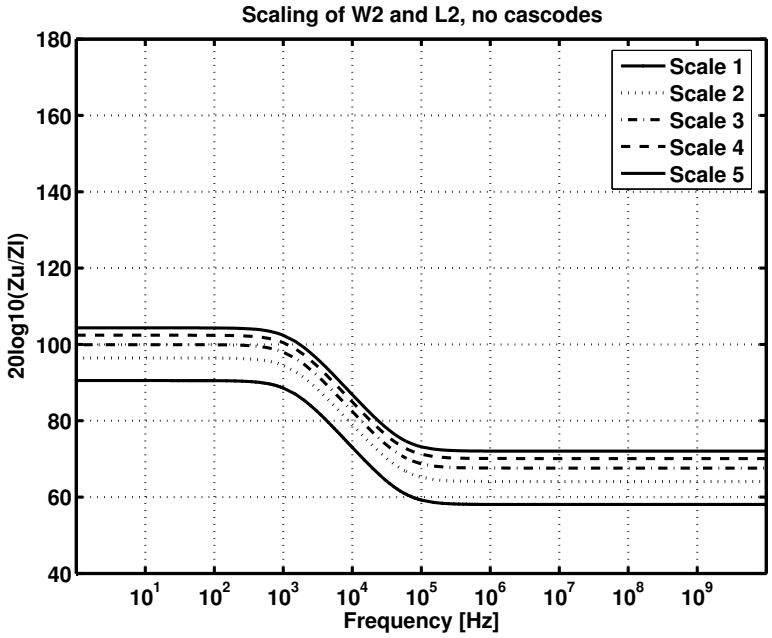

Figure 6.53 Current source with one cascode transistor.

and g_{ds} of the switch, since it increases the impedance at all frequencies, whereas scaling the cascodes seems to increase the impedance on the frequencies below the second pole.

The impedance can also be increased by scaling only W of transistors; however, the impedance is more effectively increased if L is also scaled. It should be noticed that it is not possible to increase L, since it would affect the operating point and drop the transistors out of the saturation.

As a rule of thumb, the impedance should be increased first by maximizing the gain of the switch (the topmost transistor) within the limits set by the driving capability of the switch driver and capacitive coupling from the switch gates. After that, the impedance on the lower frequencies can be further boosted by increasing the size of the cascode transistors if the pole due to the capacitor at the source node of the switch is not dominant.

From the output impedance point-of-view, it would be probably beneficial to use either transistor M2 or M3 as a switch and place additional cascodes above them; in this case, the large transistor would not reduce the switching speed. This kind of structure is used in the prototype described in Section 7.1.

6.6 From discrete- to continuous-time domain

In the digital domain, the signal is a train of impulses with certain values without any other nonideality but quantization noise. When the signal is fed to the D/A converter, the converter adds nonidealities to the signal while converting it to the analog form. In previous sections, the static nonlinearities were discussed and it was stated that *INL* sets the limit of the linearity of the converter. In this section, the conversion from the discrete-time to continuous-time domain is analyzed by presenting the D/A converter as a system consisting of several subsystems each providing a response to the digital discrete-time excitation signal. Typical subsystem impulse responses and their Fourier transforms are presented in Appendix D. Observations on the nonidealities generated due to conversion are listed at the end of the section.

The discrete-time excitation signal can be modeled as a continuous-time signal sampled with a sequence of Dirac's delta impulses (the effect of the quantization is not considered here)

$$s(t) = s(nT_s) = s(n) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s), \tag{6.84}$$

in which T_s is the sampling interval. The time domain sampling signal has an frequency domain equivalent of

$$S(f) = F_s \sum_{n=-\infty}^{\infty} \delta(f - nF_s), \tag{6.85}$$

in which F_s is the sampling frequency. Sampling of signal $X_c(t)$ with the sampling signal $S(t)$ results in a continuous-time model of the sampled signal

$$x(t) = x(nT_s) = x(n) = x_c(t) \sum_{n=-\infty}^{\infty} \delta(t - nT_s), \tag{6.86}$$

which has a spectrum

$$X(f) = F_s \sum_{n=-\infty}^{\infty} X(f - nF_s). \tag{6.87}$$

A system diagram of a D/A converter is presented in Fig. 6.54. An ideal D/A converter can be modeled with only one subsystem $h_1(t)$, which usually has the impulse response of a unit pulse of length T_s ($\tau = T_s$), forming a sampled-and-held non-return-to-zero (NRZ) type output signal (Fig. 6.55). The sample-and-hold impulse response of the subsystem is defined as

$$h_1(t) = u(t) - u(t - \tau), \tag{6.88}$$

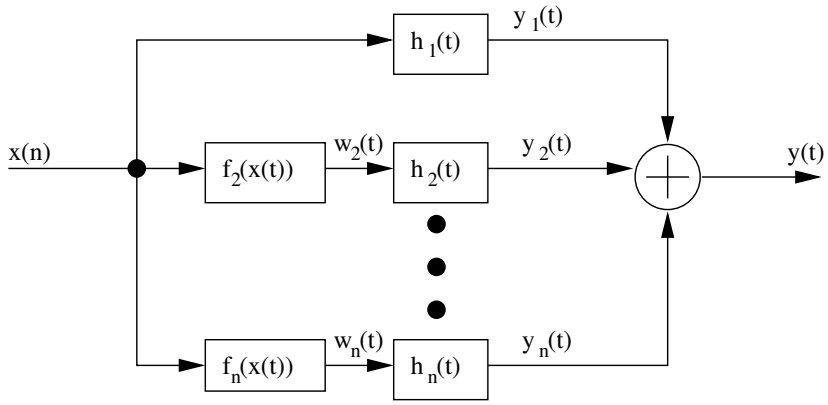

Figure 6.54 System diagram of the D/A converter.

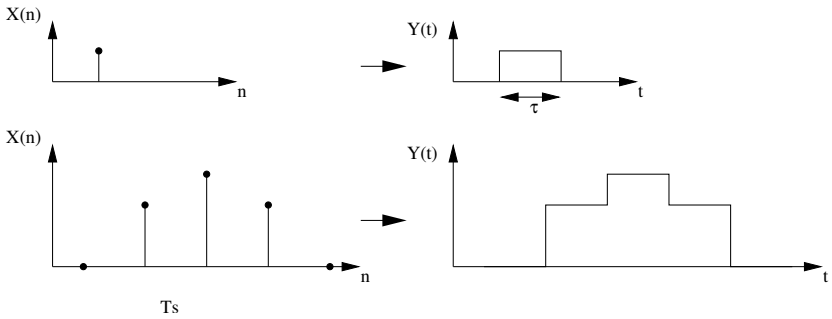

Figure 6.55 Sample-and hold response of the D/A converter.

in which τ is the hold time. The sample-and-hold system has a frequency response

$$H_1(f) = \tau \frac{\sin(\pi f \tau)}{\pi f \tau}. \tag{6.89}$$

If $\tau = T_s$, the system is an NRZ sample and hold. The response $y_1(t)$ of the subsystem $h_1(t)$ to excitation signal $X(t)$ can now be determined as a convolution of the input and the impulse response

$$\begin{aligned} y_1(t) &= x(t) \otimes h_1(t) \\ &= \int_{-\infty}^{\infty} x(t)\, h_1(t-\tau)\, dt \\ &= \sum_{-\infty}^{\infty} x(nT_s) \left(u(t-nT_s) - u(t-nT_s-\tau)\right), \end{aligned} \tag{6.90}$$

which can be presented in the frequency domain as

$$Y_1(f) = \frac{\tau}{T_s} \frac{\sin(\pi f \tau)}{\pi f \tau} \sum_{n=-\infty}^{\infty} X(f - nF_s). \tag{6.91}$$

In addition to the sample-and-hold subsystem, there may exist several subsystems that may or may not provide nonlinearity to the output signal. One of the simplest nonidealities to be included in the converter model is the effect of the finite rise and fall times and their asymmetry.

The effect of the finite rise and fall times can be included in the model by adding a subsystem $h_2(t)$ which provides an impulse response, which, together with the sample-and-hold impulse response of $h_1(t)$, models the finite rise/fall time during the transition (Fig. 6.56).

The impulse responses corresponding to the linear rising and falling transitions can be written as

$$h_r(t) = \left(\frac{t}{T_r - 1}\right) \left(u(t) - u(t - T_r)\right) \tag{6.92}$$

$$h_f(t) = \left(\frac{t}{T_f - 1}\right) \left(u(t) - u(t - T_f)\right), \tag{6.93}$$

in which T_r and T_f are the rise and fall time, respectively. The Fourier transforms of h_r and h_f are

$$H_r(f) = \frac{\left(e^{-j\omega T_r} - 1\right) - 1}{\omega^2 T_r} - \frac{1}{j\omega} \tag{6.94}$$

$$H_f(f) = \frac{\left(e^{-j\omega T_f} - 1\right) - 1}{\omega^2 T_f} - \frac{1}{j\omega} \tag{6.95}$$

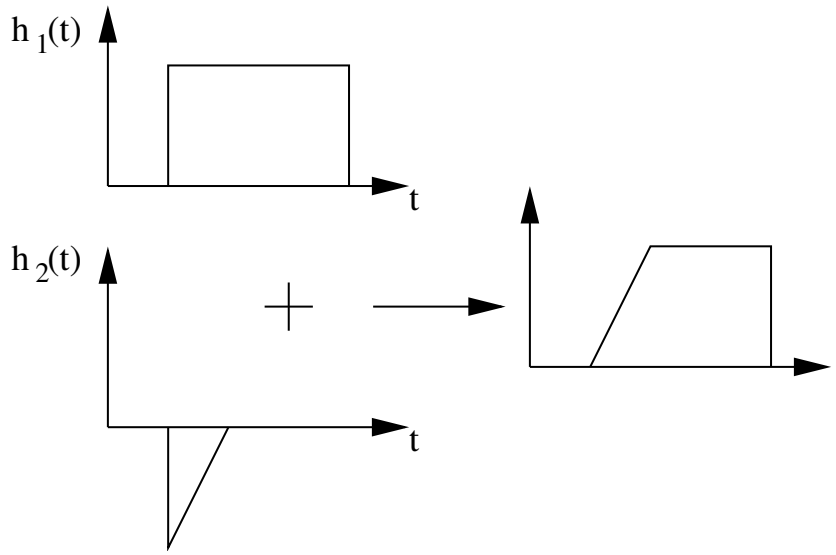

Figure 6.56 Modeling the transition with subsystem $h_2(t)$.

The effect of the finite transition time can be modeled as follows.

$$y_2(t) = \begin{cases} w_2(t) \otimes h_r(t), & w_2(t) \geq 0 \\ w_2(t) \otimes h_f(t), & w_2(t) < 0 \end{cases} \tag{6.96}$$

in which $w_2(t)$ is the discrete-time derivative of the input signal

$$w_2(t) = (x_c(t) - x_c(t - T_s)) \sum_{n=-\infty}^{\infty} \delta(t - nT_s). \tag{6.97}$$

The Fourier transform of $w_2(t)$ is

$$W_2(f) = \left(1 - e^{-\frac{2\pi f}{F_s}}\right) \sum_{n=-\infty}^{\infty} X(f - nF_s), \tag{6.98}$$

which approaches $j\omega X(F)$as F_s approaches infinity, corresponding to the Fourier transform of the continuous-time derivative of the signal $x_c(t)$.

The effect of the difference of the rise and fall can be determined by decomposing the response of the subsystem h_2 as follows. The average response common for both the rising and falling edges can be defined as

$$h_{2a}(t) = \frac{h_r(t) + h_f(t)}{2}, \tag{6.99}$$

resulting in average transition response

$$y_{2a}(t) = w_2(t) \otimes h_{2c}(t). \tag{6.100}$$

The difference in the rise times can be included by adding a subsystem having an impulse response

$$h_{2b}(t) = \frac{h_r(t) - h_f(t)}{2} \tag{6.101}$$

with excitation signal

$$\begin{aligned} w_{2b}(t) &= |x_c(t) - x_c(t - T_s)| \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \\ &= |w_2(t)|, \end{aligned} \tag{6.102}$$

resulting in response due to differences between rise and fall times

$$y_{2b}(t) = w_{2b}(n) \otimes h_{2b}(t). \tag{6.103}$$

For the sine signal, the spectrum of w_{2b} can be computed by using Eqs. (D.6), (D.8), (D.7), and (D.19) resulting in

$$\begin{aligned} W_{2b}(f) &= A\left(1 - e^{j2\pi f_0}\right) F_s \sum_{n=-\infty}^{\infty} \sum_{k=-\infty,odd}^{\infty} \\ &\frac{1}{\pi k} \left(\delta(f - (k+1) f_o - nF_s) - \delta(f - (k-1) f_o - nF_s)\right). \end{aligned} \tag{6.104}$$

From Eq. (6.104), it can be clearly seen that the absolute value of a sine signal contains even-order harmonic components. When the transition shapes of the converter output are known, the amount of distortion due to transition asymmetry can be easily computed with Eqs. (6.103) and (6.104).

As a conclusion, the following observations are made:

1) The shape of the impulse response of a subsystem does not introduce any nonlinearity as long as it is independent from the excitation signal. This means that glitches at the output do no harm as long as they are a part of the signal independent impulse response of the subsystem. Therefore, observing the glitches at the output of the converter does not necessarily give any information about the linearity of the converter.

2) Transitions can be modeled with a subsystem with discrete-time derivative as an excitation signal. Differentiation is a linear operation and therefore finite transition speed does not generate distortion if the rising and falling transitions are symmetrical.

3) Asymmetry in transition-related nonidealities can be modeled with a subsystem with an absolute value of the discrete-time derivative of the input signal as an excitation signal. Absolute value is a nonlinear even function, introducing even-order harmonics, as demonstrated in the case of the sine signal in Eq. (6.104). Total power of the harmonic component is defined by the shapes of transitions according to Eq. (6.103).

4) With sinusoidal signals, the amplitude of the derivative is linearly dependent on the signal frequency (Eq. (D.7)), resulting in increased distortion as the signal frequency increases. This holds for all transition-related nonidealities, including jitter, as can be seen later in Section 6.7.

5) As the converters are usually segmented, the effect of the subsystems for the binary weighted part should be analyzed on a per switch basis. However, the performance is usually dominated by the thermometer coded part.

6.7 Sampling jitter

During the past few years, timing has become an important issue in the design of current-steering D/A converters, due to the rapid increase of the sampling rates. The importance of the static timing mismatch has been demonstrated by Chen et al. [145], and the reduction of the data dependent clock load (which is a source of jitter) was mentioned by Schofield et al. [139]. Special attention has been paid on timing and jitter issues in the design reported in [146]. Also, various analyzes of timing and jitter have been published [147], [148], [149].

In this section, a jitter model for a fully thermometer-coded D/A converter is developed. Thermometer coding is chosen because it simplifies the computation while not resulting in significant inaccuracies, since the effect of timing inaccuracy of the binary weighted LSB bits is typically the order of $\frac{1}{64}$ to $\frac{1}{16}$ compared to the effect of thermometer-coded MSB bits (6 to 4 thermometer-coded MSB bits). Jitter analysis for the binary-weighted part has to be performed on a per switch basis and would result in very large signal matrices, because, in simulations, a high oversampling ratio has to be used in order to be able to model the jitter.

In the following sections, a brief introduction to the relations of the jitter on *SNR* of the converter is given. It is demonstrated by simulations how certain types of timing uncertainty are mapped to the output signal of a D/A converter. Some sources of jitter in current-steering D/A converters are identified and analyzed by design examples.

6.7.1 Effects of sinusoidal timing jitter

The jitter analysis in this book concentrates on the sinusoidal jitter signals because they are most likely to produce spurious or harmonic tones to the output of a converter, thus limiting *SFDR*.

The output signal of a non-return-to-zero (NRZ) D/A converter with sampling jitter can be defined as the sum of the ideal sampled-and-held signal and the timing-jitter-induced error signal (Eq. (6.105) and Eq. (6.106))

$$\begin{aligned} y(t) &= \sum_{n=-\infty}^{\infty} x(nT)\,[u(t-nT) \\ &\quad -u(t-(n+1)T)] + e(t), \end{aligned} \tag{6.105}$$

$$\begin{aligned} e(t) &= \Delta x(n)\, g(t) \\ &= \sum_{n=-\infty}^{\infty} [x(nT) - x((n-1)T)] \\ &\quad \times [u(t-nT-w(nT))) - u(t-nT)], \end{aligned} \tag{6.106}$$

in which T is the sampling interval, $u(t)$ is the unit step function $x(nT)$ is the digital discrete-time input of a converter, and $w(nT)$ is the value of the timing-jitter signal at the time instant nT (Fig. 6.57). For simplicity, it is assumed here that the rise time of the signal is zero. The effect of the nonzero rise time is analyzed later in this section. A block diagram of the jitter model is presented in Fig. 6.58

The effect of jitter on the output signal of the converter can be determined by computing the spectrum of the jitter signal $g(t)$ as

$$G(f) = \sum_{n=-\infty}^{\infty} \int_{nT+w(n)}^{\infty} e^{-j\omega t}\,\mathrm{d}t - \int_{nT}^{\infty} e^{-j\omega t}\,\mathrm{d}t \tag{6.107}$$

$$= \sum_{n=-\infty}^{\infty} \frac{\left(e^{-j\omega w(nT)} - 1\right) e^{-j\omega nT}}{j\omega}. \tag{6.108}$$

The spectrum of the error signal $g(t)$ resulting from the sinusoidal jitter signal $w(n) = a\sin(\omega_1 nT)$ is then

$$G(f) = \sum_{n=-\infty}^{\infty} \frac{\left(e^{-j\omega a \sin(\omega_1 nT)} - 1\right) e^{-j\omega nT}}{j\omega}. \tag{6.109}$$

If a is small, Eq. 6.109 can be approximated with the Taylor-series expansion as

$$G(f) \approx F_s \sum_{n=-\infty}^{\infty} \frac{-a}{j2} \left(\sigma(f + f_1 + nF_s) - \sigma(f - f_1 + nF_s)\right). \tag{6.110}$$

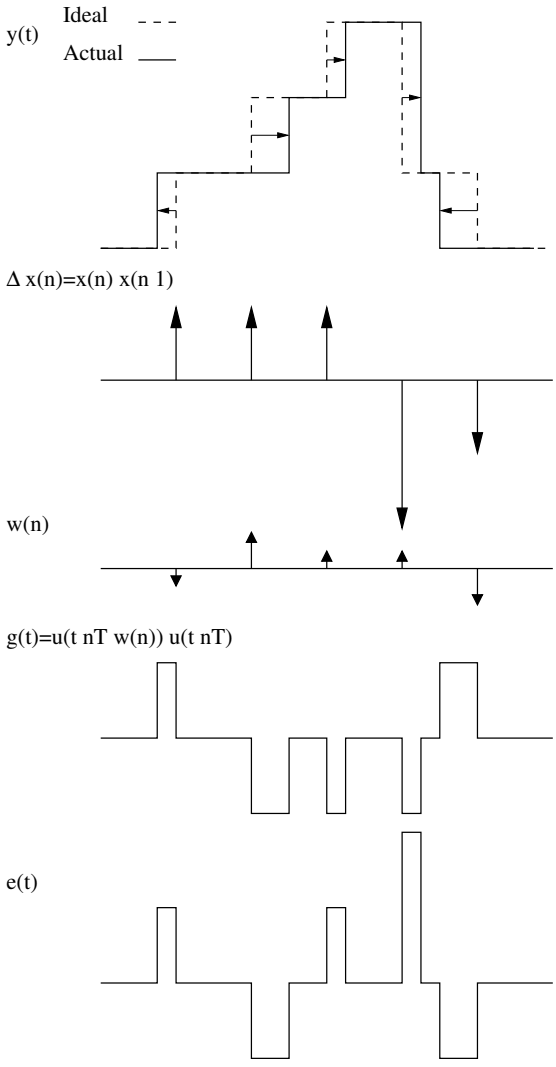

Figure 6.57 Components of the output $y(t)$.

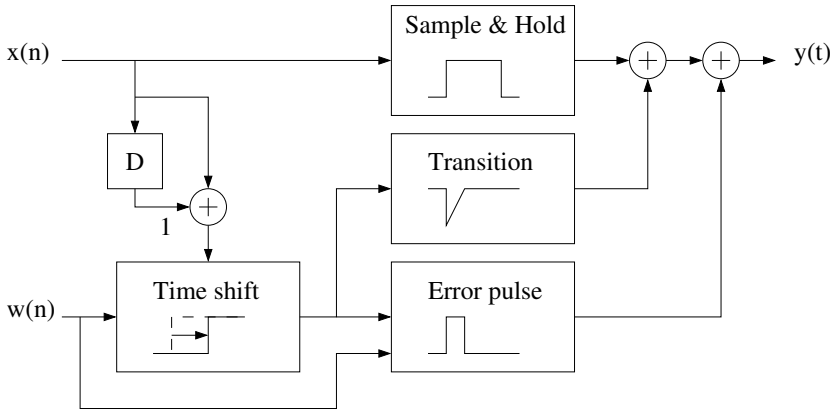

Figure 6.58 Block diagram of the jitter model.

An analytical solution for the power spectrum of sinusoidal signals with the presence of sinusoidal jitter and Gaussian noise is presented in[150], and is given as

$$S(f) = \frac{|C(f)|^2}{(\pi f T)^2} \sum_{p,q} \sum_{k=1,odd} \frac{A_k^2 \sin^2(\pi f T)}{4} I_p(M)\, \sigma(f \pm f_k - p f_1 - q f_s) \quad (6.111)$$

$$|C(f)|^2 = e^{-\omega^2 \sigma_E^2} \quad (6.112)$$

$$M = \frac{8\pi^2 f^2 \sigma_E^2 A_1^2}{(2^N - 1)} \sin\left(\frac{\omega_1 T}{2}\right), \quad (6.113)$$

where $C(f)$ is the Fourier transform of the autocorrelation function of jitter, A_k is the amplitude of the signal and its harmonic components and $I_p(M)$ is a p^{th}-order modified Bessel function of the 1st kind and σ_E is the standard deviation of the Gaussian noise.

Further analysis of the effects of jitter are made by simulations in order to gain some insight into the jitter behavior.

Fig. 6.59 represents the eight first pulses of the jitter error signal of

$$g(t) = u(t - nT - w(n)) - u(t - nT), \quad (6.114)$$

$$w(n) = 0.01T \sin(2\pi 0.188n), \quad (6.115)$$

where the jitter signal $w(n)$ has the frequency of $0.188F_s$ and amplitude of $0.01T$, T being the sampling interval.

The spectrum of $g(t)$ with $T_r = 0$ is presented in Fig. 6.60, in which the power of the signal $g(t)$ is presented relative to the power of sinusoid $\sin(\omega nT)$. It can be observed that, in the frequency range 0 to $\frac{Fs}{2}$, the dominant frequency component is at the frequency $0.188F_s$, indicating that the energy of the jitter is mainly concentrated on the frequency of the jitter signal, as predicted by Eq. (6.110). It may also be observed that a significant amount of energy lies at frequencies higher than $\frac{F_s}{2}$.

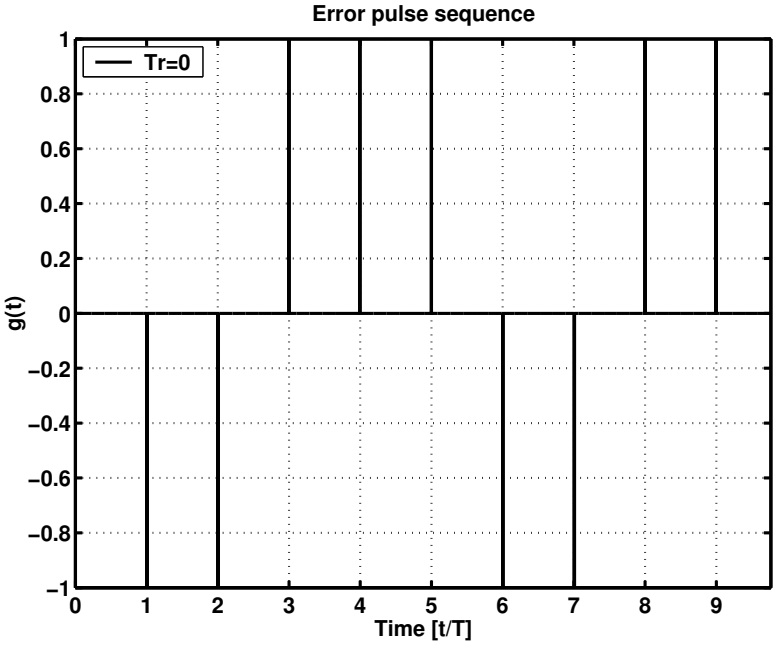

Figure 6.59 Error pulse signal $g(t)$, $T_r = 0$.

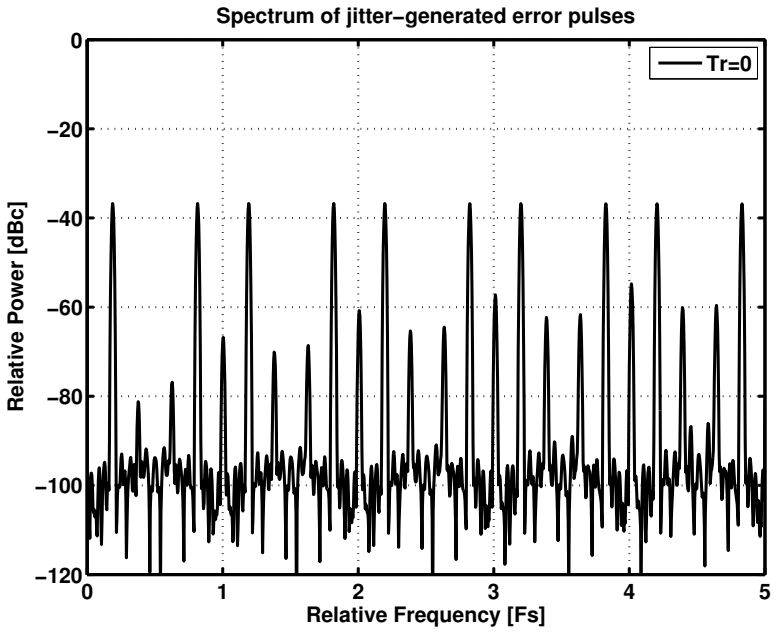

Figure 6.60 Spectrum of error pulse signal $g(t)$.

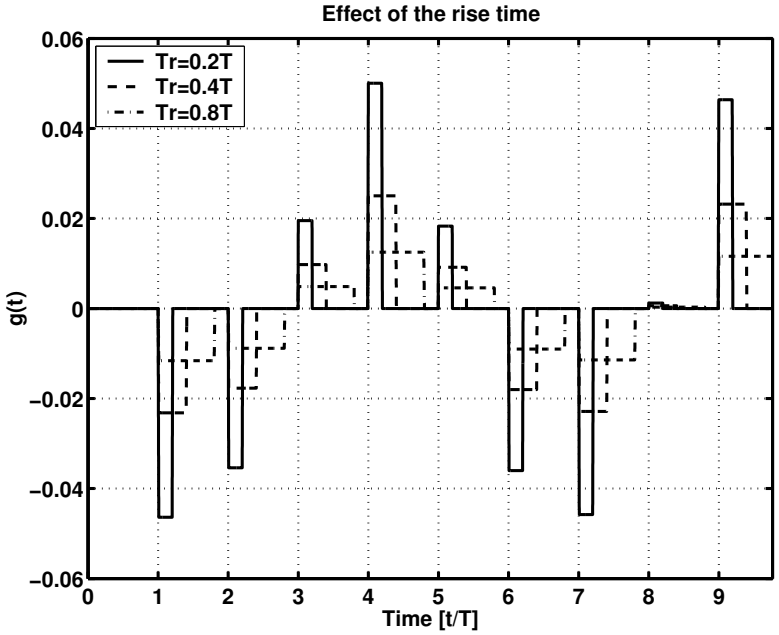

Figure 6.61 Eight first pulses of $g(t)$ with various T_r.

As the rise time increases, for example, because of the capacitive load at the output of the converter, it effects the duration and amplitude of the $g(t)$ according to

$$g(t) = \frac{t-nT-w(n)}{T_r}u(t-nT-w(n)) - \frac{t-nT-T_r-w(n)}{T_r}u(t-nT) - \frac{t-nT}{T_r}u(t-nT) + \frac{t-nT-Tr}{T_r}u(t-nT-Tr). \quad (6.116)$$

The effect of finite rise and fall time on the time-domain error pulses can be seen from Fig. 6.61. It can be observed that the shape of the error pulse is an isosceles trapezoid with duration and amplitude dependent on $w(n)$. Spectra of error pulses under various rise-time conditions are presented in Figs. 6.62 and 6.63. $w(n)$ is still defined by Eq. (6.115).

It is obvious that, as the rise time increases, the energy at the higher frequencies is reduced, resulting in reduced total jitter energy, whereas energy in the frequency range of 0 to $\frac{F_s}{2}$ is barely affected. This is verified by simulations, and the result is presented in Fig. 6.63, which represents the signal-to-error power ratio and *SFDR* as a function of rise time.

The signal-to-error power ratio is computed from the time-domain signal, whereas *SFDR* is calculated from spectral components from 0 to $F_s/2$. Simulation results indicate that, in the case when *SFDR* is determined by the jitter, *SFDR* can neither be

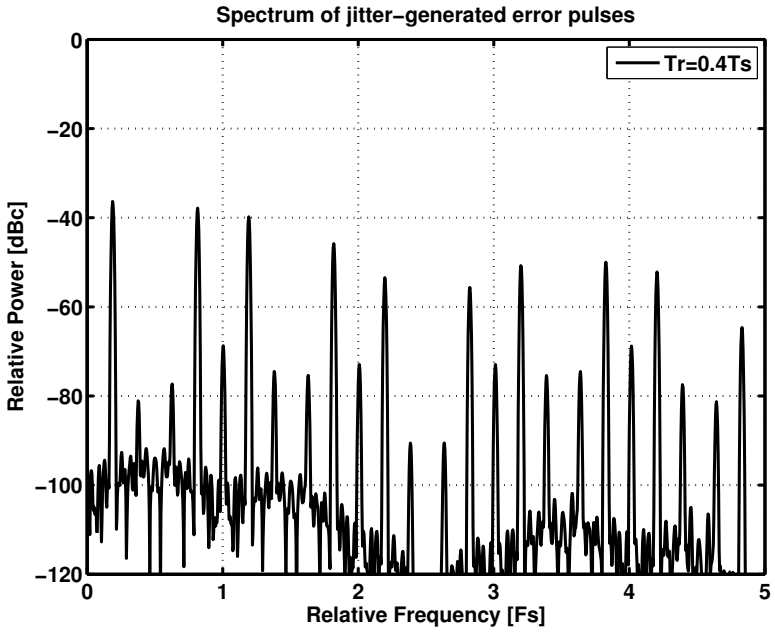

Figure 6.62 Spectrum of $g(t)$ with $T_r = 0.4T$.

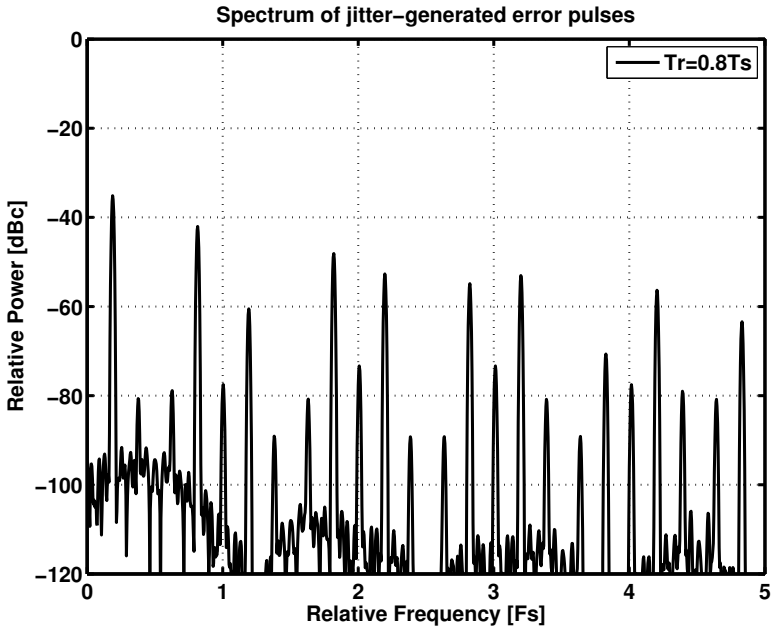

Figure 6.63 Spectrum of $g(t)$ with $T_r = 0.8T$.

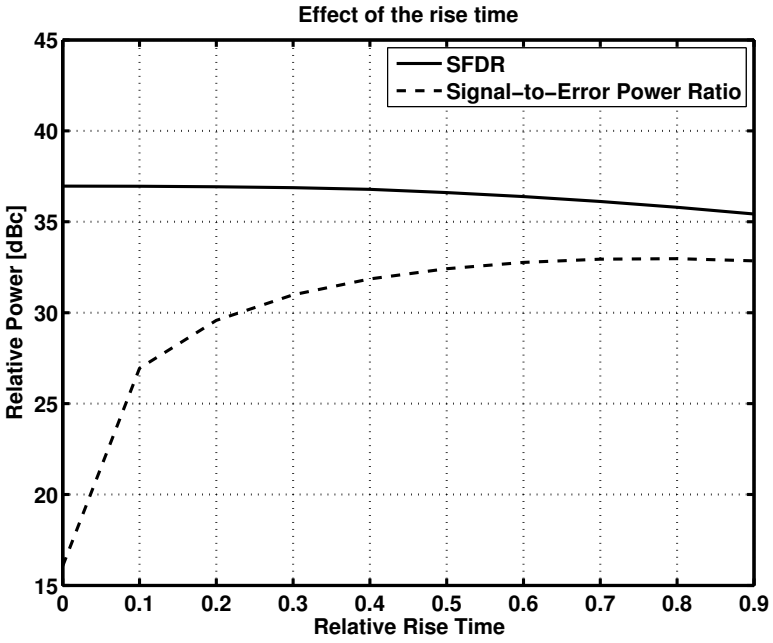

Figure 6.64 Signal-to-error power ratio and $SFDR$ as a function of rise time.

increased by altering the rise time nor does there exist an optimal rise time in the $SFDR$ sense, meaning that $SFDR$ is determined by only the standard deviation of $w(n)$. Typical behavior of $SFDR$ as a function of the jitter magnitude of sinusoidal jitter $w(n)$ is presented in Fig. 6.65.

For Gaussian jitter, the derivation for the jitter energy on a certain frequency band from $-F_b$ to F_b resulting in SNR for a single sinusoid at the frequency F_{sig} sampled with the rate F_s, is presented in Appendix C.

$$SNR(F_b) \approx \frac{F_s^2}{4\pi^2 F_{sig}^2 F_b \sigma_w^2} \tag{6.117}$$

In the case of return-to-zero (RZ) type of signals, jitter-induced $SFDR$ decreases, since shortening the signal pulse reduces signal energy linearly, whereas the amount of jitter error energy remains constant.

6.7.2 Distortion due to signal-dependent jitter

Fig. 6.66 represents the spectra of a quantized sinusoidal signal $x(nT)$ and a jitter signal $w(nT)$, which has second-order dependency on the input signal as

$$x(nT) = Q[\sin(2\pi f nT)] \tag{6.118}$$

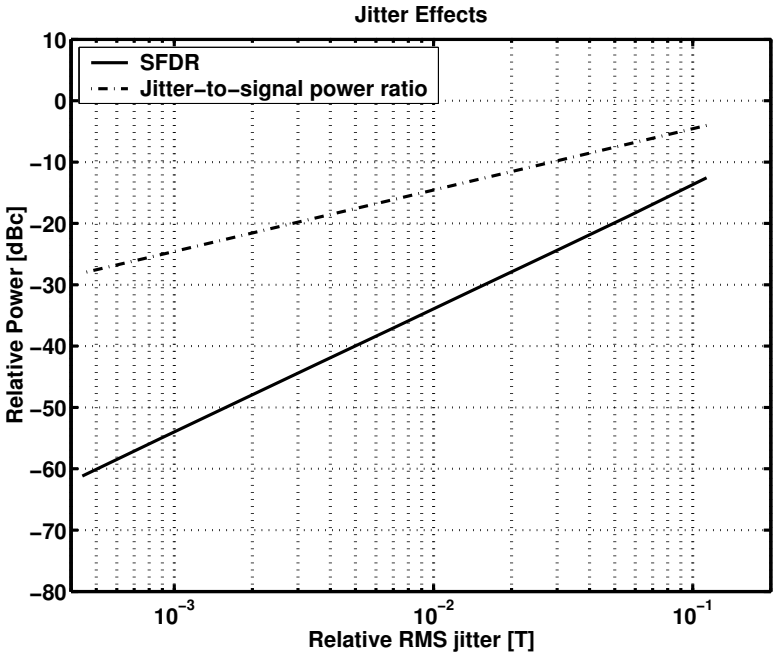

Figure 6.65 Signal-to-error power ratio and $SFDR$ as a function of jitter magnitude.

$$w(n) = K|x(nT) - x((n-1)T)|^2, \tag{6.119}$$

in which K is chosen so that

$$\max\left(|K(x(nT) - x((n-1)T))|^2\right) = 0.01T. \tag{6.120}$$

$w(n)$ in Eq. (6.119) is chosen to be dependent on the absolute value of the discrete time derivative of the input signal since it reflects the physical origin of the jitter; often the jitter is caused by activity of the circuitry, which is usually dependent on the number of changing thermometer-coded bits.

In Fig. 6.67, the spectrum of $e(t)$ in the case of second order jitter $w(nT)$ is presented.

It can be observed that due to multiplication by the discrete time derivative (Eq. (6.119)), the resulting distortion is of odd order. As a conclusion it can be stated that even-order dependency of the $w(nT)$ on the input signal $x(nT)$ results in odd-order distortion of the output signal $y(t)$ and vice versa.

The signal-dependent jitter can originate from power-rail interference of the switch driver circuitry of the converter or from the digital circuitry due to the substrate coupling, from which the first one is usually the dominant source of jitter. Clock loading, if not taken into account, may also result in jitter. Examples of these two jitter sources

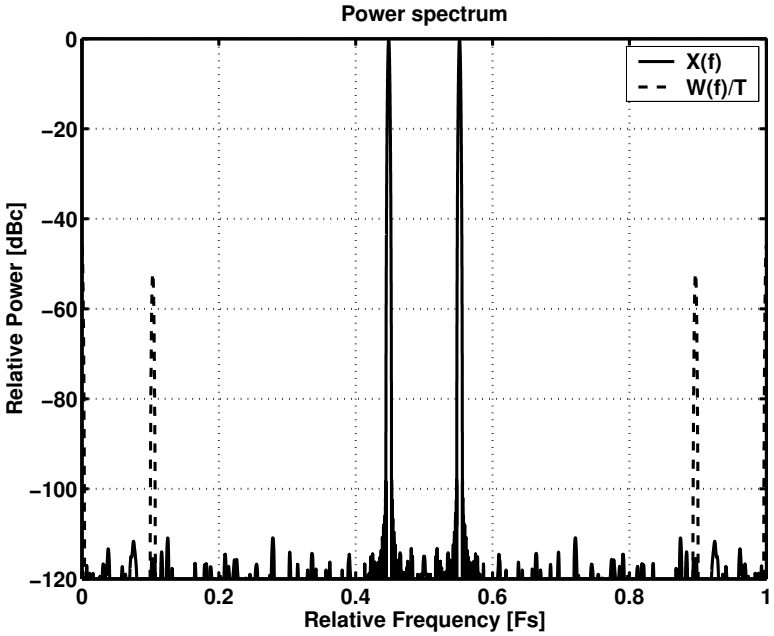

Figure 6.66 Spectra of the sinusoidal signal $x(nT)$ and second order jitter signal.

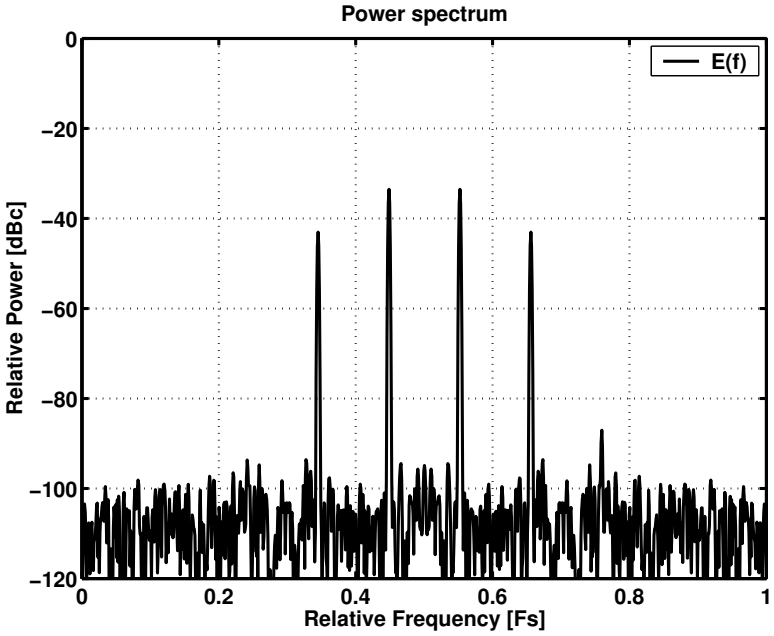

Figure 6.67 Spectrum of the error signal $e(t)$ caused by second order jitter signal $w(n)$.

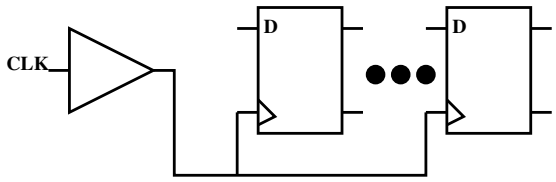

Figure 6.68 Buffering of the switch drivers with single buffer.

are given in following sections.

6.7.3 Jitter due to code-dependent clock load

Sometimes, in the cases in which relatively accurate synchronization is required, the clock pins of the synchronizing elements may be driven by one, usually very large, buffer (Fig. 6.68). If the synchronizing elements driven are the latch/flip-flop stages of the switch-driver stage of the D/A converter, this may result in distortion of the output signal. This is because the load of the clock driver is dependent on the internal states of the switch driver flip-flops, which are dependent on the data that is fed to them. It can be assumed that code-dependent clock load causes code-dependent jitter. This assumption can be validated by simulations as follows.

Fig. 6.69 represents the output spectrum of a 16-bit current-steering D/A converter with the clock driving scheme of Fig. 6.68. This spectrum is the result of the transistor-level simulation of the converter implemented in a 0.35 μm Si-Ge BiCMOS process. In order to validate the assumption that the third-order distortion in Fig. 6.69 is due to jitter, the jitter was extracted from the clock signal of the transistor-level simulation. In Fig. 6.70, the output signal and the relative timing error of the clock signal are presented. It can be observed that the jitter seems to be correlated with the output signal, and thus dependent on the input code, and that the dependency is second-order. As stated before, in the case of even-order jitter, odd order distortion may be expected at the output as in Fig. 6.69.

In order to verify the origin of the distortion to be jitter, the jitter extracted from the simulation is applied to the model described by Eqs. (6.105) and (6.106). The spectra of $x(nT)$ and $w(n)$ from the jitter-model simulations are presented in Figs. 6.71 and 6.72.

It can be observed that the spectral behavior due to jitter follows quite accurately the spectrum obtained from the transistor-level simulation, and therefore it is justified to say that the distortion is due to the code-dependent jitter.

Distortion caused by code-dependent clock loading can be reduced by, for example, adding a small buffer to drive the clock pin of each of the switch driver flip-flops (Fig. 6.73). The simulation results for the D/A converter with modified clock buffering are

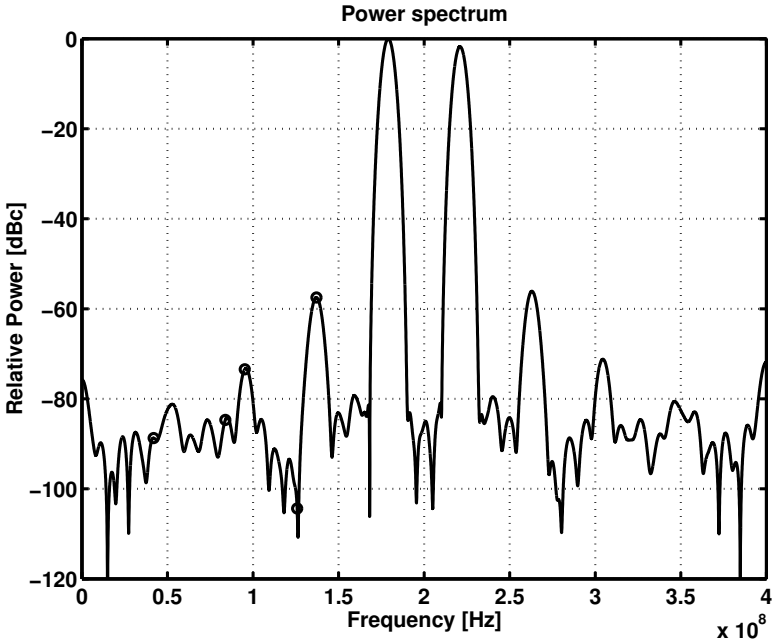

Figure 6.69 Simulated spectrum obtained from the transistor-level simulation.

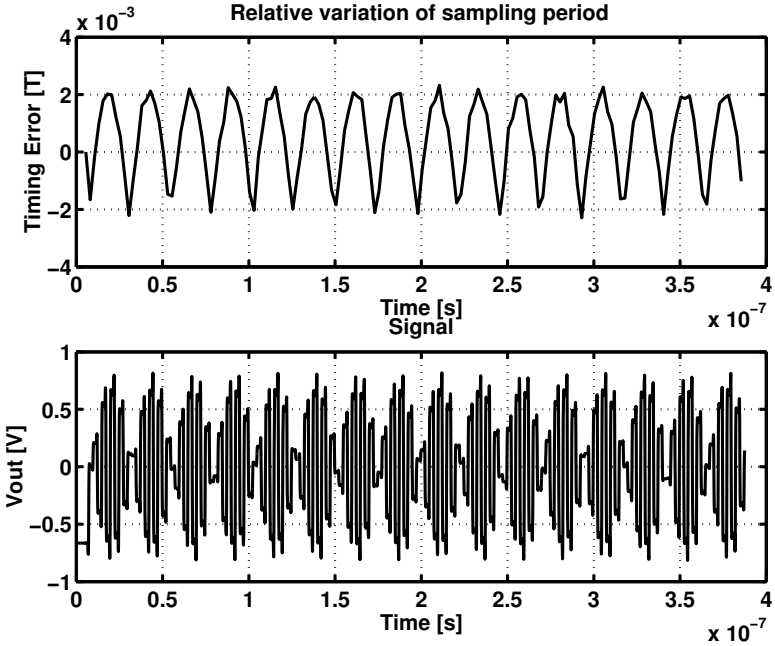

Figure 6.70 Relative timing error and the output signal of a 16-bit D/A converter.

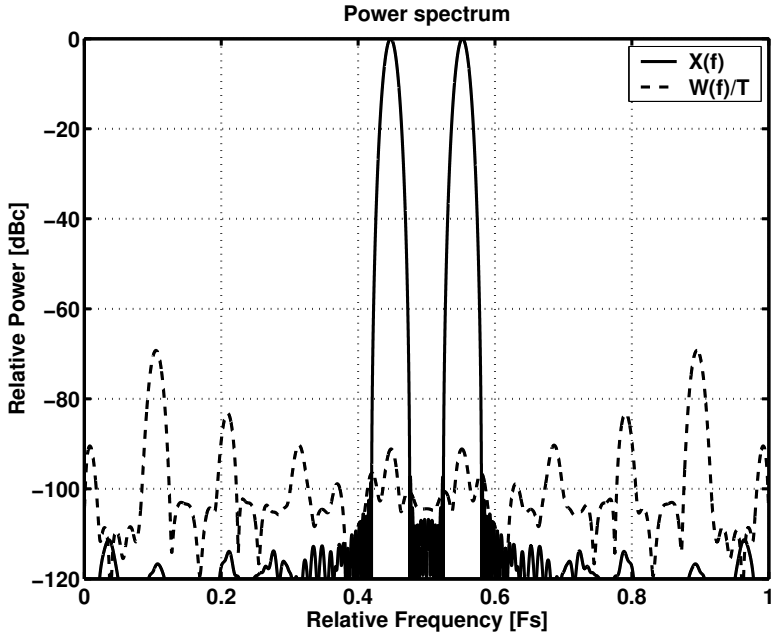

Figure 6.71 Spectra of the input signal $x(nT)$ and $w(nT)$.

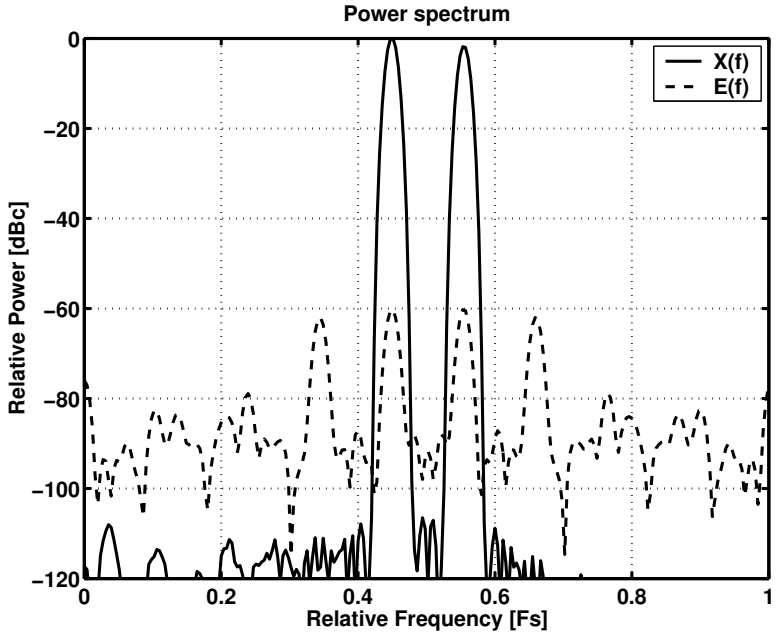

Figure 6.72 Spectra of the input signal $x(nT)$ and error signal $e(t)$.

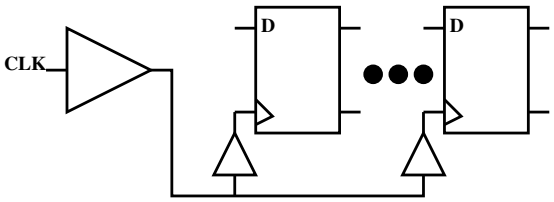

Figure 6.73 Modified clock buffering.

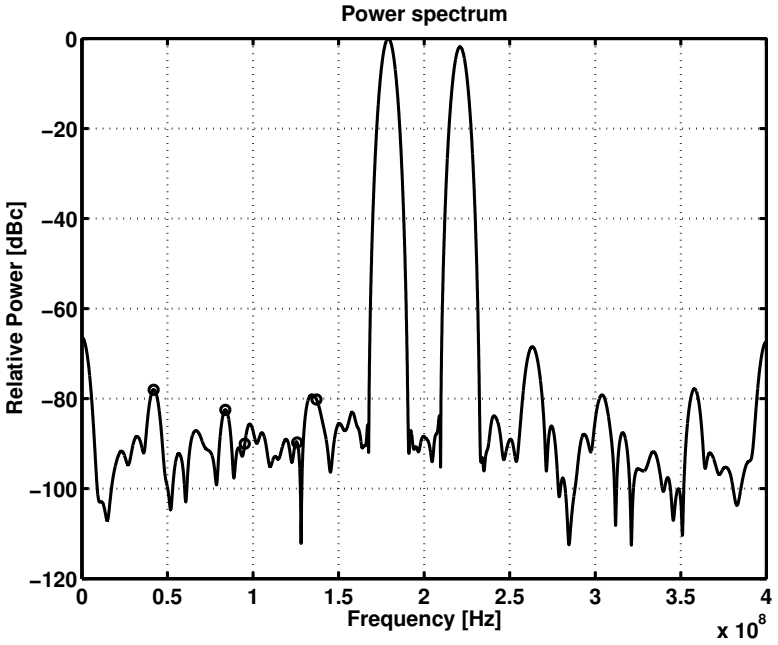

Figure 6.74 Simulated spectrum of the D/A converter with modified clock buffering.

presented in Figs. 6.74 and 6.75. It can be observed that the jitter-induced distortion is effectively reduced with the altered buffering scheme.

6.7.4 Jitter due to power-rail interference

It can be assumed that the parasitic resistance of the power supply rails of the switch driver circuitry may cause code-dependent interference during the transitions of the switch driving signals, which may result in jitter. The interference on the supply rails affects not only the clock signal (in those cases where the clock buffers share the same supply rails), but also the switch driving signals. Therefore, in order to extract the proper jitter signal from the transistor-level simulations, the jitter is extracted from the switch driver signals instead of the clock signal. In order to reduce the code dependency of the supply-rail interference, a transition of the switch driver signal may be

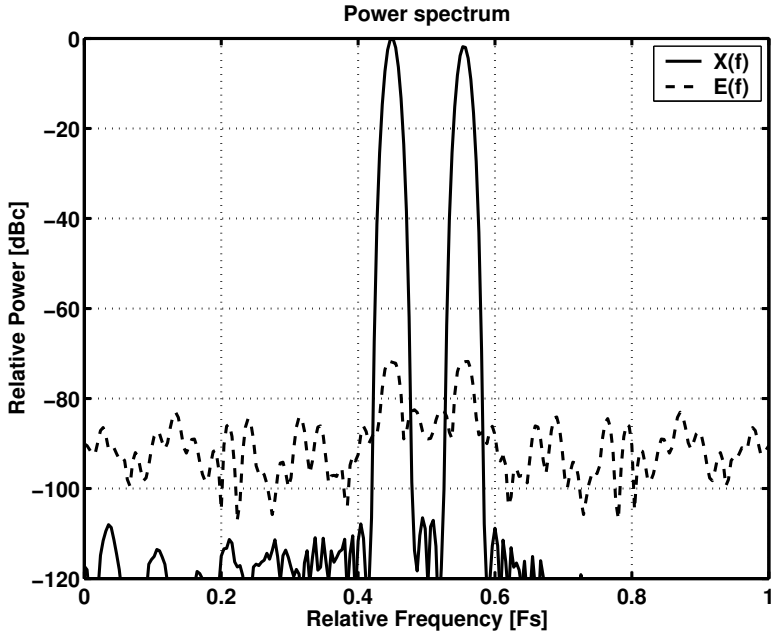

Figure 6.75 Effect of the jitter with modified clock buffering.

ensured on every clock cycle by using the differential-quad switching scheme [151]. Switch drivers are designed so that, when a transition occurs, the switch is opened before the other switch is closed. Jitter is extracted by monitoring the time instants when $V_{gs} - V_T$ of the closing transistor is zero, ensuring that there is only one conducting transistor at that time instant.

A lumped resistor model is used for the supply-rail resistance in order to avoid distortion being dependent on the position of the switch driver, which would be the case with a distributed resistance model. The position-dependency of the jitter would further exacerbate the performance of the D/A converter; it is, however, beyond the scope of this analysis.

Fig. 6.76 represents the output spectrum of the converter obtained from the transistor-level simulation, and Figs. 6.77 and 6.78 represent the reconstruction of the output spectrum with the mathematical jitter model. It can be observed that resistive supply rails can generate an excessive amount of jitter even though the differential-quad switching scheme [151] is used to reduce the code dependency of the power supply interference. The benefits of using the differential-quad switching scheme are discussed in more detail in Section 7.2.4.

In this section, the effects of the jitter on the spectral performance are demonstrated by using a mathematical jitter model. Two design-related examples of the jitter-induced distortion are given by analyzing the jitter signal extracted from transistor-level simu-

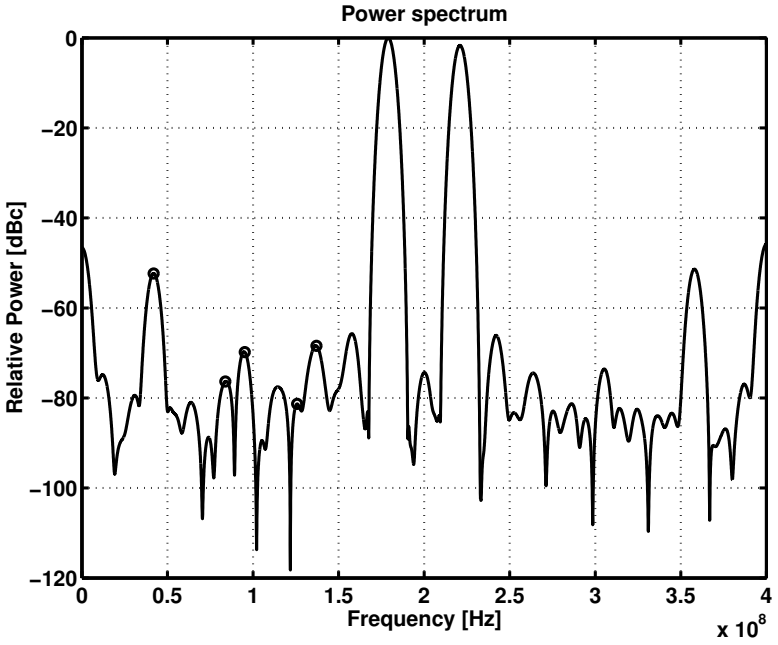

Figure 6.76 Simulated output spectrum of the D/A converter with resistive supply rails.

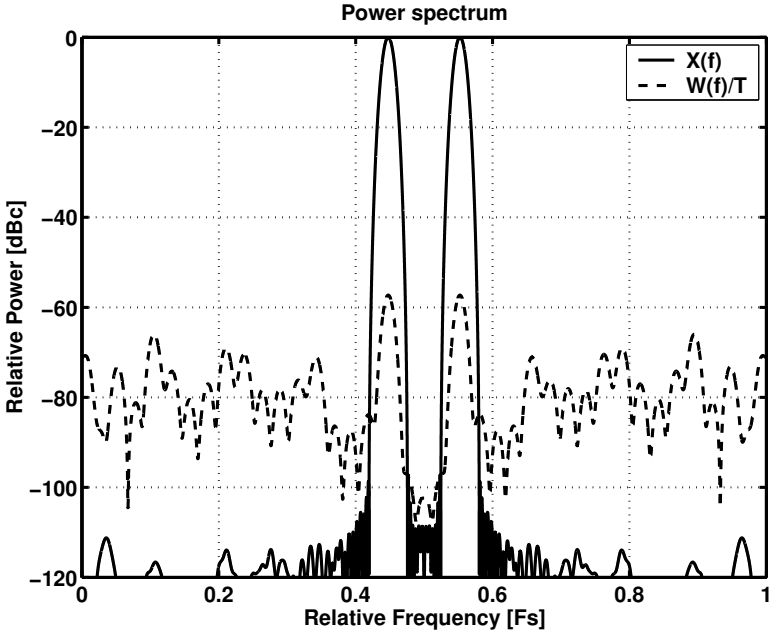

Figure 6.77 Spectra of the input signal $x(nT)$ and $w(n)$ due to supply rail interference.

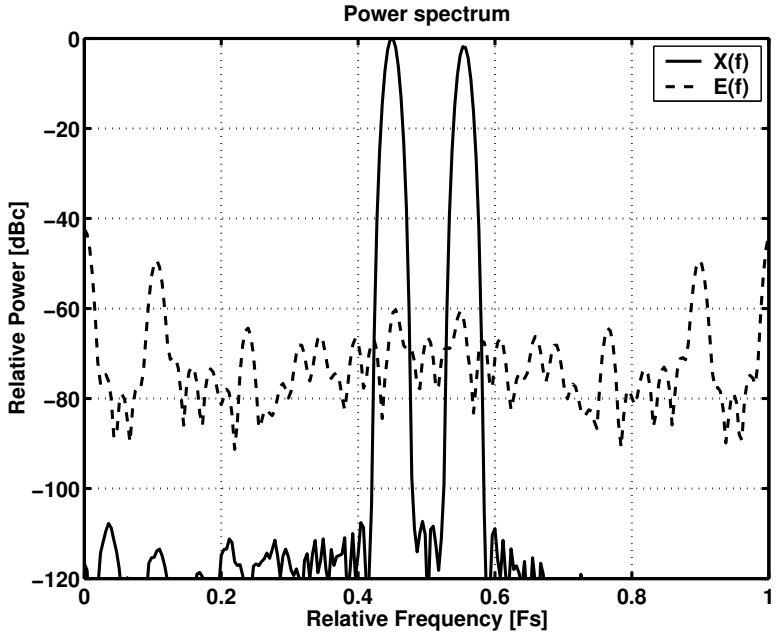

Figure 6.78 Effect of the jitter due to supply rail interference.

lations with the developed mathematical model. This model can be used to determine whether the distortion at the output of the converter is due to jitter or not.

As a summary, the following observations can be made of these analyzes: spectral components due to jitter are defined by the discrete-time derivative of the input signal and the jitter signal $w(n)$ The jitter signal that has an odd-order dependency on the output signal of the converter will generate even-order harmonics due to multiplication by the discrete-time derivative of the signal and vice versa.

Even though the total jitter-to-signal energy ratio is affected by the transition of the signal, the increase in transition time mainly reduces the frequency component above $\frac{F_s}{2}$ thus indicating that $SFDR$ due to jitter can not be optimized by altering the transition times.

It is demonstrated by simulations that the jitter-to-signal power ratio increases 3 dB if the amplitude of the jitter is doubled. However, $SFDR$ on the Nyquist band increases 6 dB due to the fact that increasing jitter amplitude transfers jitter energy from higher to lower frequencies.

6.8 Layout techniques for current source mismatch reduction

In addition to random variation of the process parameters that was discussed in Section 6.3, the gradient errors are an additional source of static nonlinearity. The gradient errors are mainly due to the variation of the oxide thickness on the wafer, which is stated to have a linear dependency on the distance between devices [125], and die stress gradients, which has been reported to have a quadratic behavior over the device matrix [152]. The gradient errors have a significant effect on the static linearity of the converter, since the error due to gradients correlate with each other and therefore the errors may accumulate, resulting in a large INL. The effect of accumulation can be reduced by choosing the switching order so that the gradient errors of current sources corresponding to the subsequent code values cancel each other. On the other hand, if the errors do not correlate, the switching order has no statistical effect on the INL.

There have been several proposals for the layout techniques (also called ”Switching schemes”) for the reduction of nonlinearity caused by process gradients. The methods can be divided in two categories. In heuristic methods [153], [154], [155], [156], [126], [157] , the cancellation of the gradient effect is based purely on the geometrical aspects without any knowledge of the relations between linear and quadratic (odd- and even-order) gradients. In the analytical methods presented in [158], [159], the optimal placement of the current sources (i.e. the optimal switching sequence) in the INL sense is evaluated by numerical methods. These methods require a priori knowledge about the relations of the linear and quadratic gradients.

In the following paragraphs, the layout techniques are presented in the order of publication to demonstrate the evolution of the layout techniques. The layout method used in the prototype circuit presented in Section 7.2 is also discussed since it differs slightly from the previously published methods.

The simplest possible sequence for ordering the current sources is presented in Fig. 6.79. In this ”sequential switching” scheme, the current sources are selected in the order they appear in the matrix in such a manner that, after all the sources in the first row are selected, then the first source of the second row is selected and so on. This method results in the accumulation of the error in both X and Y directions.

The first improvement is the ”symmetrical switching” presented in [153]. The principle of symmetrical switching is presented in Fig. 6.80. A symmetrical switching scheme ensures the cancellation of the linear gradient errors; however, the quadratic errors ere accumulated.

The symmetrical switching scheme is further improved in [154], introducing the ”hierarchical symmetrical switching” scheme (Fig. 6.81), which is designed to reduce the effect of the quadratic error gradient. In Fig. 6.81, the switching sequence in X

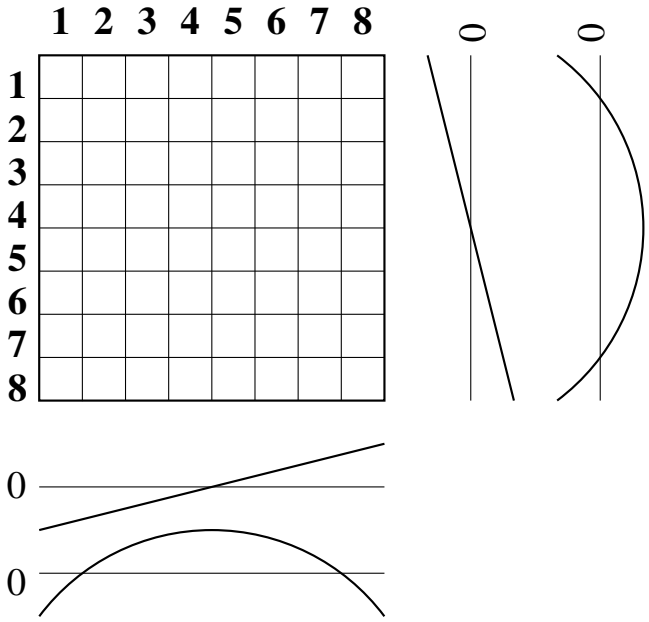

Figure 6.79 Sequential switching.

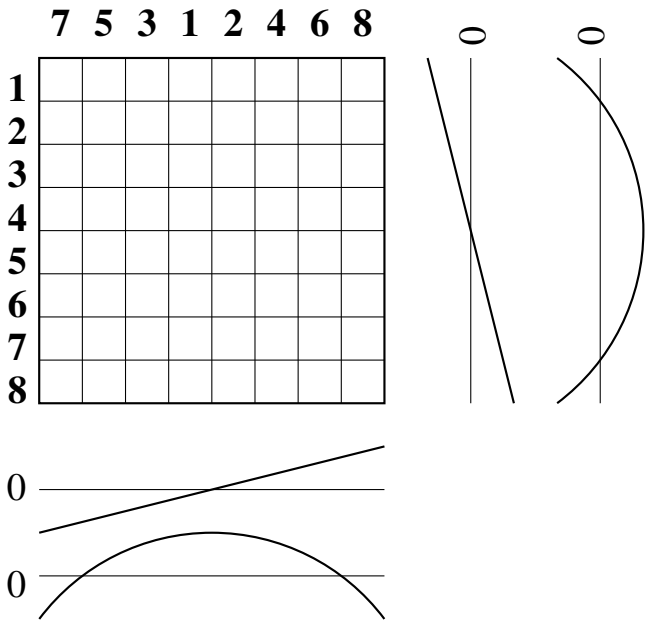

Figure 6.80 Symmetrical switching.

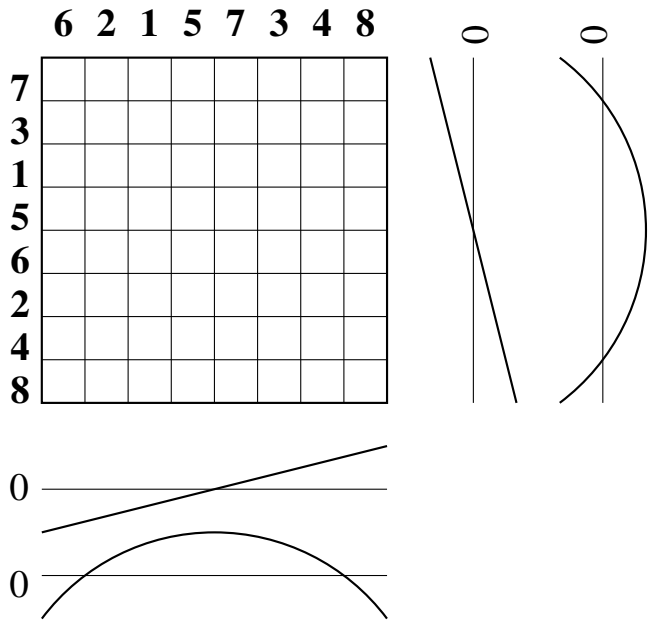

Figure 6.81 Hierarchical symmetrical switching.

direction is ordered according to ”Type A” hierarchical symmetrical switching, and Y direction is ordered according to ”Type B” hierarchical symmetrical switching.

Hierarchical symmetrical switching is further improved in [155]. All introduced switching schemes so far exploit ”Row-Column Decoding”, meaning that all the current sources on the row are selected before the row is changed. It is correctly stated in [155] that this results in an accumulation of the error, since the error builds up column-wise while the error in the direction of rows is canceled. This problem is avoided in [155] by using four identical current source matrices mirrored with respect to X and Y axis. Mirroring is also exploited in [126], [156].

The effect of using mirror symmetric matrices can be analyzed in the following way. Let us begin with the plain current source matrix without any of the mirror symmetry presented in Fig. 6.82. The solid lines represent gradient, dashed lines represent canceled gradient. It is well known that the linear (and odd-order) gradient can be canceled by using common centroid matrices of unity current source transistors, as in Fig. 6.83. Again, the dashed line represents the canceled gradient. This kind of structure, however, does not necessarily cancel or reduce the second- (and even-) order gradient effects. The second (and even) order mismatch can be reduced by further splitting the unity current source and by using the mirror symmetric sub-matrices (Fig. 6.84).

The theory can be generalized as follows. For simplicity it is assumed that the odd-order mismatch has only the first order (linear) component and the even-order

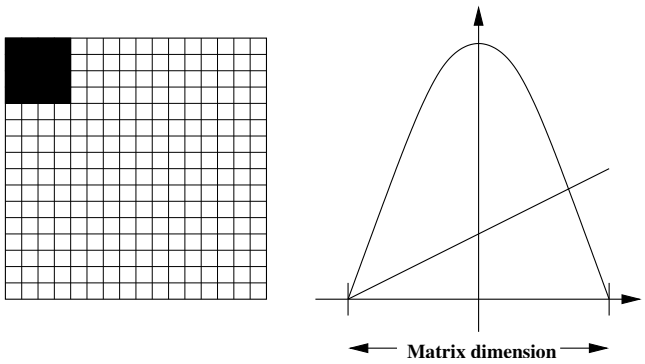

Figure 6.82 Current source matrix with even and odd order error gradient.

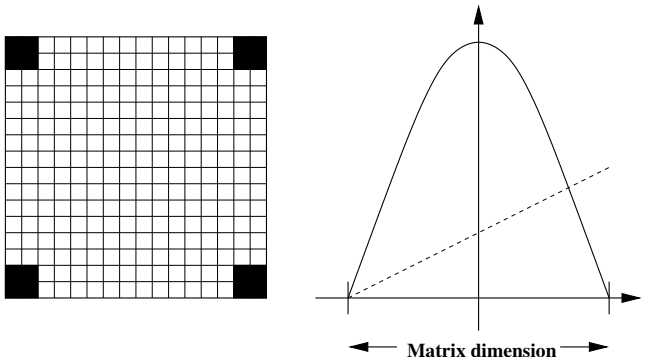

Figure 6.83 Compensation of linear (and odd order) gradient with mirror symmetry.

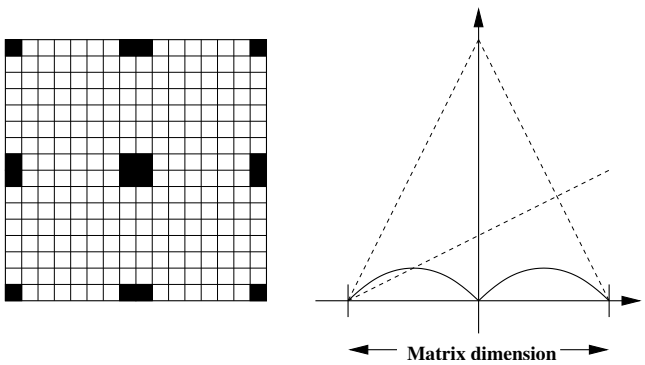

Figure 6.84 Reduction of second (and even order) error gradient by further splitting the matrix.

mismatch has only the second-order component. In addition, we may assume that any constant error has no effect, since it is the same for all the units.

The linear dependency can be removed as in Fig. 6.83, and the residual second order mismatch in X-dimension is defined by the parabola with its extreme at the middle of the chip and zeros at the edges of the chip. The equation for this parabola can be written as

$$E_1(x) = M(x-p_1)(x-p_2). \tag{6.121}$$

It has its extreme at

$$x_{e1} = \frac{(p_1+p_2)}{2}, \tag{6.122}$$

which is exactly in the middle of the zero points. This extreme has the value

$$E_{e1} = \frac{M(p_1-p_2)(p_2-p_1)}{4}. \tag{6.123}$$

The line that goes through the point $(p_1,0)$ and (x_{e1},E_{e1}) has the equation

$$S_{11}(x) = \frac{M(p_1-p_2)(x-p_1)}{2}. \tag{6.124}$$

This linear mismatch is again canceled if the current source is further split and divided into the common centroid matrices that are mirror symmetric relative to the centerline of the original matrix. After Eq. (6.124) is subtracted from Eq. (6.121) the residual error can be written as

$$E_2(x) = M(x-p_1)\left(x-\frac{(p_2+p_1)}{2}\right). \tag{6.125}$$

This is a parabola with zeros at p_1 and $\frac{p_1+p_2}{2}$and the extreme at $x_{e2} = \frac{3p_1+p_2}{4} = \frac{x_{e1}+p_1}{2}$. The extreme value is

$$E_{e2} = \frac{M(p_2-p_1)(p_1-p_2)}{16}. \tag{6.126}$$

It can be seen from Eq. (6.123) and Eq. (6.126) that the ratio of these extreme values is $R_e = \frac{E_{e1}}{E_{e2}} = 4$, so every time the unity source is halved and mirrored in one dimension, the residual error in that dimension is divided by four.

Figure 6.85 represents the cancellation of linear and quadratic gradients by using mirror symmetry. Gradients are modeled as

$$G_x(x) = g_{x1}x\cos(\theta) + g_{x2}x^2,\ -1 \le x \le 1 \tag{6.127}$$

$$G_y(y) = g_{y1}y\sin(\theta) + g_{y2}y^2,\ -1 \le y \le 1, \tag{6.128}$$

in which x and y ranges have been normalized to be ± 1 at the matrix boundary, g_{x1}, g_{x2}, g_{y1},n and g_{y1} are the gain factor for linear and quadratic gradients in x and

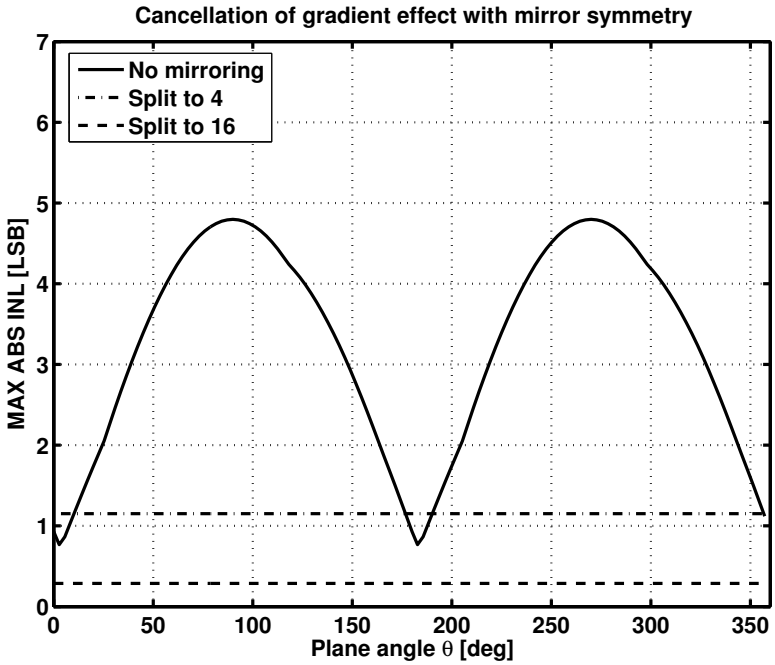

Figure 6.85 Reduction of INL of 16 current sources with mirror symmetry.

y direction, and θ is the plane angle of the linear gradient. The effectiveness of the mirror symmetry in removing the linear gradient can be clearly seen. Additional splitting further reduces the INL due to quadratic gradient by a factor of 4, as calculated.

In the simulation result presented in Fig. 6.85, the plain "sequential switching" scheme was used. The effect of the residual error can be further reduced by using some of the switching schemes discussed before.

In addition to mirror symmetry, the splitting of the current source into smaller units can be used to very effectively reduce the gradient error. In [157], it was presented that if the current sources are divided into 2^2N elements, and if these elements are arranged so that a current source has an element on every row and every column, both linear and quadratic errors are canceled. The main shortcoming of this method is the large number of unit elements. To overcome this problem, a new switching scheme was developed for larger numbers of thermometer-coded bits providing very promising results. The principle of this switching scheme is to translate the positions of 2^{2N} elements to positions of unit current sources of 2N thermometer-coded bits with a simple algorithm developed with numerical optimization to minimize the effect of the quadratic gradient. This algorithm seems to be a very promising solution for the gradient cancellation, and the main advantage of this method is that it does not need any kind of a priori knowledge about the gradients.

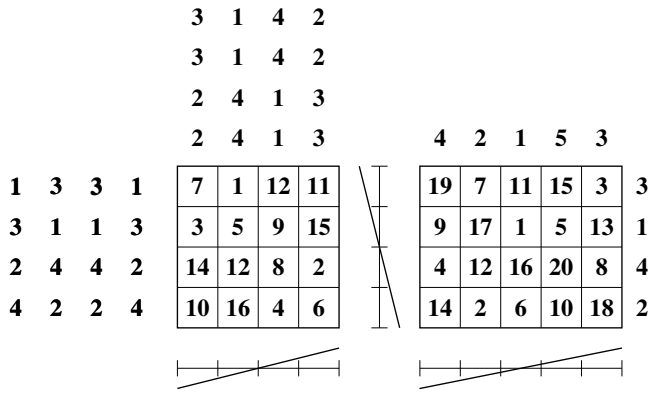

Figure 6.86 Reduction of INL with optimal row-column sequencing.

A rather similar method was developed by the author and used in the prototype presented in this book. The method is based on the ideas of switching sequences presented in [159].

It is possible to minimize the maximum value of accumulated linear error of the unit elements in dimension (X or X) by selecting them in certain order. Let us call this an “”optimal sequence”. The principle of finding optimal sequences was presented in [159]. As mentioned earlier, the problem with the ”Row-Column Decoding” is the accumulation of the error in one dimension while it is canceled in the other. This problem can be alleviated by applying optimal sequences in the both dimensions simultaneously (this kind of result is also obtained by applying the technique presented in [157]). Two examples are given in Fig. 6.86. The number sequences around the matrices indicate the order of selection and the numbers inside the matrices indicate the number of the source, i.e. source number 1 is placed first into the position indicated by numbers 1,1 of sequences etc.

This technique has at least two shortcomings. First, if the number of rows and columns are not relatively prime, the number of positions selectable with one sequence pair is defined by the length of the longer sequence, after which a new optimal sequence has to be selected as was done in the case of 4x4 matrix in Fig. 6.86. The newly selected optimal sequence should also be valid in such a manner that already-occupied positions are not selected again. It can be seen that in 4x4 case, there are too few optimal sequences, so suboptimal sequences have to be used. On the other hand, there is no problem if the number of rows and columns are relatively prime, as in the case of the 4x5 matrix of Fig. 6.86.

Usually, the number of current sources (including biasing) is 2^N in the matrix, and therefore it may become difficult to find the optimal sequences, since 2^N cannot be presented as the product of relatively prime numbers. If, for some reason, such as in

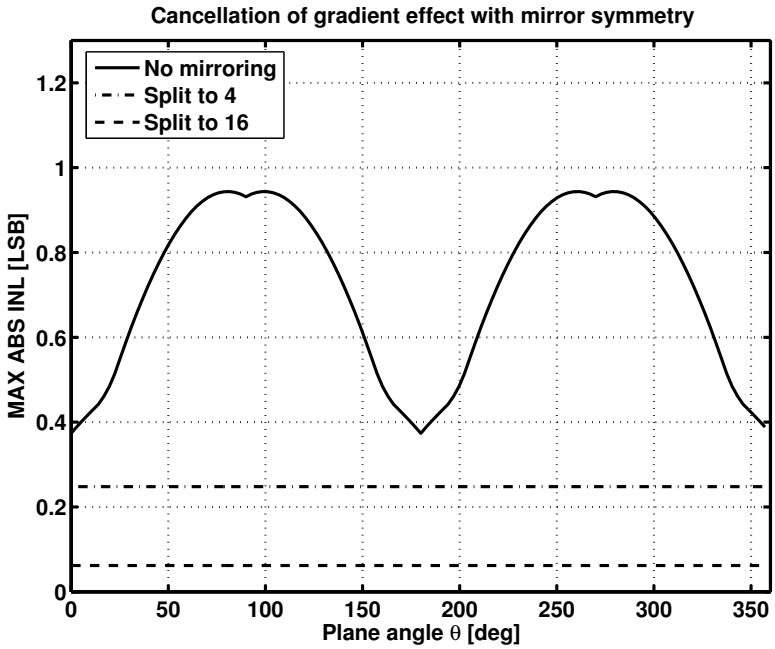

Figure 6.87 INL and effect of splitting with developed switching sequence.

the case of digital calibration, or due to including the LSB sources into the matrix, the number of sources differ from 2^N, the situation can be easier to handle. Simulation results for this sequence in the case of 4x4 matrix is presented in Fig. 6.87. Its effectiveness can be verified by comparing Fig. 6.87 to Fig. 6.85.

The methods of the layout design of binary-weighted converters are seldom discussed in the papers. Due to the binary weighting, the number of unit sources is large for currents of larger weight, resulting in almost automatic cancellation of gradient errors. Two examples are given in Fig. 6.88. The basic principle is to fill every other empty position, starting from the opposite corners of the matrix simultaneously. The filling of a row is started from side opposite the previous row was started. The same holds for the sources of different weights.

6.9 Survey of published D/A converters

In this section, the key features and design methods of the recently published current steering D/A converters are briefly introduced. The figures of merit of the converters are listed in Table 6.3.

In [153], the principle of symmetrical switching was introduced and used to alleviate the process-gradient-related mismatch of the current sources.

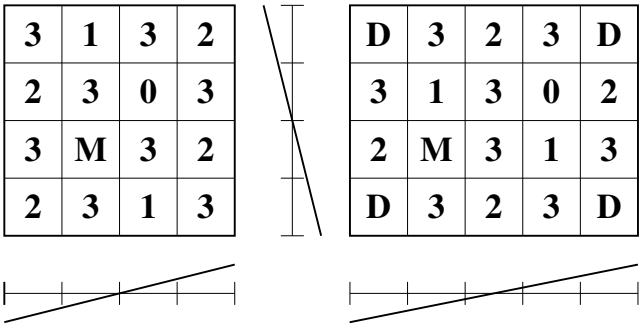

Figure 6.88 Two possible current source matrices for 4-bit binary weighted D/A converter.

In [154], the switching method was further improved from symmetrical switching to hierarchical symmetrical switching.

In [160], a switch structure is introduced to improve the switch driver timing, reduce the switching asymmetry and code dependent glitching at the output. Segmentation is mentioned as a technique for glitch reduction.

In [126], the development of the switching schemes continued with four quadrant randomization technique. *INL* and *DNL* are analyzed as a function of the segmentation, and the estimates of the standard deviations of *INL* and *DNL* are given based on Monte-Carlo simulations. It is mentioned that glitches that are linearly dependent on the code transition do not introduce distortion. The effect of segmentation is emphasized, and the segmentation level is optimized in order to optimize *DNL*, *INL*, *SFDR* and area. The cascode current sources are also used.

In [155], the switching sequence optimization is continued by the introduction of a hierarchical symmetrical switching scheme using common centroid matrices. Linear and quadratic error gradients are discussed. A Gaussian distribution is used as an approximation for *INL*. Glitches due to switch driver feed-through are minimized. Segmentation in two thermometer-coded segments with different weights is also used.

In [161] the return-to-zero output stage is used to reduce the effect of glitches, code-dependent settling and timing skew between current sources.

In [158], the linearity errors due to current source mismatches are further optimized with the Q^2-random walk switching scheme. A dynamic latch is used for the switch driver synchronization and the crossing point of the switch driver signals is optimized. The effect of linear and quadratic gradients are analyzed.

In [162], a self-trimming D/A converter with a track-and-attenuate output stage is presented. Self-trimming is based on the error value stored on the gate capacitance of the current source, and the error is measured with a $\Sigma\Delta$ analog-to-digital converter (ADC) over the resistance in series with the current source transistor.

In [163], a converter capable of 1 GS/s operation is presented. The effect of output impedance is taken into account, and the effort is put on the design of the high-speed switch driver. A double-centroid switching scheme (mirror symmetry) is used to reduce gradient effects.

In [164], as in the previous design, the effort is put on the switch driver optimization and signal symmetry in order to achieve high-speed operation. A triple centroid switching scheme (16 units in one current source + mirror symmetry) is used to improve the static matching.

In [137], the static accuracy is improved by calibration based on trimmable current sources, which are adjusted with an additional calibration DAC.

In [138], the static performance is improved with calibration. The static nonlinearity is measured with a 16-bit $\Sigma\Delta$ calibration ADC; values are stored in the memory and addressed with the 6 MSB bits. An additional calibration DAC is used to adjust the currents of the MSB sources.

In [139], a successive approximation calibration routine is used to trim the MSB current sources to 18-b accuracy. The effect of the nonlinear capacitance at the source node of the switches is emphasized, and back-gate-buffering (bulk-bootstrapping of the switches) is used to reduce the effect of the nonlinear capacitance. Switch driver timing is also emphasized, and the timing inaccuracies in switching are reduced by using a local voltage generator that follows the source voltage of the switches. Also, a differential-quad switching scheme [151] is applied in order to equalize the switching activity, and thus the charge flow from the supply lines.

In [165], trimming of the floating-gate current source transistors is used to improve the static nonlinearity. In addition, a return-to-zero output stage is used to improve dynamic performance.

In [166], dynamic element matching is used to average the current source array errors. 7-bit segmentation is used in order to improve dynamic performance and MSB current sources are composed of 16 unit cells, which are divided into 16 current source arrays in order to reduce the effect of process gradients.

In [167], capacitive current memory type of calibration is used to trim the MSB current sources. An RZ stage is used at the output to improve the dynamic performance. The dynamic performance is also compared to case in which the RZ-stage is disabled, indicating that with this particular converter, better high-frequency operation is obtained with the RZ output stage.

In [168], the importance of the switching dynamics is emphasized. The differential-quad switching scheme is applied and kick-back (variation of the source node voltage of the switch) is minimized with switch-driver crossing-point control.

In [146], attention is paid to the reduction of timing errors, and power supply and bias disturbances targeting the elimination of the nonlinearity instead of removing it

from the output. Current-mode logic is used to reduce the supply-rail and substrate cross talk. The timing of the switch drivers is equalized with a replica structure similar to as the prototype presented in Section 7.2.

Table 6.3 Survey of published D/A converter implementations.

Publication	Process	Bits	Sampl. freq. [MHz]	SFDR [dBc] @ [MHz]	Area [mm^2]	Power [mW]	Comment
Miki [153]	2μm CMOS	8	80	?	3.79	145	
Nakamura [154]	1μm CMOS	10	70	?	3.78	170	
Mercer [160]	2μm Bi-CMOS	16	40	80 @ 1.23	8.25	500	SFDR @ 10MS/s
Lin [126]	0.35μm CMOS	10	500	51 @ 240	0.6	125	
Bastos [155]	0.5μm CMOS	12	300	40 @ 60	3.2	320	
Bugeja1 [161]	0.5μm CMOS	14	100	74 @ 8.5	14.42	750	
Van der Plas [158]	0.5μm CMOS	14	150	61 @ 5	13.10	300	
Bugeja2 [162]	0.35μm CMOS	14	100	72 @ 42.5	11.83	180	
Van den Bosch1 [164]	0.35μm CMOS	10	1000	61.2 @ 490	0.35	110	Core area
Van den Bosch2 [163]	0.25μm CMOS	12	500	62 @ 125	7.14	102	
Tiilikainen [137]	0.18μm CMOS	14	100	64 @ 1	1.0	20	
Cong [138]	0.13μm CMOS	14	180	50 @ 63	0.1	16.7	Power @100MS/s, core area
Schofield [139]	0.25μm CMOS	16	400	73 @ 190	?	400	
Hyde [165]	0.18μm CMOS	14	300	71 @ 120	0.44	53	Power @250MS/s, core area
O'Sullivan [166]	0.18μm CMOS	12	320	60 @ 60	0.44	82	
Huang1 [167]	0.18μm CMOS	14	200	60 @ 90	1.0	97	RZ-mode, core area
Huang2 [167]	0.18μm CMOS	14	200	43 @ 90	1.0	97	NRZ-mode, core area
Schafferer [168]	0.18μm CMOS	14	1400	67 @ 260	6.25	400	
Doris [146]	0.18μm CMOS	12	500	60 @ 220	1.13	216	Core area

Chapter 7

Prototypes and experimental results

In this chapter, the methods used in the WCDMA prototype circuit design are described and the measurement arrangements and the experimental results obtained from the fabricated transmitter chip are reported.

The target of the design process was to advance our knowledge of the efficient realization of the DSP algorithms, discover the system limitations set by the digital-to-analog converter, learn more about the D/A converter design and apply the knowledge gained to the design of the IF DSP part of the transmitter to see how the different parameters affect the transmitter performance. The performance of the transmitter is characterized by EVM and $ACLR$.

7.1 Prototype of WCDMA transmitter

The principle of the designed WCDMA transmitter is presented in Fig. 7.1. The transmitter consists of pulse shaping filtering and interpolator blocks for four independent I and Q data streams in order to alter the sampling rate from the input symbol rate to the sampling rate used in the CORDIC vector-rotation-algorithm-based quadrature modulators. After the modulation, the modulated carriers are summed and fed to the inverse SINC predistortion filter [169]. The purpose of the filter is to eliminate the SINC attenuation caused by the sample-and-hold function of the D/A converter. Two constant scalers are used to maximize the signal dynamics before feeding it to the D/A converter. The values of the scalers are discovered with simulations. Finally, the data is fed to the D/A converter. A 14-bit current steering architecture is chosen because of its capability in terms of high-speed operation and good static linearity. The design of

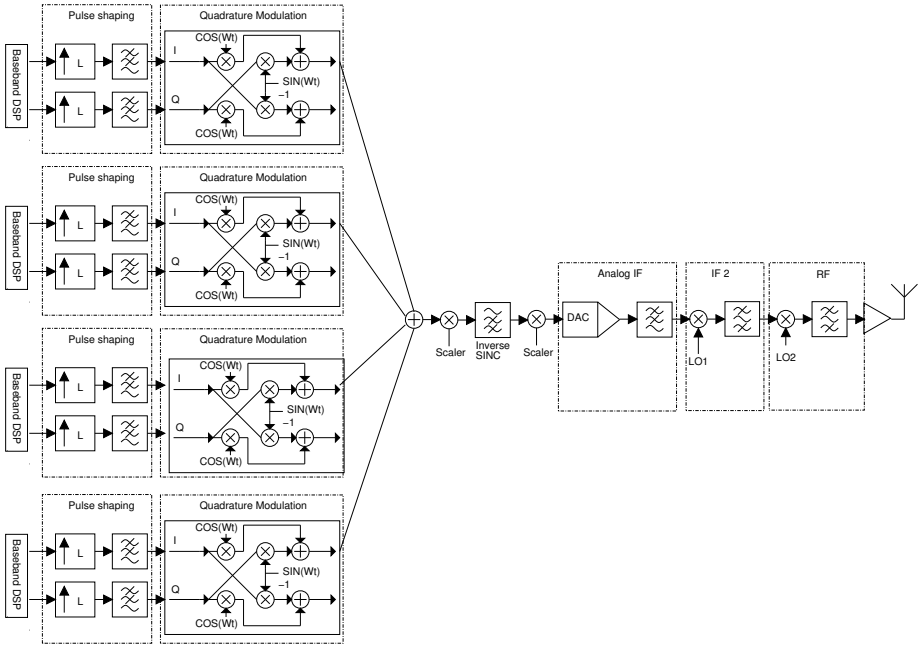

Figure 7.1 Multicarrier transmitter.

Table 7.1 Design parameters of the system.

Symbol rate	3.84 Msymbols/s
Pulse shaping filter	Root-raised cosine $\alpha = 0.22$
Number of carriers	4
Assumed input signal statistics	Normally distributed and truncated with crest factor 10 dB
Signal bandwidth	3.84MHz
Carrier spacing	5MHz
EVM	Tested with 4-QAM signal with symbol magnitude 40dB below maximum
$ACLR_{123}$	Simulated with channel separation of 5, 10 and 15 MHz respectively
Number of input bits	13

the IF filters, mixers and PA are beyond of the scope of this work.

The parameters and signal conditions used in the transmitter design are listed in Table 7.1.

7.1.1 Frequency planning

The design begins with the frequency planning. The number of carriers and carrier spacing of 5MHz defines the total bandwidth that needs to be 19.68MHz. However, in order to make the image filtering possible after the D/A conversion and after the upconversion to the second IF, the frequency band used is 5.16-24.84MHz. To get the first image far enough to be filtered, the sampling frequency should be selected high enough. The lowest integer multiple of the input symbol frequency 3.84MHz that is high enough to make the filtering possible is 61.44MHz, which is selected to be the clock frequency of the CORDIC rotator, resulting in the total interpolation ratio of 16 for the input data.

7.1.2 Interpolation strategy

In order to be able to select the most effective strategy for the interpolation, the total number of filter coefficients must be determined. The number of FIR filter coefficients required for the PSF with certain stop-band and pass-band ripples and relative transition bandwidth can be approximated with the equation that holds for the equiripple low-pass FIR filter [58]

$$N_1 \approx \frac{-20\log_{10}\left(\sqrt{\sigma_p\sigma_s}\right)-13}{14.6\Delta f}+1 = K\frac{L_1F_s}{f_s-f_p}+1 \quad L_1>1 \tag{7.1}$$

in which $K=-20\log_{10}\left(\sqrt{\sigma_p\sigma_s}\right)-13$, F_s is the sampling rate at the input of the PSF, L_1 is the oversampling ratio of the PSF, σ_p and σ_s are the pass-band and stop-band ripples, respectively, $\Delta f=\frac{f_s-f_p}{L_1F_s}$ is the width of the transition band relative to the out sampling frequency of the PSF, and f_s and f_p are the stop- and pass-band edge frequencies, respectively.

When a chain of interpolation filters is used to perform the possible residual interpolation after the PSF, the number of taps required for each of these interpolation stages may be calculated as

$$N_k \approx K\frac{L_kL_{pk}F_s}{L_{pk}F_s-2f_s}+1 = K\frac{L_k}{1-\frac{2f_s}{L_{pk}F_s}}+1 \quad L_k>1, k>1 \tag{7.2}$$

in which L_k is the interpolation ratio of the current stage, and $L_{pk}=\prod_{i=1}^{k-1}L_i$. With equations (7.1) and (7.2), the total number of coefficients for each interpolation strategy can be calculated. It can be observed that the total number of filter coefficients is minimized either when the interpolation is made in four cascaded stages, each interpolating by factor two, or with the cascaded interpolation stages with interpolation factors 2, 2, and 4, respectively. Comparison of the efficiency of the interpolation strategies is

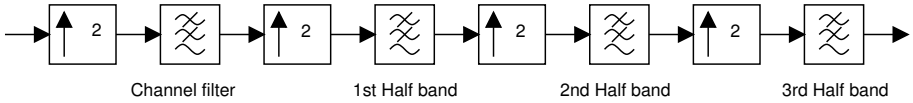

Figure 7.2 Interpolation chain.

presented in Table 7.2. The four cascaded filter stages with an interpolation factor of 2

Table 7.2 Comparison of interpolation strategies

Order of interpolators	Number of coefficients per stage	Total number of coefficients
16	297	297
8, 2	149, 11	160
4, 4	74, 25	100
4, 2, 2	74, 13 ,11	99
2, 8	38, 84	122
2, 4, 2	38, 43, 11	92
2, 2, 4	38, 22, 24	84
2, 2, 2, 2	38, 22, 13, 11	84

is chosen because it facilitates the use of half-hand (HB) filters after the PSF. The use of the HB filters further reduces the amount of hardware because approximately half of their coefficients are zero valued. The structure of the interpolation chain is presented in Fig. 7.2.

7.1.3 Digital FIR filter and interpolator design

The multi-step interpolation with an interpolation factor of 2 was chosen as an interpolation strategy because of its hardware efficiency, After the decision on the interpolation strategy, the floating point prototypes of the channel filters and the half band interpolation filters were designed. The channel filter was designed with the Lagrange algorithm described in Section 4.1.1. The half band filters were designed with the least square error algorithm and with the “trick”-method described in [62]. With this method, an even-order prototype filter with desired pass-band characteristics and a transmission zero at half of the sampling frequency is converted to an odd-order half-band filter by inserting a zero between the coefficients of the prototype and changing the centermost zero to 1. In this design phase, some margin was left for the degradation of the performance caused by CSD presentation and finite word length effects in order to ensure that the performance of the transmitter can be designed to be limited by the D/A converter.

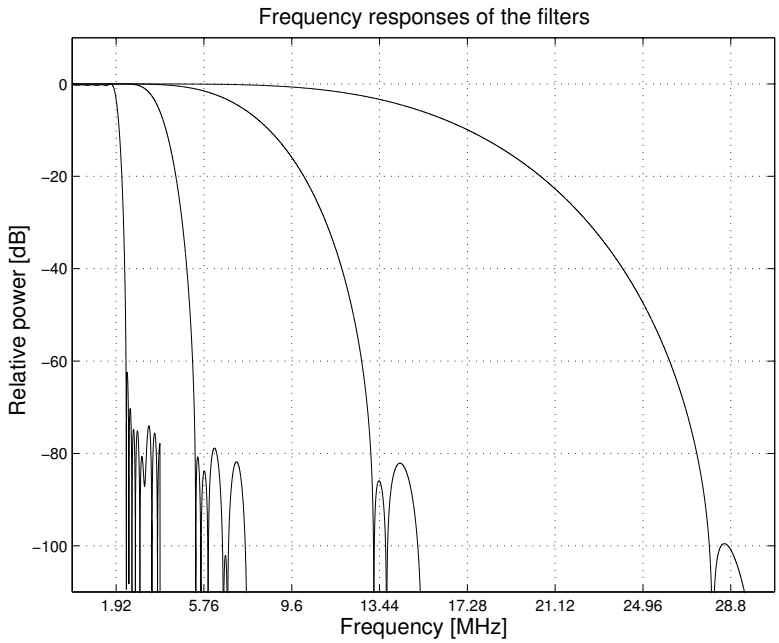

Figure 7.3 Frequency responses of the filters.

Once the coefficients of the prototype filters were discovered, the coefficients were converted to CSD format with the modification of the algorithm presented in [67]. The algorithm was modified in order to maximize *ACLR* rather than to minimize the peak ripple on the stop-band.

The frequency responses of the designed filters are presented in Fig. 7.3 and the frequency response of the interpolation chain is presented in Fig. 7.4. ISI_{rms} of the designed CSD interpolation filter chain is -45.022dB (0.561%) and *ACLR* values for the first, second and the third adjacent channel are 85.89dB, 99.13dB and 92.15dB, respectively. The quantization noise due to the D/A conversion or finite word length effects is not included at this point.

The lower limit of EVM is set by ISI of the channel filter. The finite accuracy of the computation in the filters and CORDIC modulator, and the quantization in the D/A converter adds noise to the signal thus increasing EVM. Similarly, the limit of *ACLR* is set by the D/A converter, the stop-band attenuation of the filters and the noise added by the filters and CORDIC modulator. Since the 14-bit D/A converter sets the *ACLR* value approximately to 75dB, the objective of the design was to obtain EVM_{rms} performance better than -40dB (1%) and *ACLR* to be limited by the D/A converter.

After the CSD coefficients were discovered, the analysis of the finite word length effects was performed. The simulation script was written in Matlab that modeled the

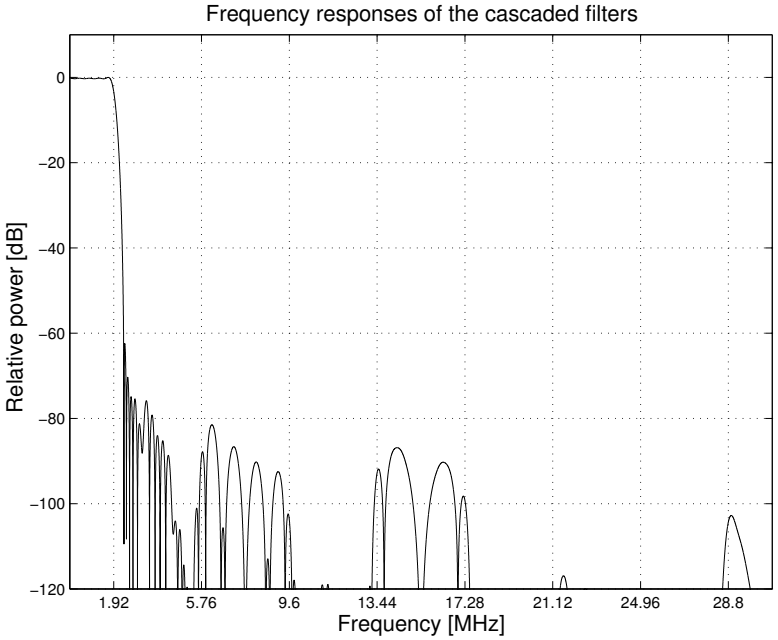

Figure 7.4 Frequency response of the interpolation filter chain.

behavior of the transmitter. The effects of the CORDIC and inverse SINC filter were left out at this point, and an ideal upconversion and reception was used. The D/A converter was modeled as a quantization at the transmitter output. A normally distributed and truncated data stream with the crest factor of 10dB was used as the input in *ACLR* simulations in order to model the sum of 4 QAM data of 100 code channels (central limit theorem [19]). In EVM simulations, the single 4 QAM modulated signal with a symbol magnitude of 40dB below the maximum was used. The internal word length of one filter was swept from 10 to 20 bits when the internal word lengths of the other filters were kept at 20 bits. With this method, the effects of the one single filter to *ACLR* and EVM were discovered. In Fig. 7.5, the effects of the internal word lengths and the effect of the D/A converter on $ACLR_1$ are presented. The effects on EVM are presented in Fig. 7.6. Similar simulations were carried out for $ACLR_2$ and $ACLR_3$. The internal word lengths chosen for the channel filter (pulse shaping filter) and for the first, second and third half-band filters are 17, 18, 18, and 18 respectively. The simulated $ACLR_{123}$ values with the finite word length effects included in the filters are 73.92dB, 75.62dB and 73.70dB, respectively, and the EVM_{rms} is -40.5398dB (0.94%).

The polyphase structure that was described in section 4.3.1 was used in the realization of the filters. In order to further reduce the area used by the filters, the interleaving technique described in Section 4.3.2.3 was used [71]. In addition to the area reduction,

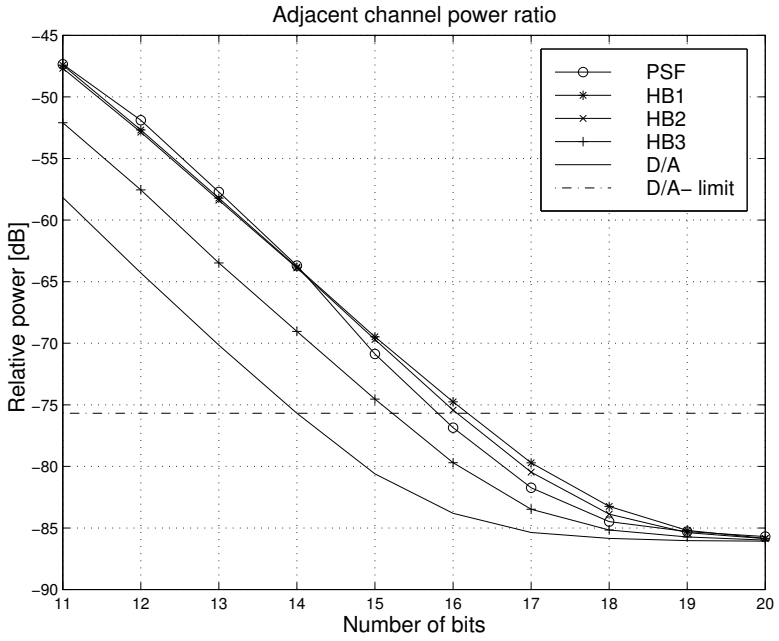

Figure 7.5 $ACLR_1$ as a function of the internal word lengths.

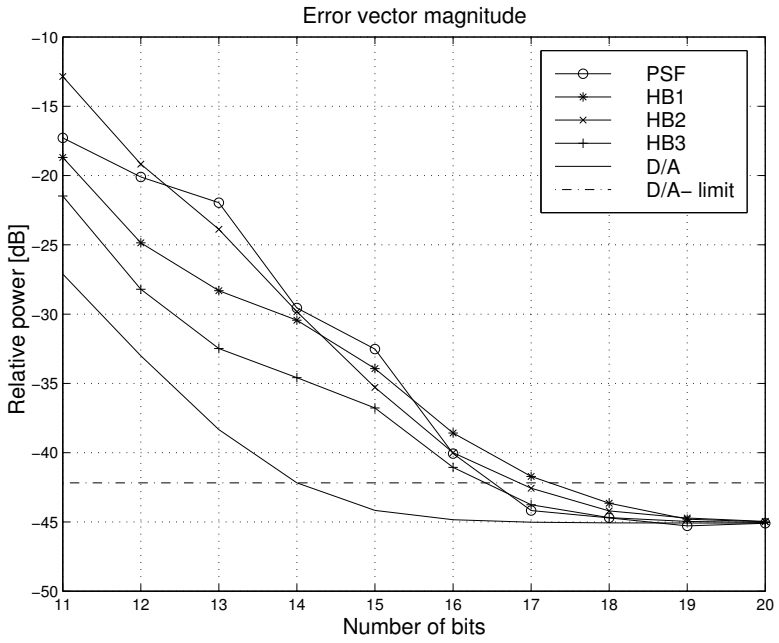

Figure 7.6 EVM_{rms} as a function of internal word lengths and D/A converter resolution.

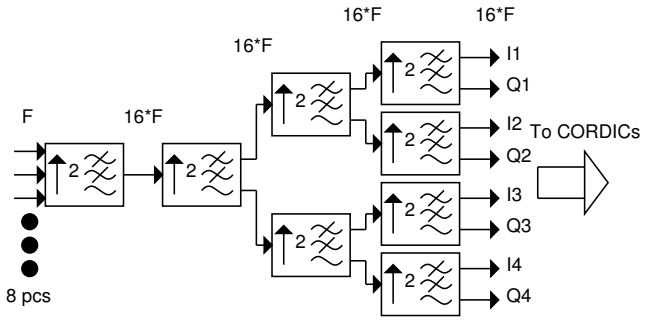

Figure 7.7 Pipelined/Interleaved interpolation filter chain.

Table 7.3 Number of filter coefficients and the number of realized coefficients in polyphase decomposition.

	Channel filter	1st half-band	2nd half-band	3rd half-band
Number of coefficients	37	23	11	11
Maximum number of terms in CSD coefficient	6	5	4	4
Maximum shift in CSD coefficient	13	13	12	12
Number of realized coefficients in polyphase composition (subfilter 1/subfilter 2)	9/10	6/1	3/1	3/1
Internal word length	17	18	18	18

the advantage of using the interleaving technique is that all the clock frequencies used in the filters are above the desired signal frequency band, reducing the possible spurs generated by substrate coupling. The clock frequencies were selected so that the maximum clock frequency equals the one used in the CORDIC rotator, 61.44MHz. The first filter, however, uses a clock of only 30.72MHz because there are only 8 data streams to interleave and because the polyphase structure allows the computation at half of the sampling frequency (sampling frequency after the interpolation). The structure of the interleaved interpolation filter chain is presented in Fig. 7.7. The number of filter coefficients and realized filter coefficients in the polyphase composition are listed in Table 7.3.

The simulated $ACLR$ and ISI_{rms} and EVM_{rms} values for the CSD-filters without and with the finite word length effects are listed in Table 7.4.

Table 7.4 *ACLR* and ISI_{rms}/EVM_{rms} values.

	$ACLR_1$	$ACLR_2$	$ACLR_3$	ISI_{rms} and EVM_{rms}
Without internal word length effects	85.89dB	99.13dB	92.15dB	-45.022dB (0.561%)
With internal word length effects	73.92dB	75.62dB	73.70dB	-40.5398dB (0.94%)

7.1.4 CORDIC and inverse-SINC filter design

7.1.4.1 CORDIC design

The word lengths of the CORDIC and the inverse SINC filter were determined in a manner similar to that in the interpolation filter chain. The output of the interpolation filter chain was used as the input of the CORDIC rotator and the number of stages ($N = 15$) (Fig. 5.4), number of bits in the amplitude path ($a = 20$) and the number of bits on the phase calculation path ($p = 17$) were discovered.

According to the theory presented in Section 5.2, the maximum output value of the CORDIC rotator is $A_{outmax} = 1.6468\sqrt{2} = 2.3281$, assuming the input values to be 1. However, when summing up four modulated wide band signals, it is very unlikely that the signal has the value $4A_{outmax}$. It can be calculated as in [170], that the I/Q modulated signal with Gaussian distributed I and Q data values and standard deviation σ is also Gaussian distributed with standard deviation σ, and thus the multicarrier signal also has a Gaussian distribution with variance linearly dependent on the number of carriers.

The simulated probability density of the sum of 1, 2, 3, and 4 wide-band signals are presented in Fig. 7.8, and the probability density function of the Gaussian distribution with corresponding standard deviations are presented in Fig. 7.9

Since the ratio of the variance (i.e. power) and maximum amplitude does not change as the number of carrier increases, the consequence is that the power of a single carrier is reduced relative to quantization noise, thus increasing *EVM* and *ACLR*. Also, because the probability of overflow in the case of a limited number of bits is relative to the standard deviation of the signal, the number of bits should be increased as a function of standard deviation ($\sqrt{N}, N =$ Number of carriers) instead of theoretical maximum amplitude, in order to maintain the probability of overflows.

One method to handle the occasional overflows and reduce the overflow originated signal degradation is clipping. To clip the large values of the multicarrier signal, the saturating scaler is added to the output of the CORDIC. In the case when the peak value of the signal is just slightly above the half of the full scale, approximately 6dB of the dynamic range is lost when the signal is truncated. Scaler maximizes the dynamic range of the signal and saturates the output in the cases of overflow. This also minimizes the amount of hardware needed in the inverse SINC filter, since the internal

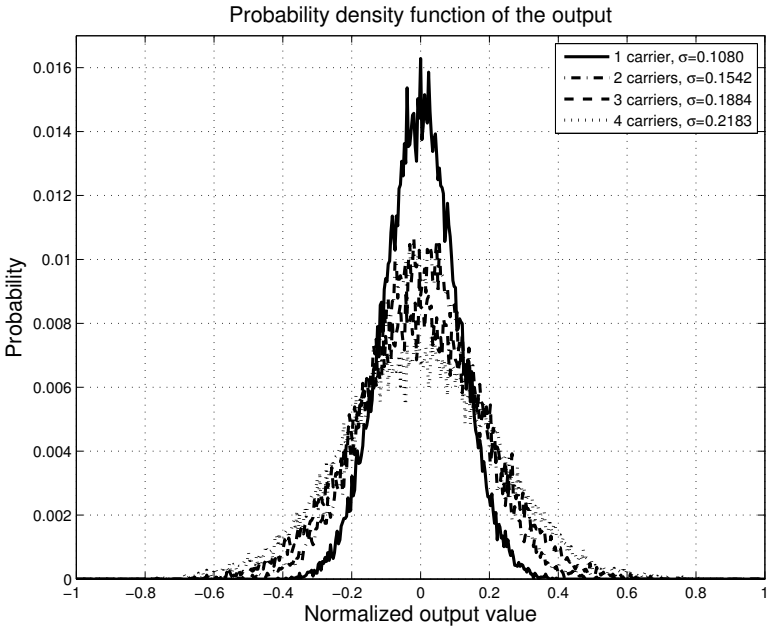

Figure 7.8 Probability density function of simulated multicarrier signals.

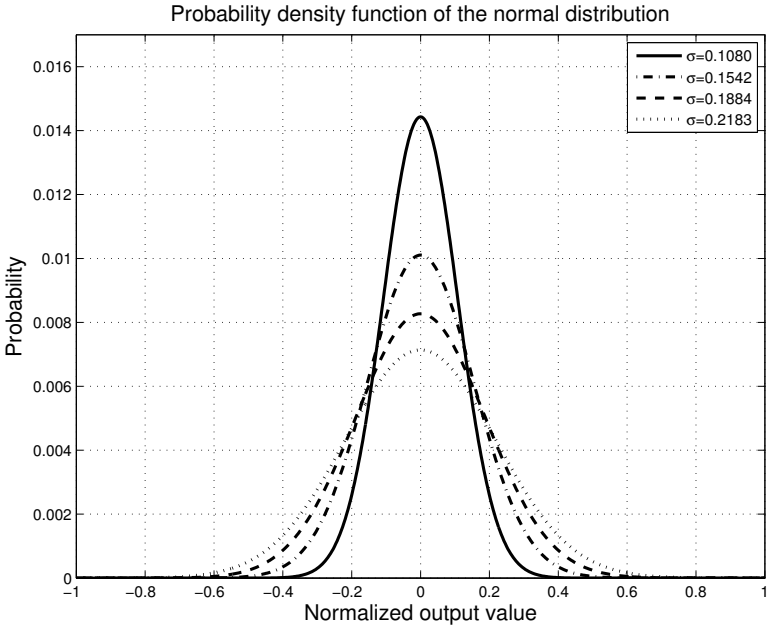

Figure 7.9 Probability density functions of normal distribution with standard deviations of simulated multicarrier signals.

wordlength of the filter is reduced. The value of the scaler is determined by simulations. The effects of the clipping and various algorithms for clipping the wide band signals are discussed in more detail for example in [171].

7.1.4.2 Inverse-*SINC* filter design

The inverse-*SINC* predistortion filter is used to compensate the droop in the frequency response of the transmitter caused by the sample and hold operation of the D/A- converter. This droop is -2.476dB at 24.84MHz which is not acceptable.

The inverse-*SINC* filter was designed by the method described in [169]. It has 7 coefficients which are symmetric relative to the center coefficient. The transposed direct form is used in the realization of the filter. After correction, the gain ripple at the band from 5.16 to 24.84MHz is 0.0895dB. The simulated internal word length for the correction filter is 18 bits. A scaler similar to one after the CORDIC is used at the output of the filter in order to maximize the signal dynamics. After scaling, the signal is rounded to 14 bits and fed to the D/A converter.

7.1.5 Simulated QAM performance

After the internal resolution and the values of the scalers had been defined, the overall performance of the multicarrier QAM modulator was determined by simulation. The final *EVM* value is -40.38dB (0.96%), and the final $ACLR_{123}$ values are 73.58dB, 75.58dB, and 73.65dB, respectively. The simulated spectrum of the multicarrier signal at the transmitter output is presented in Fig. 7.10. The single carrier case is presented in Fig. 7.11.

7.1.6 Digital ASIC synthesis flow

The digital part of the QAM modulator was realized with logic synthesis based on high level description language. The synthesis flow is presented in Fig. 7.12.

Tools that was used for the synthesis were Synopsys Design Analyzer for the logic synthesis and static timing verification, Synopsys VSS for VHDL simulations, Synopsys Formality for static functional verification, Cadence Envisia Design Planner and Silicon Ensemble for floorplanning, Cadence Envisia Silicon Ensemble for place and route, Cadence Envisia PB-opt and CT-gen for placement-based optimization and clock-tree generation and Mentor Graphics IC-Station for top-level manual editing, layout versus schematic checking and design-rule checking.

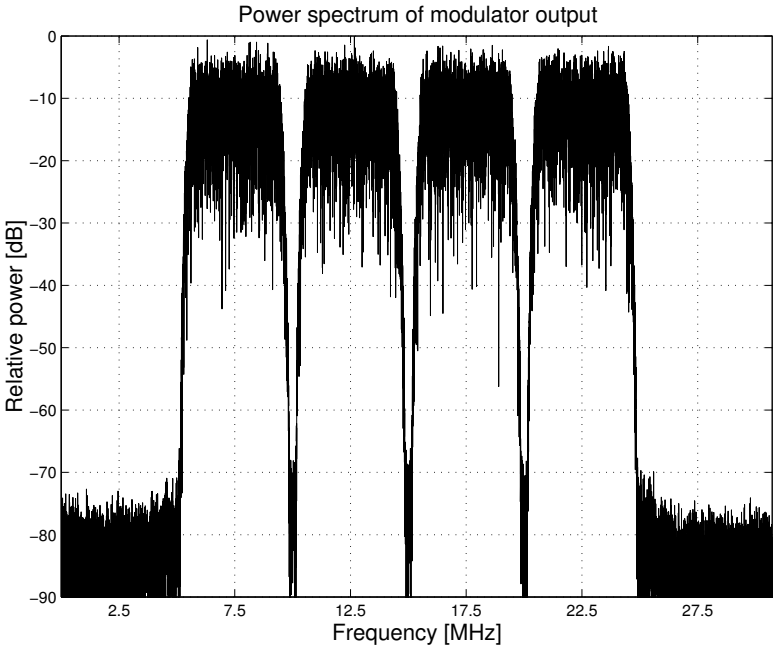

Figure 7.10 Spectrum of the multicarrier WCDMA signal at the output of the transmitter.

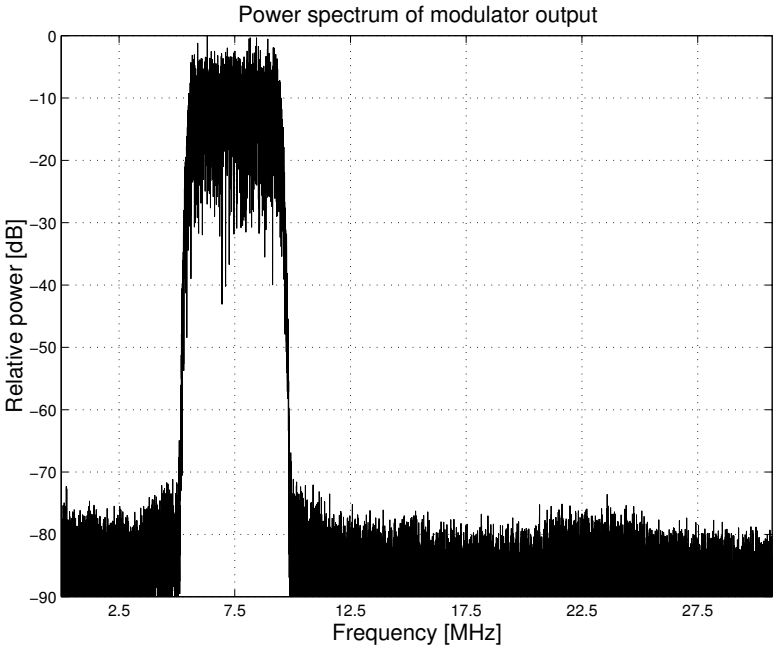

Figure 7.11 Spectrum of the single carrier WCDMA signal.

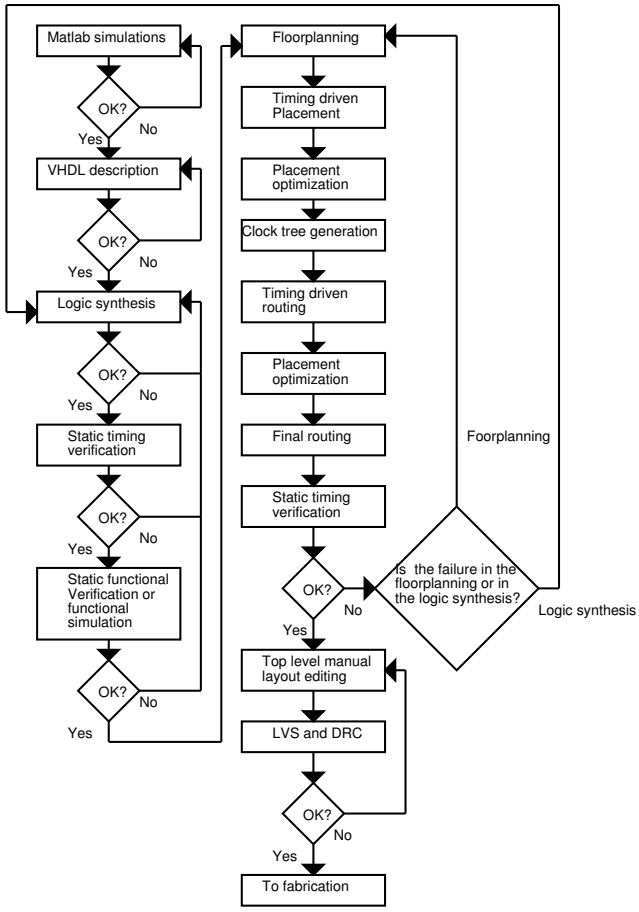

Figure 7.12 Logic synthesis flow

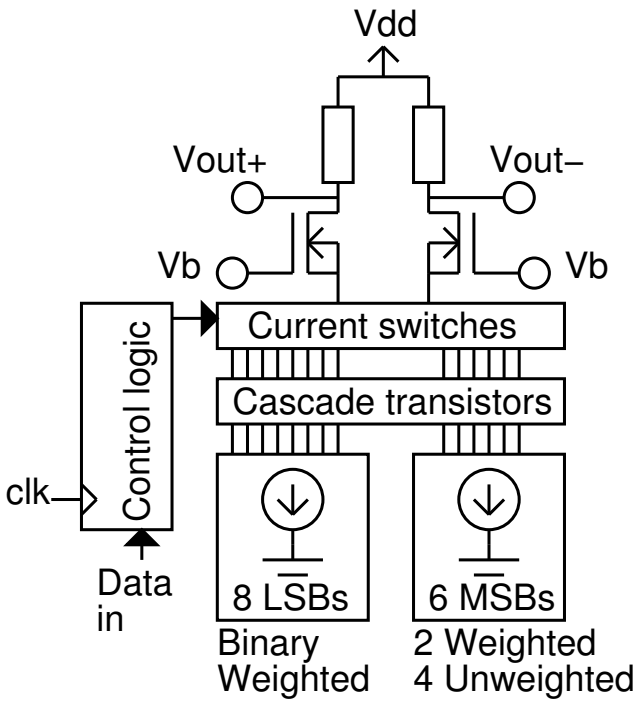

Figure 7.13 Block diagram of the D/A-converter.

7.1.7 14-bit 70MHz Digital-to-analog converter

In this section, only the theory applied to this particular design is discussed. In case a more accurate analysis of the design methods is performed after the design of this prototype, references are given.

The D/A converter designed for the transmitter is a segmented current steering type converter. It has two current source matrices. The matrix of 8 LSBs is binary weighted and the matrix for the 6 MSBs is partially weighted. Two least significant bits of the MSB matrix are binary weighted and four MSB bits are thermometer coded (unweighted). Each of the fifteen MSB current sources corresponding to the four thermometer coded MSB bits provides a current of four times the least significant bit of the MSB matrix. The topology of the D/A converter is presented in Fig. 7.13.

7.1.7.1 Static matching

It was assumed as a coarse approximation that *INL* follows the normal distribution with variance

$$\sigma_{INLmax} = \sqrt{2^N - 1}\frac{\sigma_{lsb}}{I_{lsb}}. \tag{7.3}$$

The analysis of *INL* presented in Section 6.3 reveals that this assumption is not absolutely accurate, but gives a good starting point for the dimensioning of the current sources.

Method for finding the dimensions of the current sources is presented in Section 7.2.1. The area of the LSB current source is obtained by using the matching equations presented in [125], [124]

$$\frac{\sigma_{id}^2}{I_d^2} = \frac{4A_{vt}^2}{WL(V_{gs}-V_T)^2} + \left(\frac{A_{beta}}{\sqrt{WL}} + B_{beta}\right)^2, \tag{7.4}$$

where A_{vt}, A_{beta} and B_{beta} are process dependent constants given by the silicon vendor, V_{gs} is the gate-source voltage of the transistor, V_T is the threshold voltage of the transistor, W is the channel width and L is the channel length.

The area of the LSB current source required for 14-bit matching, as determined by equations Eq. (7.3) and Eq. (7.4), was too large to be realized. Therefore the area used for the current source was determined in order to obtain 12-bit linearity (*INL* and *DNL* < 2LSB).

In order to increase output impedance of a single current branch, additional cascode transistors were added between the load and the current switches. These transistors were made common to all of the switches connected to the particular signal branch. Cascode transistors are also used between the current switch and the current source transistor to further increase the output impedance of the current sources. Analysis of the behavior of the output impedance is presented in more detail in Sections 6.5 and 7.2. The topology of the one current source, switch and output cascode set is presented in Fig. 7.14.

7.1.7.2 Enhancement of dynamic properties

In order to minimize the capacitive coupling, swing as small as possible is used to drive the switches. It is also ensured that both of the switches are not closed at the same time, which would lead to the situation where no current flows through the current source. In this case, the sources of the switch transistors would be driven to V_{ss}. This would cause a spike at the output, and the current sources would require a relative long time to recover from the bounce. The switch driving principle is presented in Fig. 7.15.

Inaccuracies in synchronization of the switch drivers may cause harmonic distortion. Because the currents in LSB and MSB matrices are different, the switches are also of different size, which minimizes the difference between bias points and thus improves the static matching. The synchronization of switching is enhanced by adding dummy transistors in parallel with the gates of the smaller LSB switches in order to equalize the load seen by the switch driver (Fig. 7.16).

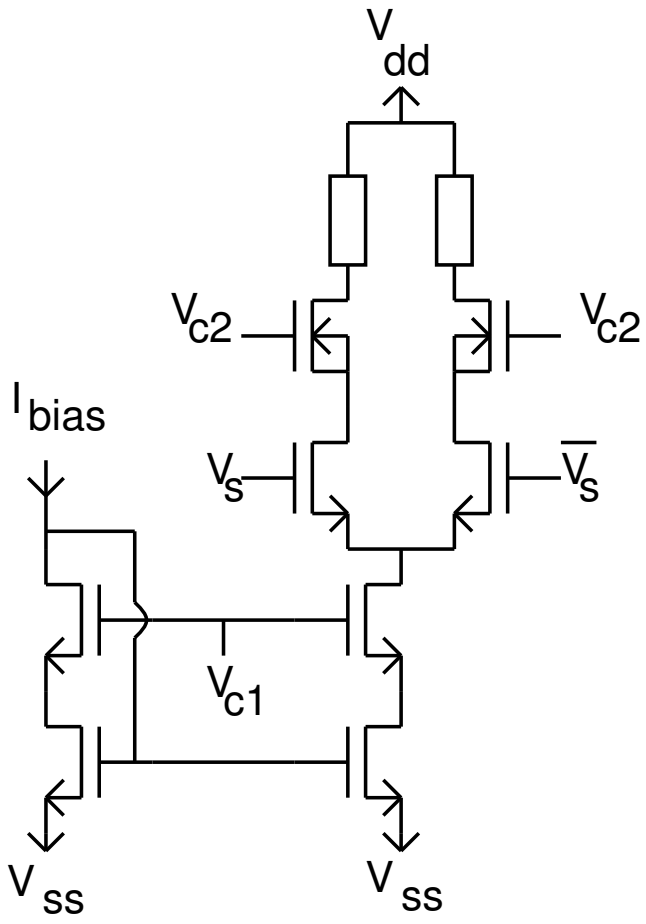

Figure 7.14 One set of current source, switches and cascode transistors.

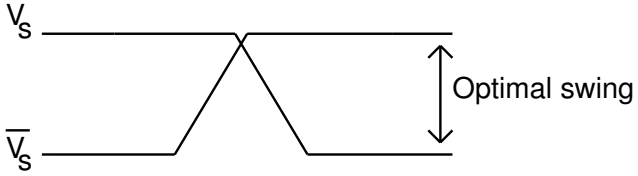

Figure 7.15 The switching waveform.

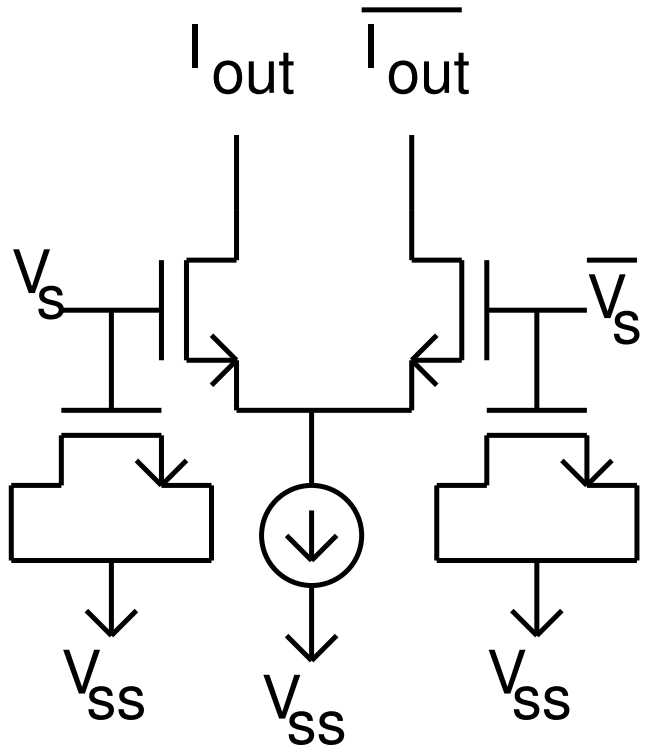

Figure 7.16 Dummy transistor as additional load.

d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d
d	d	d	d	24	11	22	13	20	15	18	16	19	14	21	12	23	10	d	d	d	d
d	d	17	16	19	14	21	12	23		8	10	24		9	11	22	17	20	15	18	13	d	d
d	d	13	18	15	20	17	22	11		9	24	10	m	23	12	21	14	19	16	17	d	d
d	d	d	d	10	23	12	21	14	19	16	18	15	20	13	22	11	24	d	d	d	d
d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d

Figure 7.17 The common centroid MSB current source matrix.

7.1.7.3 Layout issues

In order to achieve the best possible static matching, it is preferable to make the environment of each current source identical to each other. The best way to do this is to put the current sources in matrices and add some dummy transistors around the current source transistors. It is also preferable to use the common centroid approach when drawing the matrices in order to minimize the effect of the linear-gradient-type processing parameter variation as the function of position on the wafer. It is also preferable that the transistors are arranged so that all of the transistors of one source are not, for example, on the periphery of the matrix. A better way is to balance the location so that the mean distance from the center of the matrix is constant. This means that, if two transistors of the single current source are at the edge of the matrix, another two are near the center. The method used for the MSB current source matrix is presented in Fig. 7.17. The sources are numbered from 8 to 24. The transistor marked with ”m” is for current mirroring. The transistors labeled with 8 and 9 are the weighted sources consisting of 1 and 2 transistors, respectively. The rest of the current sources have 4 transistors each. The common centroid matrix approach has also been used in the bias generation. The layout techniques for improving the static matching are analyzed more thoroughly in Sections 6.8 and 7.2.3.

In addition to the static matching optimization by common centroid matrices, the dynamic performance is enhanced by using different supply voltage sources for the analog part and bias, switch drivers and digital logic. Also, guard lines are used wherever possible. Ground planes are also added to decrease the capacitive coupling between the digital and analog signals.

7.1.7.4 D/A converter simulations

The simulations for the spectral purity of the D/A converter were carried out with sinusoidal signals of different frequencies. The sampling frequency used for the simulation was 66.67MHz. In Fig. 7.18 a spectrum of a 5.43MHz sine signal is presented. It can

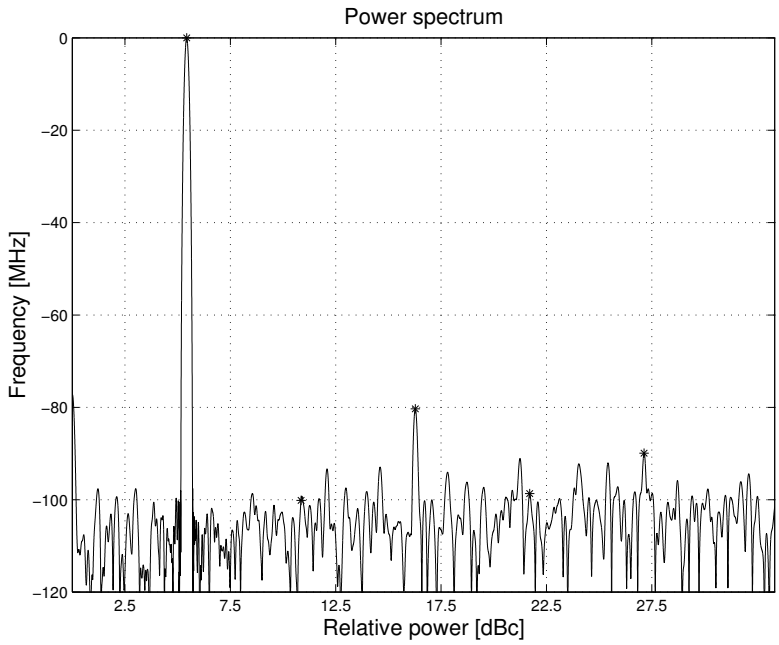

Figure 7.18 Spectrum of 5.43MHz sine signal.

be seen that the third and the fifth harmonic component dominates *SFDR* and that the level of the even harmonics is low. The values for the 3rd and 5th harmonic components are -80.32dB and -89.92dB, respectively.

In Fig. 7.19 the signal frequency of the sine is increased to 28.5MHz. Now the dominant components of *SFDR* are the 2nd and the 3rd harmonic component.

7.1.8 Experimental results

The chip was fabricated in a 5 metal 2 poly 0.35μm BiCMOS process, but only CMOS transistors were used. Isolation between the digital part of the chip and the D/A converter is achieved by the usage of isolation rings. The same kind of isolation is also available in triple well CMOS processes. The design features that are not available in conventional CMOS processes are also used. Such features are, for example, isolated NMOS transistors.

7.1.8.1 Measurement setup

The measurement setup for the circuit is shown in Fig. 7.20. The parallel port of the PC is used to control the functionality of the chip. The control program, which controls the parallel port, is written in C programming language [172]. With the program, it

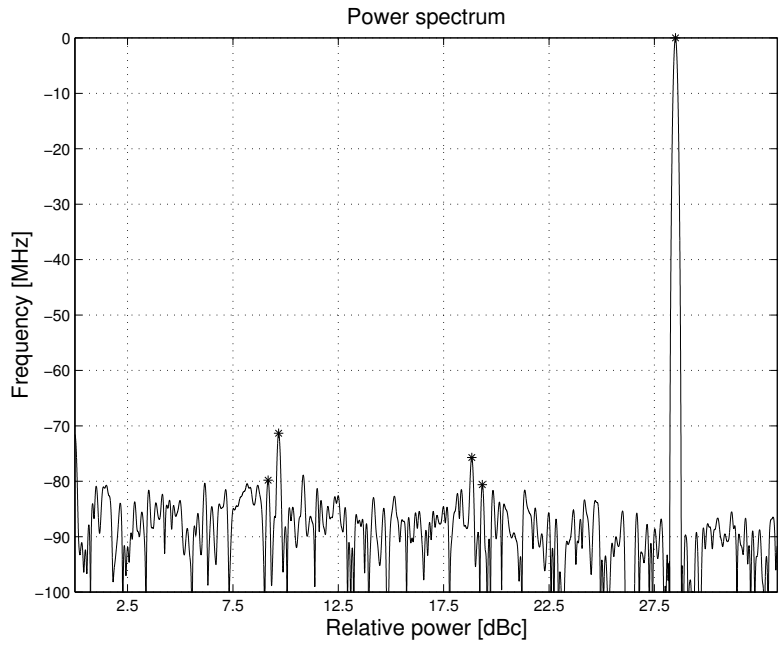

Figure 7.19 Spectrum of 28.5MHz sine signal.

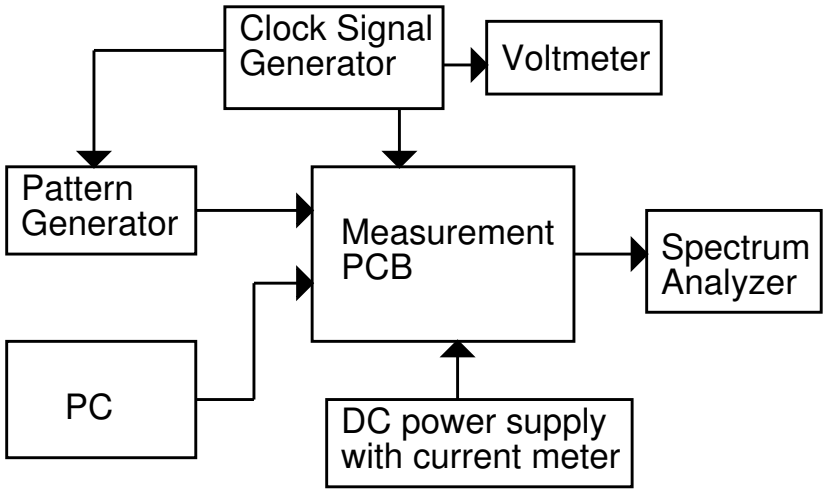

Figure 7.20 Measurement setup for the circuit.

is possible to set the carrier frequencies and select which part of the chip is tested. It is possible to test each of the filters separately and also to test the CORDICs by selecting either one or all of them to be used. The D/A- converter can be measured either as connected to the digital transmitter or in a stand-alone mode. Manufacturers

Table 7.5 Types of the measurement equipment

Pattern generator	Tektronix TLA 720
Clock signal generator	Rohde-Schwartz 1062.5502.11
Voltmeter	Hewlett-Packard 3457A
Spectrum/Vector analyzer	Rohde-Schwartz 1088.3494.30/FSIQ
DC-supply	Hewlett-Packard E3631A

and models of the measurement equipment are listed in the Table 7.5.

7.1.8.2 D/A-converter performance

The dynamic performance of the D/A-converter was measured with the sample frequency of 61.44MHz and the static performance with the sample frequency of 1Hz. The supply voltage was 3V, which is the typical operation condition for the converter used in the transmitter. The bias current of the current source matrices was 200μA, as designed; however, the bias current for the switching level setting were increased from 100μA to 450μA in order to lower the switching voltage levels at the input of the switches. It is assumed that this enhances the dynamic performance of the D/A converter because the switches stay better in the saturation region when they are open, resulting in a reduced level of the 2^{nd} harmonic component.

7.1.8.3 Static performance of the D/A-converter

The static performance of the D/A-converter is characterized by the differential nonlinearity and integral nonlinearity. These are measured by feeding the ramp signal with 1 LSB step to the D/A-converter operating with a 1Hz sampling frequency. The measured *DNL* and *INL* are presented in Fig. 7.21. The maximum absolute values of *DNL* and *INL* are 1.87LSB and 2.165LSB, respectively.

In Fig. 7.22 the ideal sine signal is fed to the Matlab function that distorts the signal with the static nonlinearities of the D/A-converter. It can be seen that the static nonlinearities limits the spurious free dynamic range to be about 83dB.

7.1.8.4 Dynamic performance of the D/A-converter

In Figs 7.23 - 7.30 the spectra of the sine signals of frequencies 1, 3, 7.5, 12.5, 17.5, 22.5, 27.5 and 30MHz are presented. It can be seen that the third harmonic dominates at the low frequencies. The second harmonic increases as the signal frequency is increased and becomes dominant at the signal frequencies near half of the sampling frequency. The spurious free dynamic range of the converter vs. signal frequency is presented in Fig. 7.31.

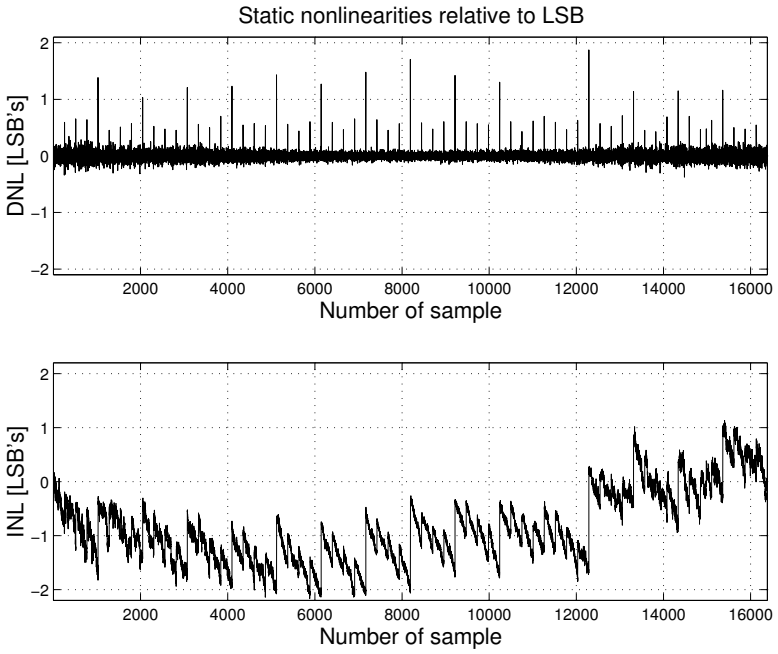

Figure 7.21 Differential and integral nonlinearities of the D/A-converter.

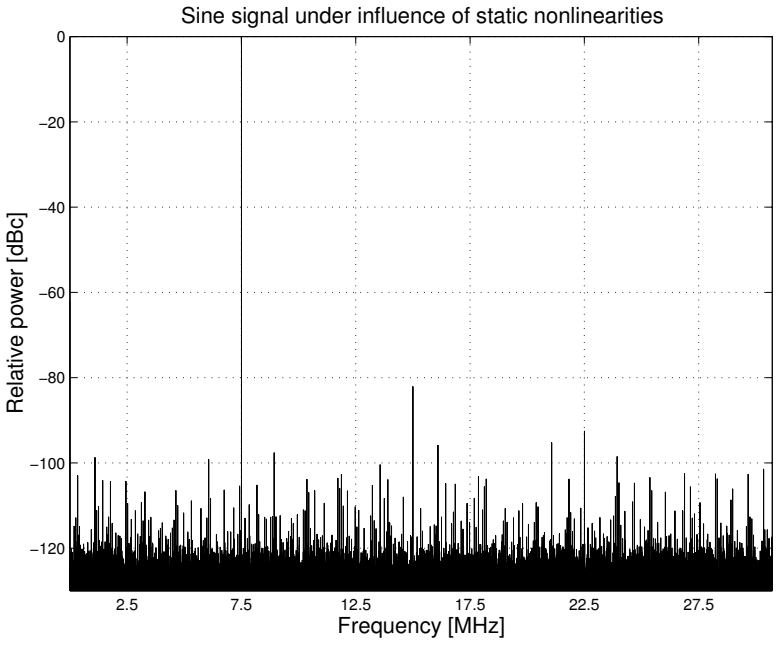

Figure 7.22 Sine signal distorted with the static nonlinearities.

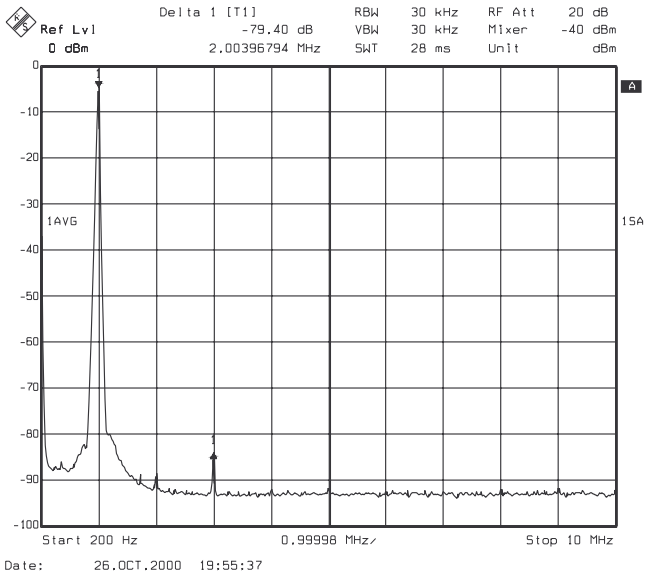

Figure 7.23 Spectrum of the 1MHz sine signal.

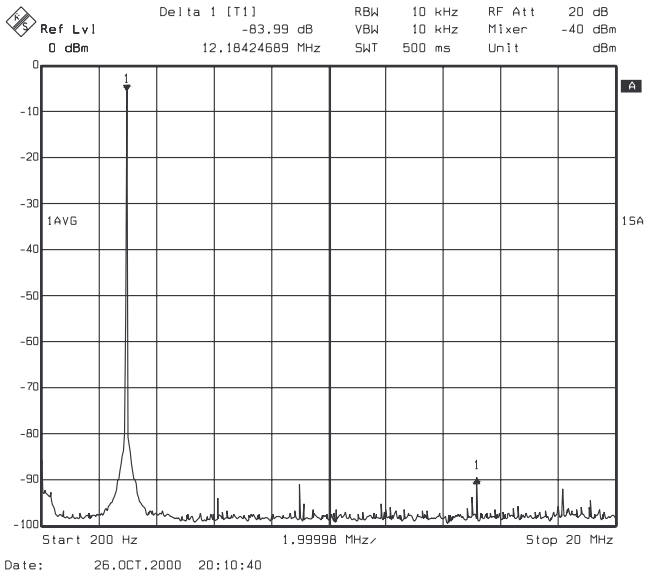

Figure 7.24 Spectrum of the 3MHz sine signal.

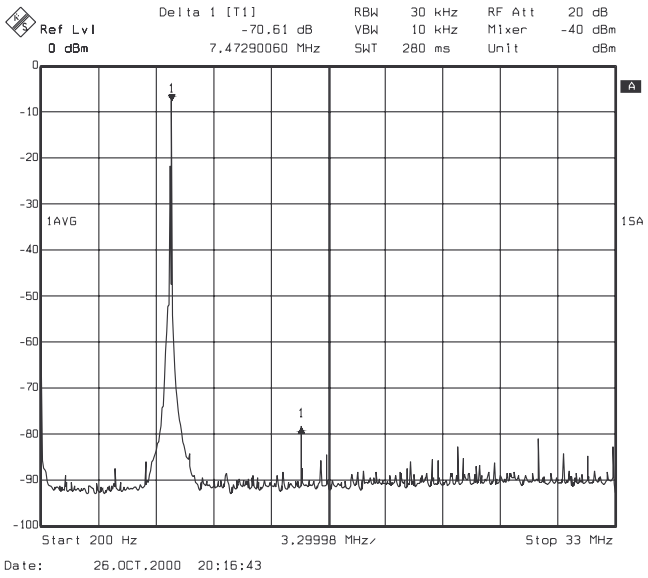

Figure 7.25 Spectrum of the 7.5MHz sine signal.

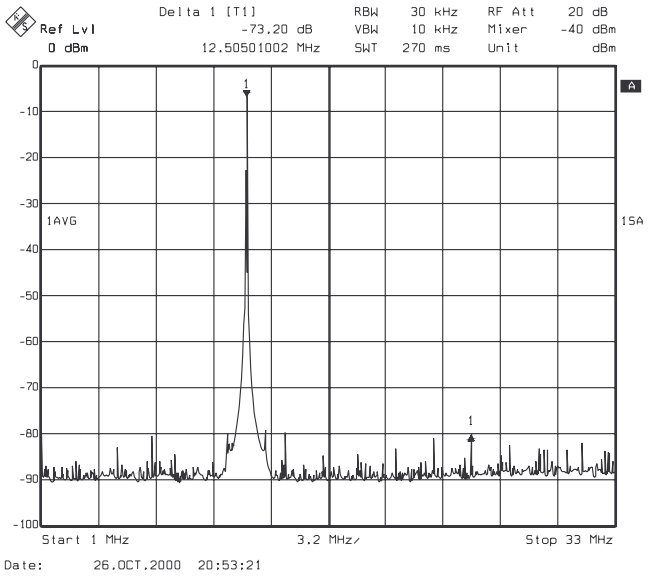

Figure 7.26 Spectrum of the 12.5MHz sine signal.

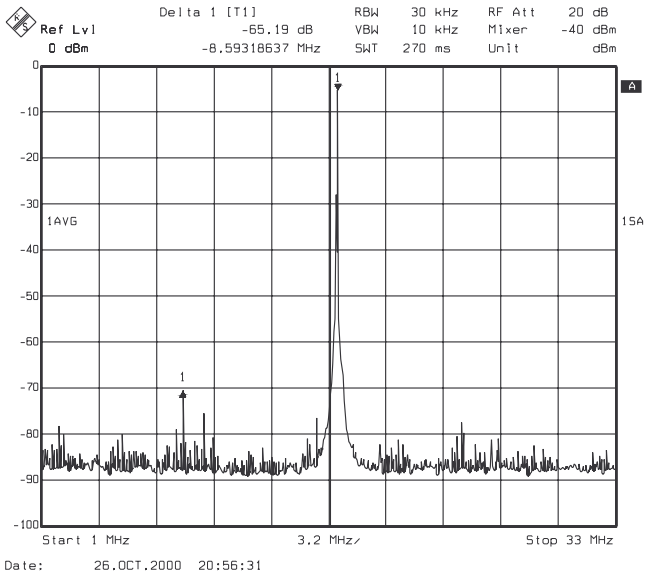

Figure 7.27 Spectrum of the 17.5MHz sine signal.

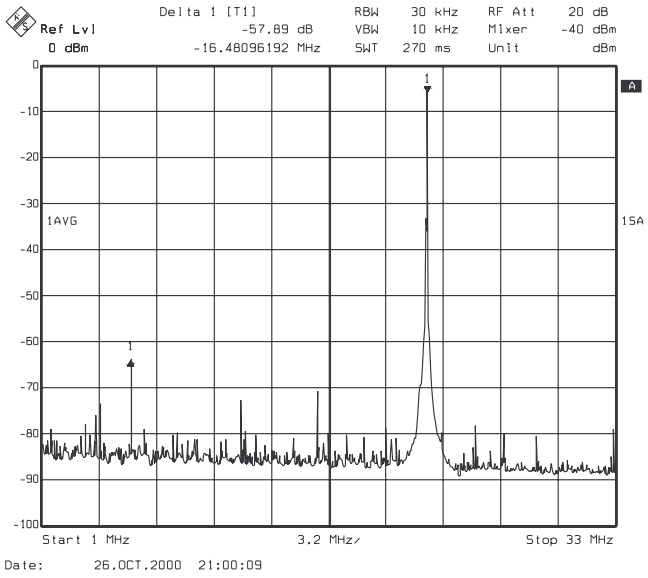

Figure 7.28 Spectrum of the 22.5MHz sine signal.

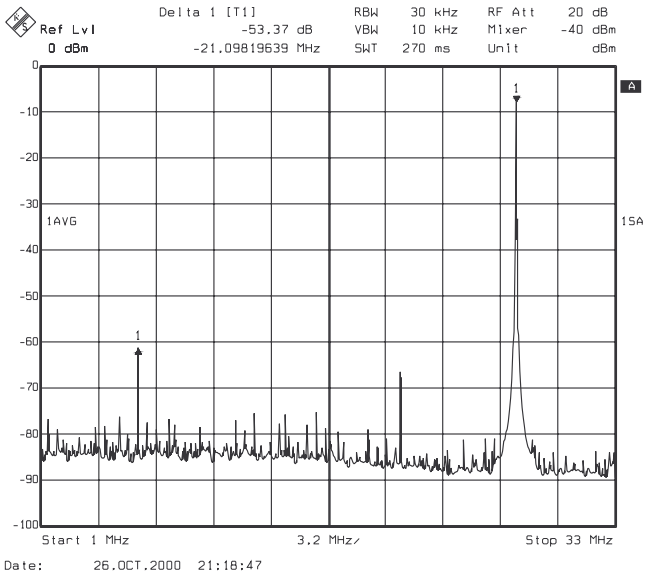

Figure 7.29 Spectrum of the 27.5MHz sine signal.

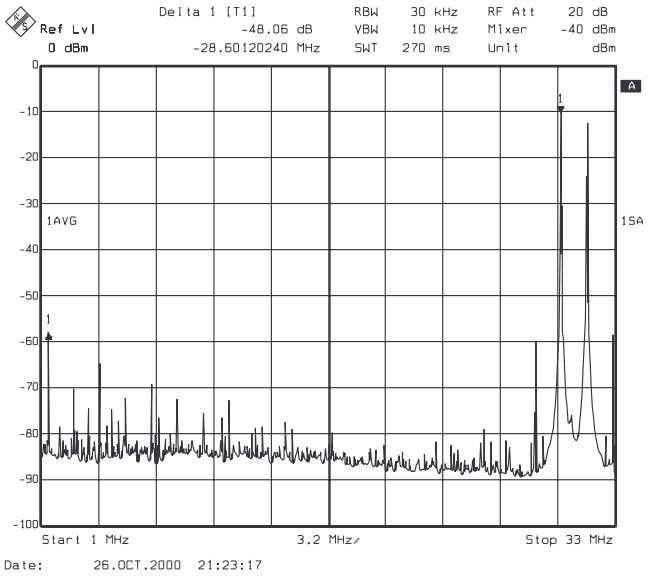

Figure 7.30 Spectrum of the 30MHz sine signal.

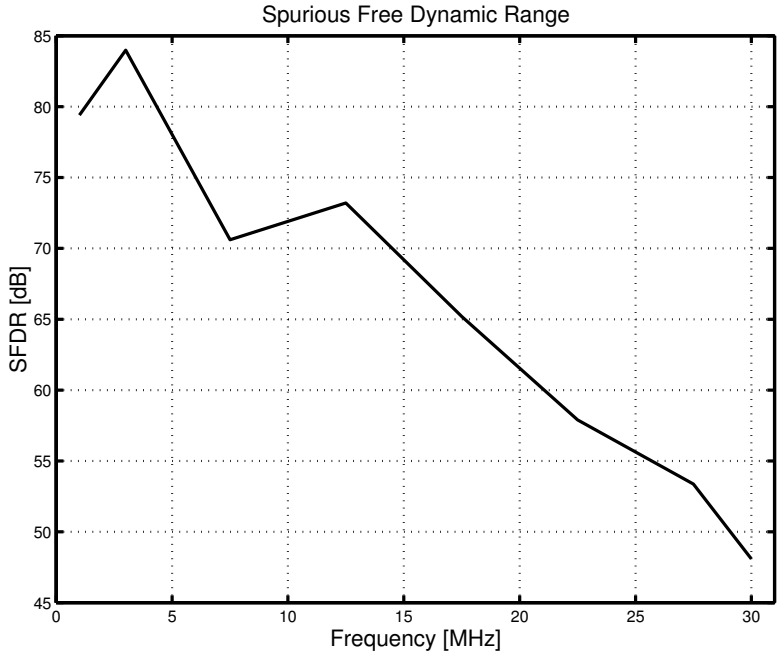

Figure 7.31 Spurious free dynamic range vs. signal frequency.

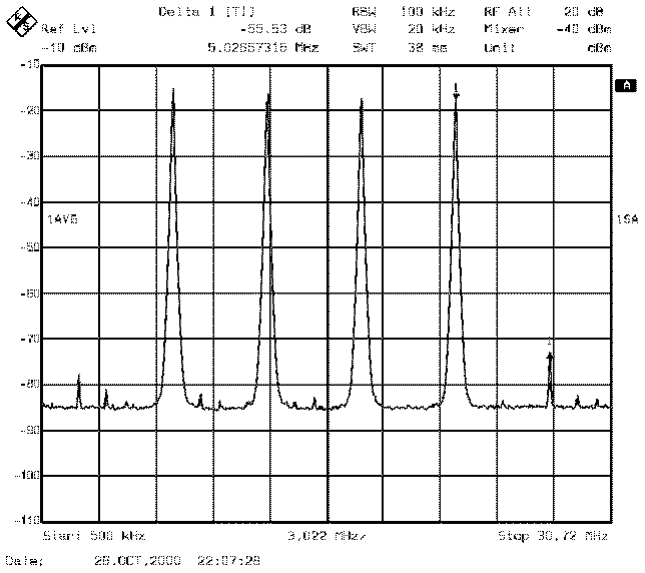

Figure 7.32 Spectrum of multiple carriers.

In Fig. 7.32 the spectrum of the sum of four carrier signals of frequencies 7.5, 12.5, 17.5, and 22.5MHz is presented.

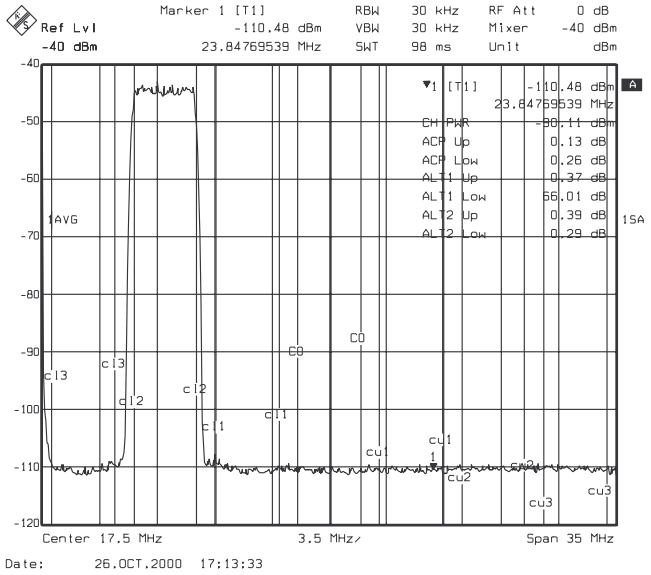

Figure 7.33 Wide-band signal at 7.5MHz.

The total power dissipation of the D/A converter was measured to be 141mW. The high power dissipation is mostly due to the static current flowing through the switch drivers. The contribution of the switch drivers to the power consumption is 93mW. The static current is used to set the level of the switch driving voltage. The power consumption can be reduced if an inverter-type driver is used to drive the switch. However, then the voltage level has to be generated by other means, with the regulator outside the chip, for example. The area of the D/A converter is 3.45 mm^2.

7.1.8.5 QAM Performance

The QAM performance was measured with normally distributed and truncated data with the crest factor of 10dB. In the multicarrier case, the data had to be scaled with the factor of 0.9 before the transmission because of a slight bug in the system. There was saturating adders at the output of the CORDIC. If the two signals saturates the adder and the third signal has the opposite sign, the signal is no longer saturated, and the step type error is generated, resulting in spurs generated on the sidebands. The system clock frequency was 61.44MHz. The data was fed to the transmitter with the pattern generator which was synchronized to the system clock. *ACLR* was measured with the spectrum analyzer. The spectra of the WCDMA signal with different combinations of the carrier frequencies are presented in Figs 7.33 - 7.37.

EVM was measured with the single and four carriers each modulated with 4 QAM signals with the symbol magnitude 40dB below the maximum. The measured constel-

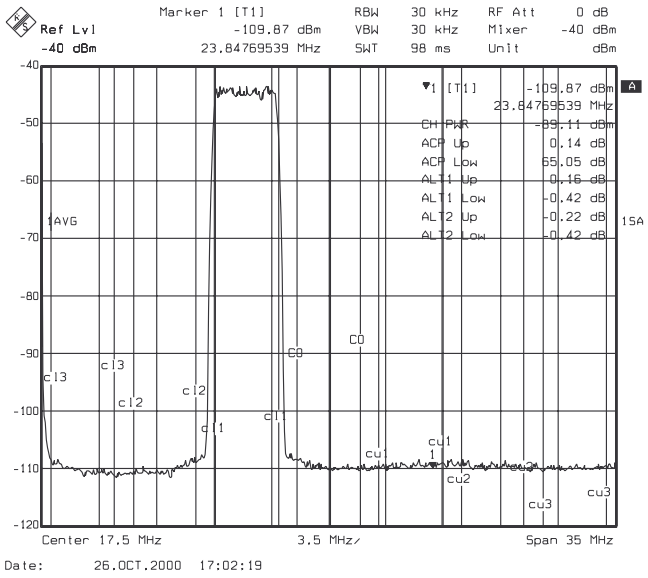

Figure 7.34 Wide-band signal at 12.5MHz.

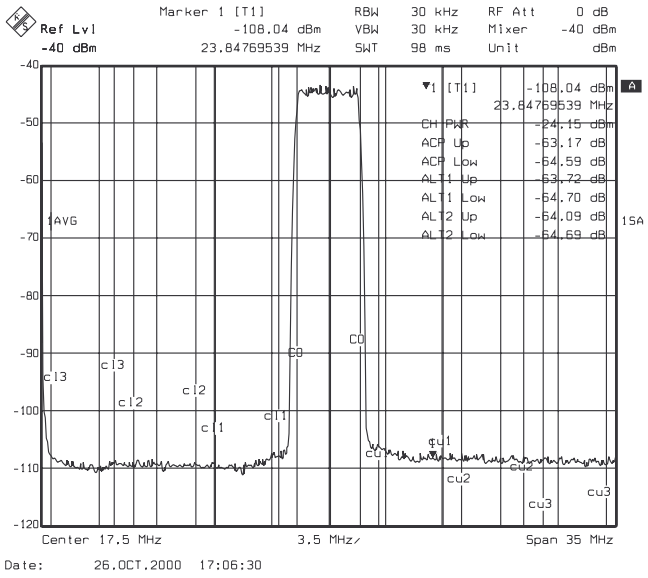

Figure 7.35 Wide-band signal at 17.5MHz.

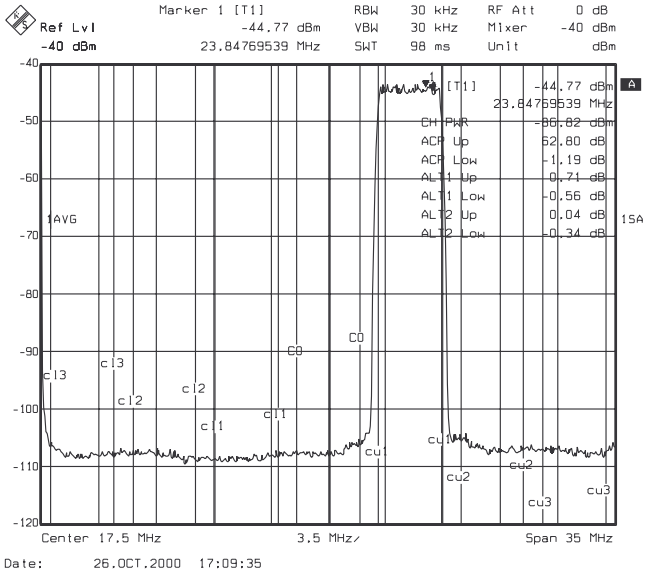

Figure 7.36 Wide-band signal at 22.5MHz.

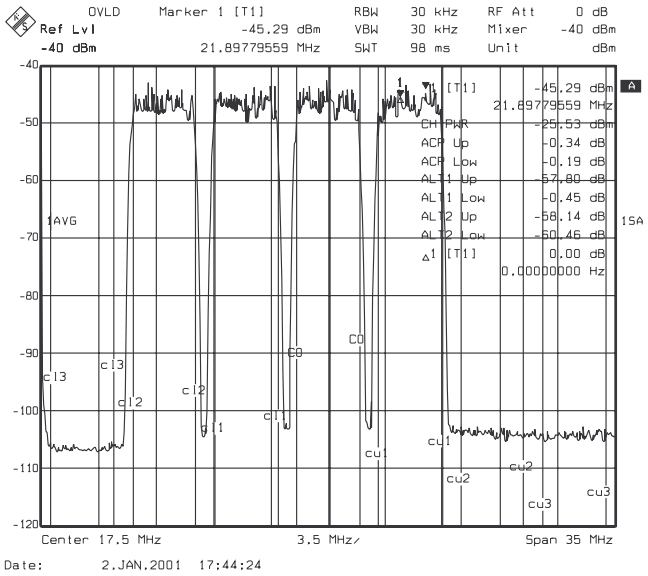

Figure 7.37 Multicarrier WCDMA signal.

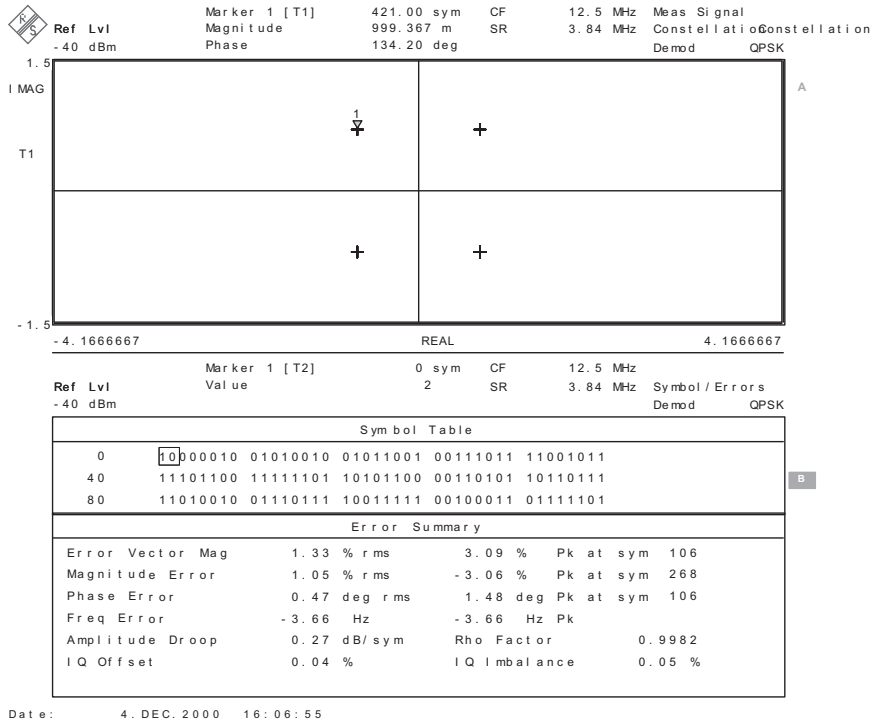

Figure 7.38 Signal constellation and statistics of EVM measurement.

lation for the 22.5MHz carrier and the corresponding EVM, phase error and amplitude error are presented in Fig. 7.38. The supply voltage of the digital signal processing part has a huge effect on the total power consumption of the transmitter. The lowest possible supply voltage for the digital part was determined by measurement. The dependence of the power consumption and the supply voltage is presented in Fig. 7.39. The minimum supply voltage that the digital transmitter operated with was 2.165V and the maximum clock frequency that the transmitter operated with a 3V supply voltage was 73MHz. The measured performance metrics are listed in Table 7.6.

ACLR specification of the WCDMA base station transmitter [20] is 45 and 50dB for $ACLR_{12}$ and 17.5% for EVM. It can be said that this digital IF part of the multicarrier WCDMA base station transmitter is applicable to the 3rd generation wireless communications systems. There is also some margin left for the signal degradation in the analog signal processing part.

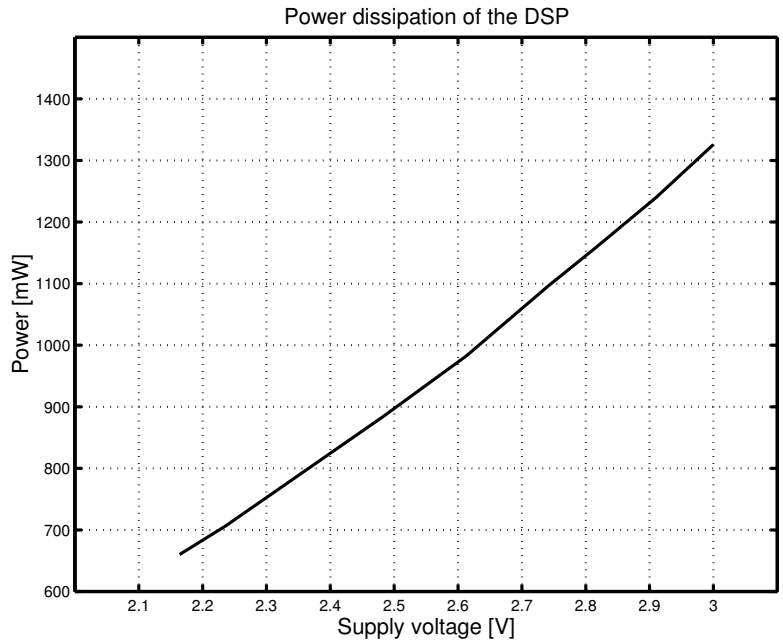

Figure 7.39 Power dissipation of the digital transmitter as a function of the supply voltage.

Table 7.6 Measured performance characteristics of the transmitter.

Symbol rate	3.84MHz
Channel bandwidth	3.84MHz
Carrier spacing	5MHz
Nominal clock frequency	61.44MHz
Worst case $ACLR_{12}$ in single carrier case (22.5MHz carrier)	62.09/62.76 dB
ACLR in multicarrier case	57.80/58.14dB
Worst case EVM_{rms}	-37.5 dB (1.33%)
Power dissipation with nominal (3V) digital power supply	1.33W+141mW=1.47W
Power dissipation with optimal (2.165 V) power supply	660mW+141mW=801mW
Maximum operation frequency	73MHz
Area die/core/D/A converter	25.75/19.18/3.45mm^2

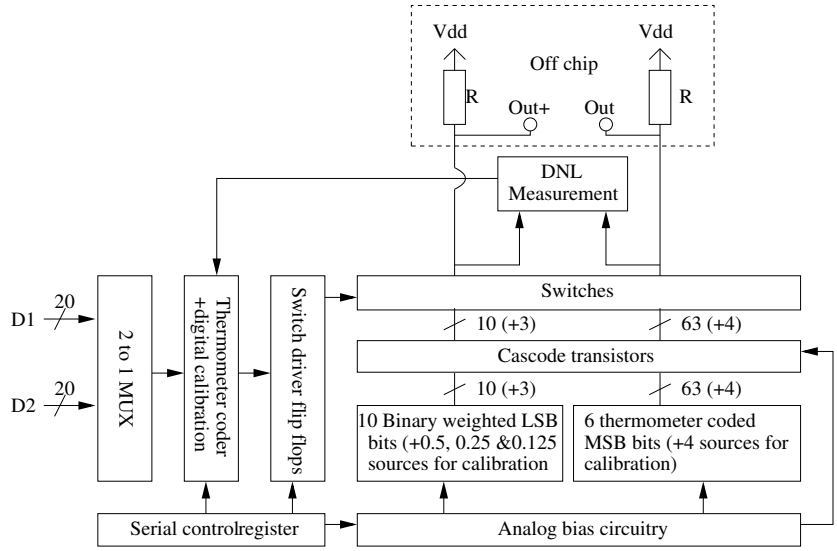

Figure 7.40 Block diagram of the prototype

7.2 Prototype of the 16-bit 400 MS/s D/A converter with digital calibration

In this section, the design and experimental results of a 16-bit 400 MS/s D/A converter prototype fabricated in a 3.3V 0.35 μm SiGe BiCMOS process are presented. The converter is designed with CMOS transistors only.

The designed converter consists of converter core, digital calibration circuitry, control registers, and measurement circuitry for the current source mismatch. The block diagram of the converter is presented in Fig. 7.40.

The architecture is a segmented current-steering converter with 6 thermometer-coded and 10 binary-weighted bits. Four thermometer-coded current sources are added to MSB sources in order to introduce redundancy in the transfer function. Redundancy is required by the calibration algorithm as described in Section 6.4. Three additional binary-weighted current sources with weights 0.5, 0.25, and 0.125 LSB, respectively, are added in order to have adequate resolution for the calibration. The 2-to-1 multiplexer (MUX) is used at the input, in order to be able to achieve 400MS/s input data rate.

Fig. 7.41 represents the single current branch and switch set used in the converter. Two cascode transistors are used to increase the output impedance and reduce the distortion due to output impedance variation. An additional set of switches is used to enable the differential quadrature switching scheme [151] [168], which will be discussed later in this section.

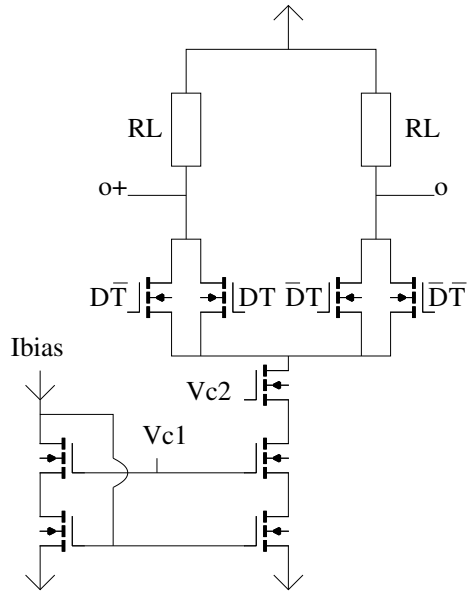

Figure 7.41 Single current branch of the converter.

7.2.1 Current source dimensions and DC matching

Once the DC operation point of a current branch was defined in order to have adequate output swing and large enough V_{ds} for all the transistor in order to keep them in the saturation region, the dimensions for the current source transistors were determined in order to have adequate matching. Even though the usage of calibration alleviates the matching requirement of the MSB current sources, the matching of the LSB sources is important since it will define the quality of the calibration and thus the static linearity of the converter. The methods for analyzing the *INL* and *DNL* behavior as a function of matching were discussed in detail in Section 6.3.

Assuming the standard deviation of the maximum value of the *INL* is related to the standard deviation of the current source as

$$\sigma_{INLmax} = C_1 \sqrt{2^N - 1} \frac{\sigma_{lsb}}{I_{lsb}}, \tag{7.5}$$

as presented in Section 6.3.3, and by defining the *INL* yield to be

$$\begin{aligned} Yield_{INL} &= P\left(-INL_{limit} \leq MAX\left(INL\right) \leq INL_{limit}\right) && (7.6)\\ &= P\left(-C_2\sigma_{INLmax} \leq MAX\left(INL\right) \leq C_2\sigma_{INLmax}\right) && (7.7) \end{aligned}$$

we obtain

$$\sigma_{INLmax} \leq \frac{INL_{limit}}{C_2}, \tag{7.8}$$

$$\frac{\sigma_{LSB}}{I_{LSB}} \leq \frac{INL_{limit}}{C_1 C_2 \sqrt{2^N - 1}} \tag{7.9}$$

$$= \frac{INL_{limit}}{C\sqrt{2^N - 1}}, \tag{7.10}$$

in which the product of constants C_1 and C_2 can be combined in a single constant C.

Combining the transistor matching equation Eq. (6.5) and Eq. (7.10), the product of channel width and length of the current source transistor are obtained as

$$WL = \frac{\frac{4A_{vt}^2}{\left(V_{gs}-V_T\right)^2} + A_\beta^2}{\frac{INL_{limit}^2}{C^2\left(2^N-1\right)} - \frac{4S_{vt}^2 D^2}{\left(V_{gs}-V_T\right)^2} - S_\beta^2 D^2}, \tag{7.11}$$

In which D is the average separation of two current sources. On the other hand, for the MOS transistor in saturation holds

$$\frac{W}{L} \approx \frac{2I_{lsb}}{\mu_o C_{ox}\left(V_{gs} - V_T\right)^2}. \tag{7.12}$$

Together these two equations determine the dimensions of the LSB current source transistor.

It can be observed from the Monte-Carlo simulation curves presented in Section 6.3.3 that, by choosing $C \approx 2$, $Yield_{INL}$ very close to 100% should be achieved. In the early phase of the design process, it was assumed as a coarse simplification that the maximum absolute value of the INL follows the normal distribution, and, by selecting $C = 3$, the $Yield_{INL} = 99.7$ should be obtained. However, in the 16-bit case, the total area required would become very large. Therefore the dimensions of the current sources are determined by the maximum area that can be used for the converter.

Due to the limited area available, it was decided to increase the error limit to be $INL_{limit} = 8$, which corresponds to the linearity requirement of a 12-bit converter. Referring to research results presented in Section 6.3, it can be stated that, when comparing the above mentioned approximation method and Monte-Carlo simulation results, the same area of the current sources is obtained with $C = 2$ and $INL_{limit} \approx 5.33$, indicating that, with the used current source area, nearly 100% of converters should have $INL \leq 5.33$ instead of $INL \leq 8$. The area required for 16-bit matching ($INL_{limit} = 0.5$, $C = 2$) is 113.7 times the current area, assuming the effect of increased distance between sources is negligible. On the other hand, there exists an optimum area of the current source that gives the best matching for a certain number of bits, as presented in

[173]. Assuming that the average distance between the current sources is

$$D = \sqrt{2\left(2^N - 1\right)WL}, \tag{7.13}$$

the area WL that results minimizes the $\frac{\sigma_{LSB}}{I_{LSB}}$ in Eq. 7.10 can be calculated to be

$$WL_{opt} = \sqrt{\frac{4A_{vt}^2 + 2A_{\beta}^2\left(V_{gs} - V_T\right)^2}{4S_{vt}^2\left(2^N - 1\right) + 2S_{\beta}\left(2^N - 1\right)\left(V_{gs} - V_T\right)^2}} \tag{7.14}$$

Increasing the area beyond this limit will, however, exacerbate the matching. However, at the time of designing the prototype, the accurate information of the transistor matching as a function of distance between transistor was not known.

7.2.2 Output impedance

As analyzed in Section 6.5, the variation of the output impedance may have a significant effect on the harmonic distortion of a current steering D/A converter. Increasing the W and L of the switches instead of using minimum L increases the switching impedance, as presented in Section 6.5. Simulation results for simultaneously stepping the W and L of the MSB switch from minimum to 5 times the minimum is presented in Fig. 7.42. The impedance value is $20\log\left(\frac{2^{10}\times Z_{umsb}}{50}\right)$ corresponding to the effect of MSB current source referred to the value of LSB unit impedance.

Increasing the W and L rapidly increases the Z_u. The practical scaling range is from 1 to 5 times the minimum due to the fact that the area of the switches can not be further increased due to the finite driving capability of the switch drivers. Also, capacitive coupling may introduce undesired behavior if the area of the switches becomes large.

Fig. 7.42 indicates that the pole p with $\omega_p \approx \frac{g_{ds3}g_{ds2}g_{ds1}}{C_3 g_{m3}g_{m2}}$ of Eq. (6.68) dominates when the scaling factor is larger than 2, and therefore the impedance on the frequency range from 10kHz to about 100MHz cannot be further improved by scaling the switch transistor dimensions (in contrast to the situation presented in Fig. 7.43). Further increasing the scaling factor will move the dominant pole to lower frequencies and increase the DC impedance, which is already large enough. It can also be observed that the capacitance of the source node of the switches (C_3 in Fig. 6.44) is determined by the transistor area, since the frequency of the zero z_2 with $\omega_{z2} \approx \frac{g_{m4}}{C_3}$ scales down according to transistor scaling. Moving down the zero also improves the impedance performance from 100MHz to 200MHz, i.e. near the Nyquist frequency of the converter, which is as desired.

In order to demonstrate the frequency behavior of the Z_u in the presence of the routing capacitance of current source matrices, the value of the current source capacitor

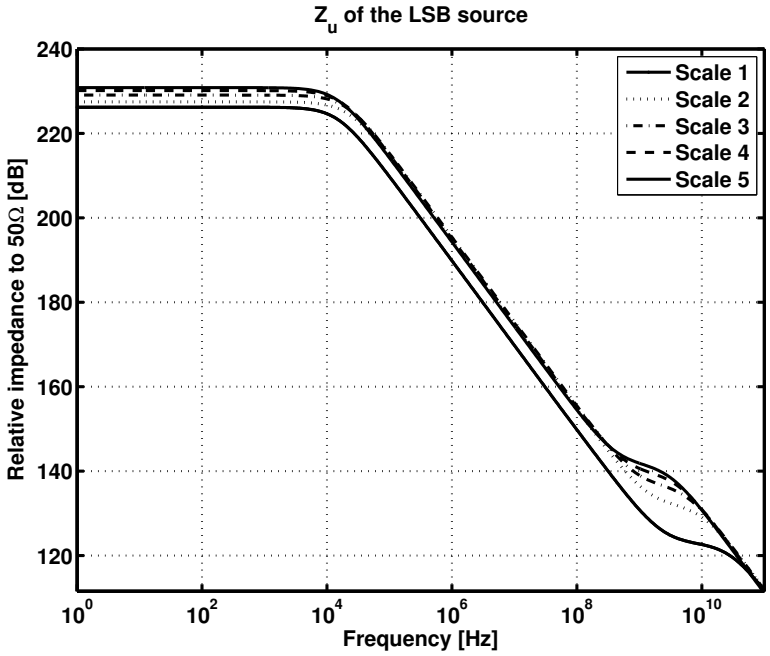

Figure 7.42 Effective Z_u of MSB current source reduced to LSB unit switch.

(C_1 in 6.44) is increased to $C_1 = 1.2nF$, which corresponds to an about 800x800μm^2 metal-metal capacitor in a typical CMOS process. This ensures the C_1 causes the dominant pole. The result of the simulation is presented in Fig. 7.43. In this case, the scaling of the switch effectively improves Z_u on the frequency range from 10KHz to 1MHz according to scaling of $\frac{g_{m4}}{g_{ds3}}$, since the pole due to C_3 is not dominant. The zero z_2 is moved down in frequency according to scaling of C_3.

In both cases, the usage of the scaling factor 3 is justified due to improvement of the impedance within the frequency ranges from 10KHz to 1MHz and from 100MHz to 200MHz. A greater improvement is achieved if C_1 initially causes the dominant pole. There also seems to exist an additional zero around 1GHz, which is not present in the small-signal analysis. However, it is above the Nyquist frequency, thus not affecting the performance of the converter. Figure 7.44 represents the theoretical *SFDR* values obtained with current switch dimensions. The true SFDR performance due to finite output impedance lies somewhere between the lines presented in Fig. 7.44, due to the fact that perfect differentiality can not be achieved.

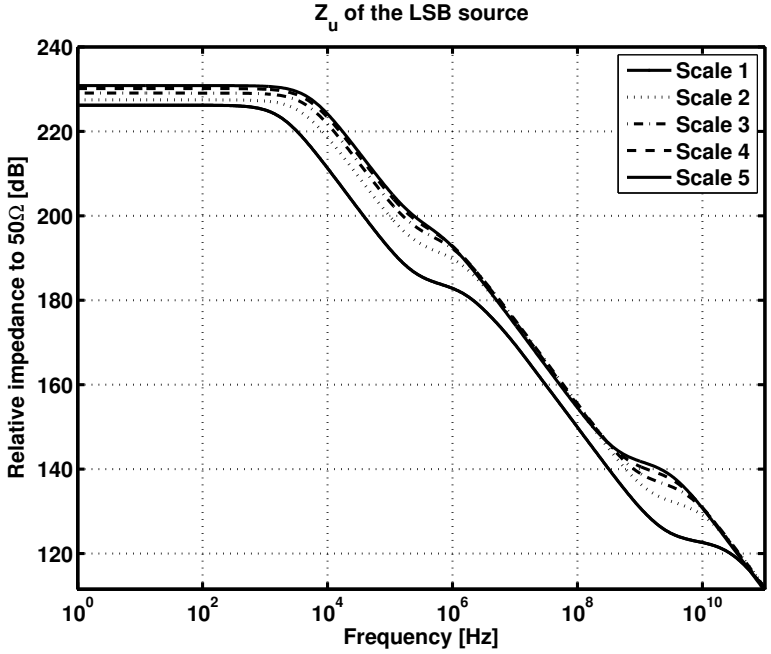

Figure 7.43 Effective Z_u of MSB current source reduced to LSB unit switch when routing capacitance C_1 is very large.

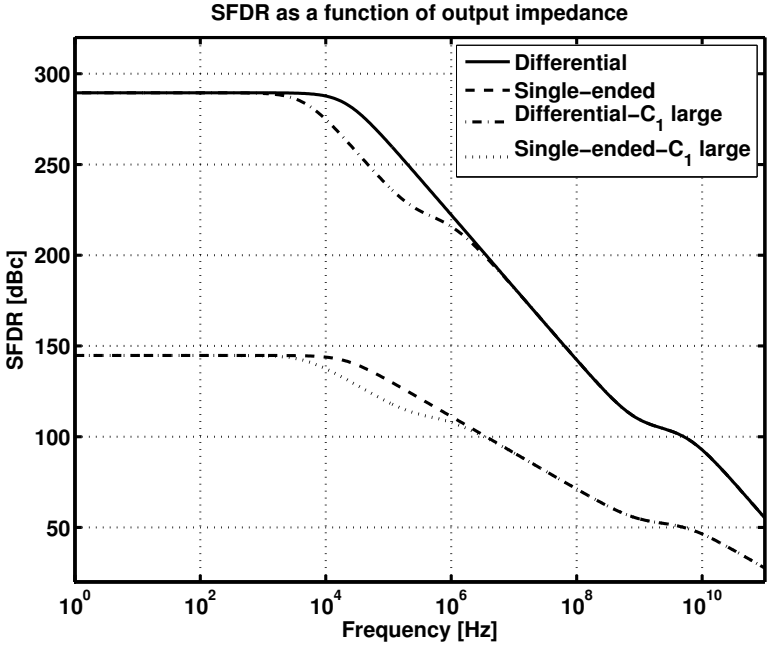

Figure 7.44 Effect of the output impedance on *SFDR* of the converter.

Q	**M**	**M**	**M**
M	**M**	**M**	**Q**
M	**M**	**Q**	**M**
M	**Q**	**M**	**M**

Figure 7.45 MSB current source matrix.

7.2.3 Current source matrices

The current source matrices were designed by using 16 sub-matrices mirrored in X-, X- or both directions in order to produce one common-centroid current source matrix that effectively reduces the effect of both even- and odd-order gradients. The principle of the mirroring is presented in Fig. 7.45

The sub-matrix Q in Fig. 7.45 was designed by applying the optimal sequences both in x- and y- direction simultaneously, as presented in 6.8. Sequences were chosen in order to minimize the effect of even-order gradient, since the residual gradient not canceled by symmetry will be of even order. Suitable shape of the matrix is obtained by using 8x9 grid with four dummies in the corners. The matrix of 67 MSB current sources and a mirroring transistor are presented in Fig. 7.46. When a dummy transistor is encountered while placing the current sources according to the sequences presented in Fig. 7.46, the current source is placed in the next available position. Therefore source ”3” is not in the corner [3,3], but in the xy-position [4,4], which is the next available position.

Fig. 7.47 represents the comparison between the maximum absolute value of INL as a function of the plane angle θ (see Eq. 6.128). It can be seen that the even-order optimal sequence results in a smaller error, since it cancels out the residual even-order error more effectively than the linear sequence.

Fig. 7.48 represents the comparison between the maximum absolute value of INL as a function of the plane angle θ in the case where the sub-matrices are placed in a 4x4 matrix with no mirroring in any direction. Comparison between the Figs 7.47 and 7.48 reveals that most of the benefit is achieved by using mirror symmetry, whereas the sequence has a large effect when no mirroring is used.

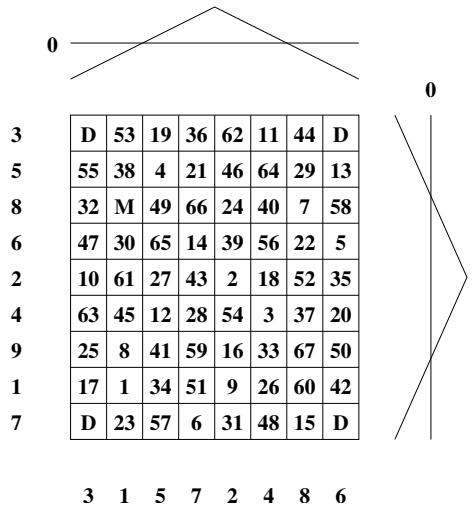

Figure 7.46 MSB sub-matrix and the optimal sequences used for even-order gradient cancellation.

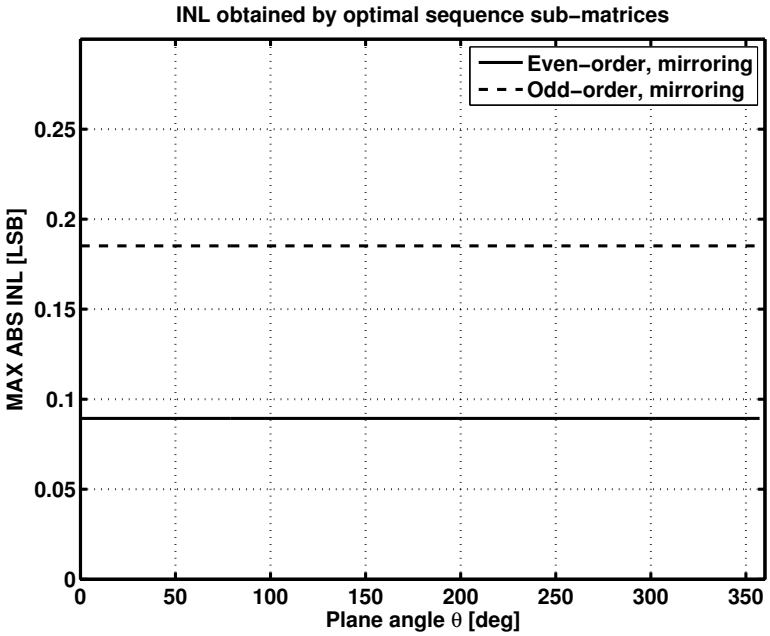

Figure 7.47 Maximum absolute *INL* with various plane angles.

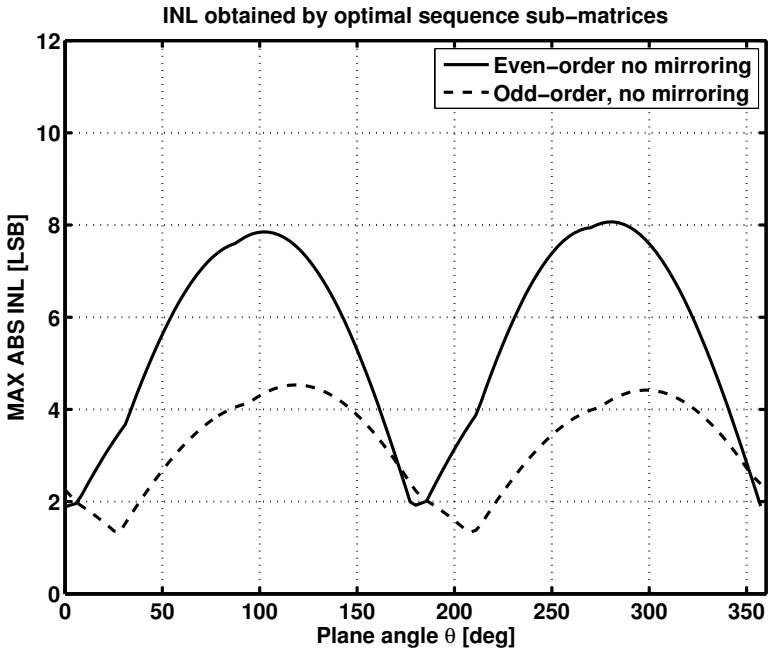

Figure 7.48 Maximum absolute *INL* with various plane angles.

The layout of the binary-weighted LSB matrix was designed by using principles presented in Section 6.8. In order to reduce the error between the MSB and LSB matrices, the current sources are biased with current mirroring ratio 1:2, resulting in the bias current of the LSB sources to be half of the bias of the MSB sources. This reduces the effect of the gain error of bias current mirroring, since the bias current in the LSB matrix is divided when the actual currents are formed, instead of being multiplied, which would multiply also the gain error.

7.2.4 Dynamic performance

In addition to code-dependent output impedance variation, the dynamic performance ($SFDR$) of the converter is determined by, for example, jitter and timing differences between signal paths to and from the switches. The distortion due to code-dependent jitter can be effectively reduced by using the differential quad switching scheme (Fig.7.41) [151] and duplicated ”toggle”-signal path [168] [139]. The toggle signal changes its state on those clock cycles on which the actual data bit does not change its state. This ensures that the circuit activity, and thus the power-rail interference, is constant, and thus code-independent. The differential quad switching scheme ensures that the actual and ”toggle” switch drivers have equal capacitive loading, and therefore the currents

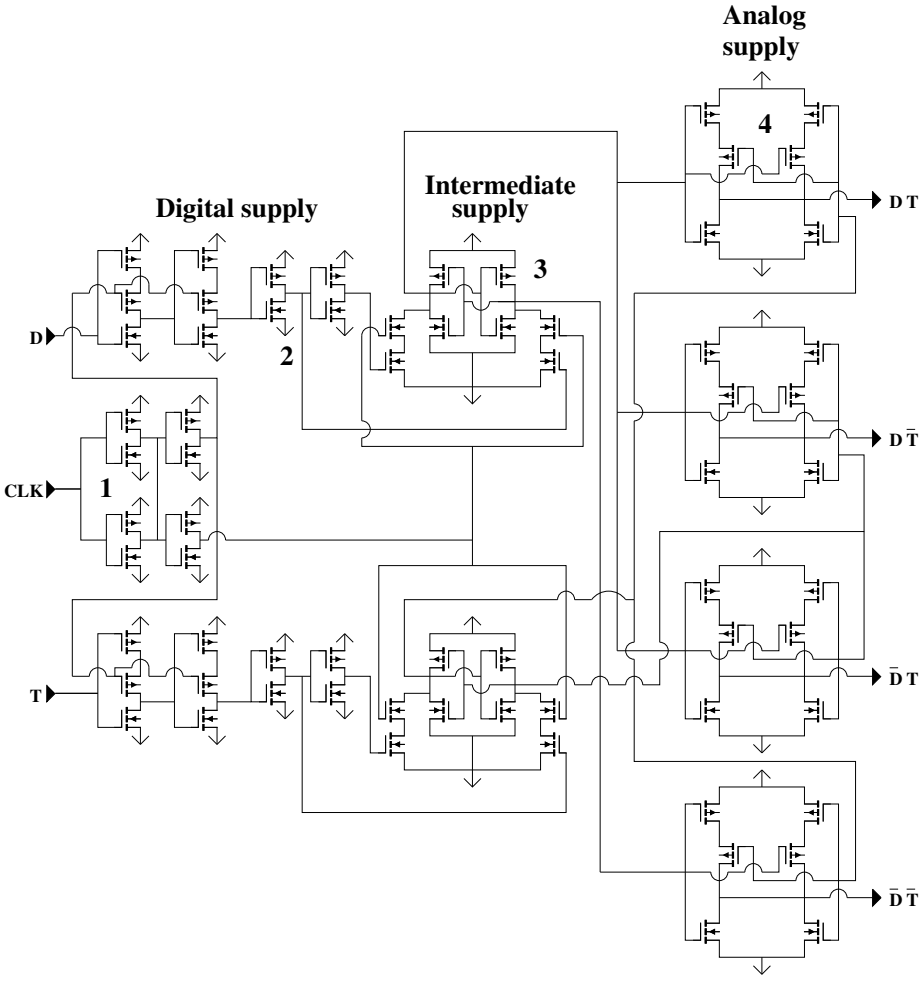

Figure 7.49 Switch driver for differential-quad switching.

drawn from supply rails should not vary between clock cycles. It also equalizes the voltage variation on the source node of the switches.

The switch driver signals are generated from the digital data and ”toggle” bits with the switch driver presented in Fig. 7.49. The clock buffers (marked with ”1”) are added in order to remove the data dependent clock load variation, the effect of which was analyzed in Section 6.7.3. The additional buffering (marked with ”2”) is used to ensure symmetry in both the signal and its inversion, which is not guaranteed when a true single-phase clocked input latch is used. The differential latch stage (marked with ”3”) is used to produce a differential switch driver signal and determine the crossing point of those signals. Finally, a symmetrical NOR-stage (marked with ”4”) is used to produce the combinations of data and ”toggle” signals required to steer the switches.

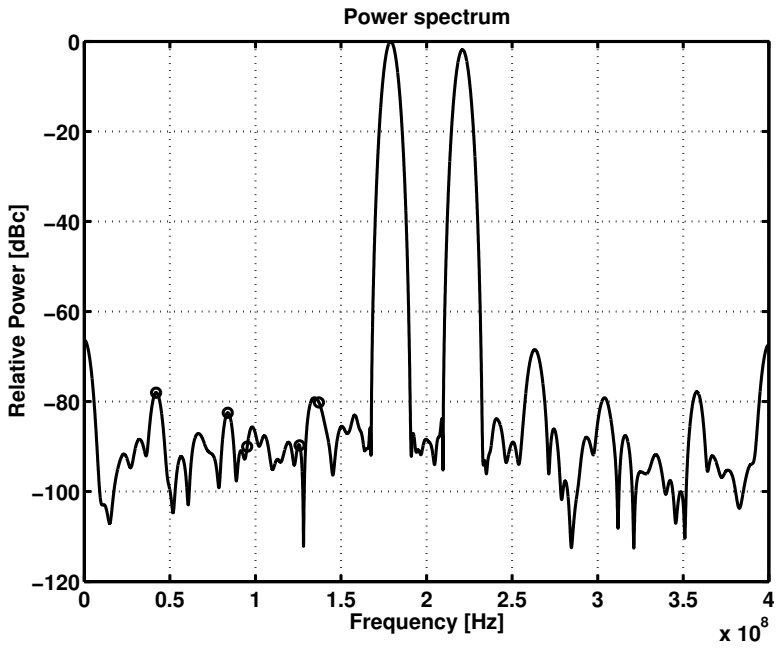

Figure 7.50 Simulated output spectrum of the converter when the differential-quad switching disabled.

Dedicated power supplies are used for the different stages in the driver in order to reduce the distortion due to power supply-interference.

The simulated spectrum when the differential-quad switching is disabled is presented in Fig. 7.50. The simulated *SFDR* is 78.04 dB on the Nyquist band, and is limited by the 2nd harmonic (DC component not taken into account).

The simulated spectrum when the differential-quad switching is enabled is presented in Fig. 7.51. The simulated *SFDR* is 62.3 dB in the Nyquist band, and is limited by the 3rd harmonic (DC component not taken into account).

The increase of the third harmonic component can be explained as follows by considering a fully thermometer-coded D/A converter for simplicity. The transition on the switch driver signals causes a bounce on the source node of the switches, as presented in Fig. 7.52. The bounce can be either due to capacitive coupling or due to simultaneously conducting switches. In addition, it can be assumed that the source node follows the output voltage of the conducting current branch attenuated by $\frac{g_{ds}}{gm}$ of the switch.

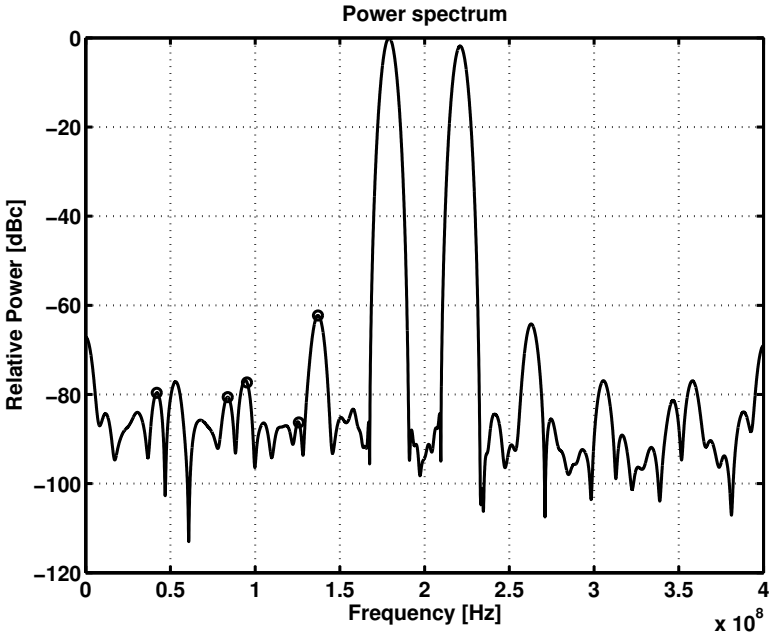

Figure 7.51 Simulated output spectrum of the converter when the differential-quad switching is enabled.

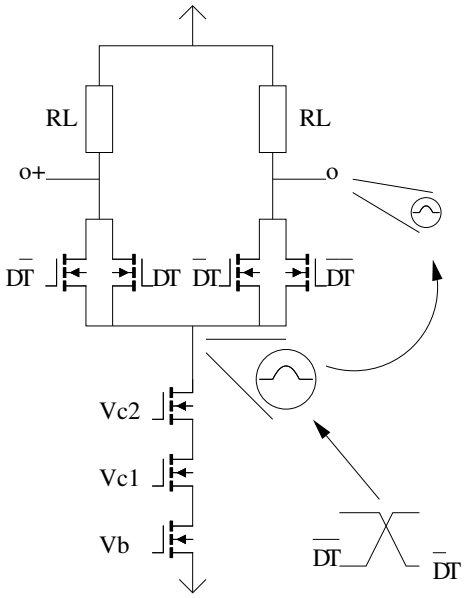

Figure 7.52 Switch driver coupling in differential-quad switching.

Next, let us define signals

$$s_p(n) = \left(2^N - 1\right)\left(\frac{1}{2} + \frac{1}{2}\sin(\Delta\phi n)\right) \tag{7.15}$$

$$s_n(n) = \left(2^N - 1\right)\left(\frac{1}{2} - \frac{1}{2}\sin(\Delta\phi n)\right) \tag{7.16}$$

$$\Delta s_p(n) = s_p(n) - s_p(n-1) = \Delta s(n) \tag{7.17}$$

$$\Delta s_n(n) = s_n(n) - s_n(n-1) = -\Delta s(n) \tag{7.18}$$

$$|\Delta s_n(n)| = |\Delta s_p(n)| \tag{7.19}$$

$$M_p(n) = s_p(n) + \frac{1}{2}\left(\Delta s_p(n) - |\Delta s_p(n)|\right) \tag{7.20}$$

$$M_n(n) = s_n(n) + \frac{1}{2}\left(\Delta s_n(n) - |\Delta s_n(n)|\right) \tag{7.21}$$

$$e_p(t) = K\frac{g_{ds}}{gm} M_p(n)\, s_p(n) \otimes p(t) \tag{7.22}$$

$$e_n(t) = K\frac{g_{ds}}{gm} M_n(n)\, s_n(n) \otimes p(t) \tag{7.23}$$

in which $s(n) = s(nT)$ is the small-signal equivalent of the output for a sinusoidal input (the "p" and "n" subscripts stands for positive and negative output, respectively), $\Delta s(n)$ corresponds to the discrete time derivative of the output, $M(n)$ is the number of unit currents in the thermometer-coded converter that does not change the output branch during the clock period, $p(t)$ is the pulse at the source node caused by the transition, gm and g_{ds} are the transconductance and output conductance of the switch, K is a constant that is used to model the attenuation of error from the source of the switches to the output, and $e(n)$ is the error seen at the output due to differential-quad switching.

Now it is possible to write an equation for the error signal seen at the differential output as

$$e(t)_{diff} = e_p(t) - e_n(t) \tag{7.24}$$

$$\begin{aligned} &= K\frac{g_{ds}}{gm}\Big(s_p(n)^2 - s_n(n)^2 \\ &\quad + \frac{1}{2}\left(s_p(n) + s_n(n)\right)\Delta s(n) \\ &\quad - \frac{1}{2}\left(s_p(n) - s_n(n)\right)|\Delta s(n)|\Big) \otimes p(t) \end{aligned} \tag{7.25}$$

$$\begin{aligned} &= K\frac{g_{ds}}{gm}\left(2^N - 1\right)^2\Big(\sin(\Delta\phi n) \\ &\quad + \frac{1}{2}\left(\sin(\Delta\phi n) - \sin(\Delta\phi(n-1))\right) \\ &\quad - \frac{1}{2}\left|\sin(\Delta\phi n) - \sin(\Delta\phi(n-1))\right|\sin(\Delta\phi n)\Big) \otimes p(t), \end{aligned} \tag{7.26}$$

which means that instability on the source node of the switches may cause odd-order harmonic distortion when differential-quad switching is used. However, since the distortion is proportional to the output signal as is also the distortion due to the output impedance variation, these two sources of distortion are very difficult to distinguish from each other in any other way than disabling the differential-quad switching.

It can be seen from the experimental results presented later in this chapter, and from the simulation results presented in Section 6.7.4, that it is beneficial to use the differential-quad switching scheme even though it seems to reduce the $SFDR$ when the converter is simulated without the power-supply-rail parasitics, as in Fig. 7.50 and Fig. 7.51.

The static timing errors due to clock and signal path imbalances may introduce timing-related harmonic distortion as presented in [145]. In order to avoid these imbalances, the load of the switch drivers is equalized with dummy structures as presented in Fig. 7.53. The method results in an almost identical operating point, and thus in the C_{gs}, of the dummy and the MSB switches. The main drawback of the presented method is (in this particular case) the 12.79% increase in static power consumption. A similar method is also used in [146]. In addition to load equalization, the timing imbalance was reduced by using a tree-like clock and output signaling, ensuring that the path length from root to leaf is equal for each signal branch.

7.2.5 Logic for digital calibration

The principle of the digital calibration method used in this prototype is described in Section 6.4.

The block diagram of the required DSP for the digital calibration is presented in Fig. 7.54 The functionality of the logic is as follows. The ”calibration request signal” resets the state machine, which takes care of the execution of the calibration algorithm as described in Fig. 7.55.

After the calibration request, the state machine shuts down all the digital blocks that are not needed in calibration, connects the comparator to the output of the converter, and the first calibration cycle starts. The state machine sets the CAL input code and steps down the REF input codes starting from CAL with steps of 4 LSB. Each comparison is repeated a maximum of N times in order to average out the effect of noise. There is also the possibility of using a tunable threshold level for comparison in such a manner that, if the value of the result counter exceeds the positive or negative threshold, the decision is made, even though the number of comparisons performed is less than N (N is also programmable from 1 to 2^{30}). This speeds up the calibration, since the large differences in which the noise does not play an important role, are resolved faster than smaller differences.

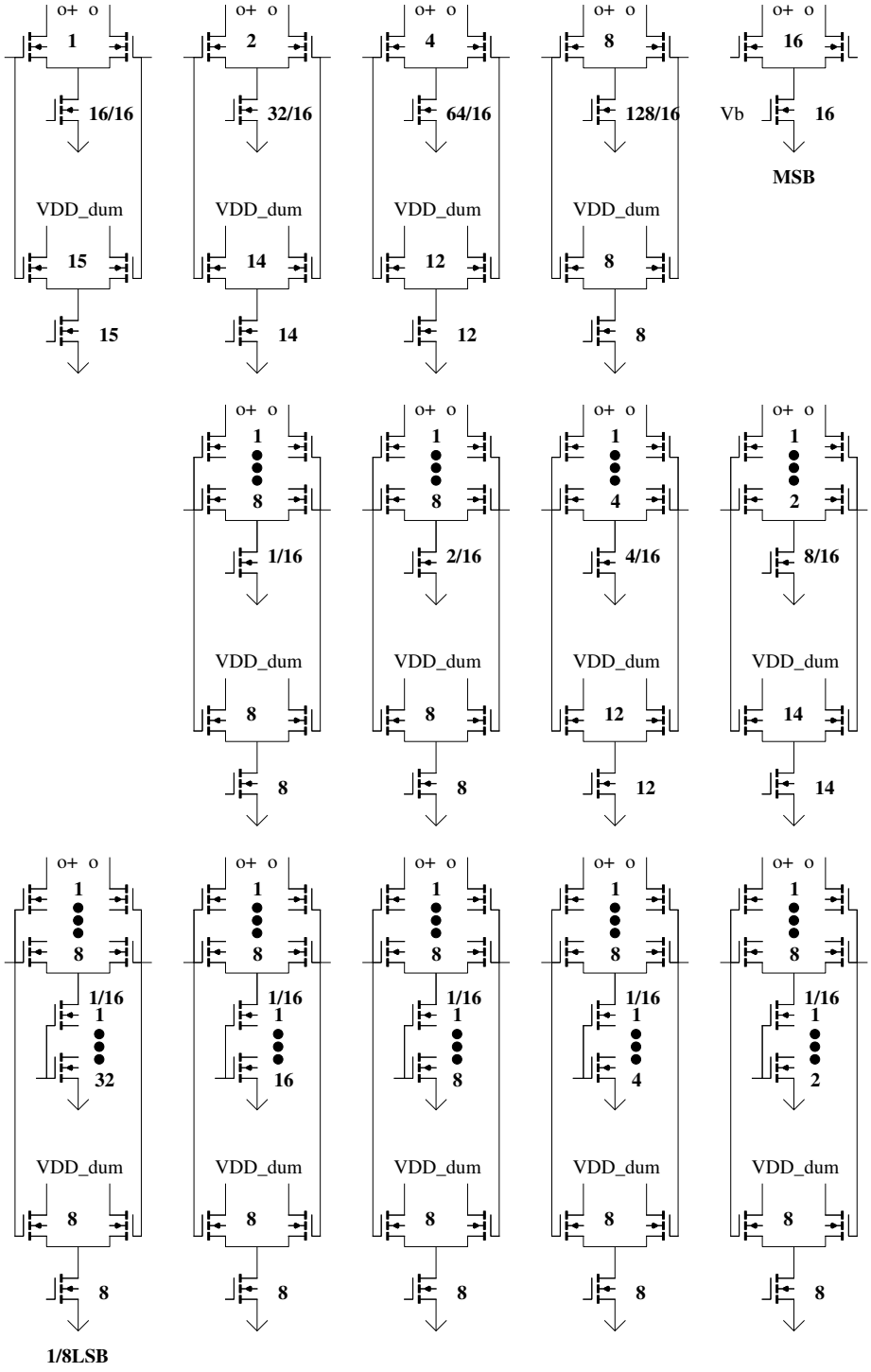

Figure 7.53 Switch driver load balancing with biased dummy structures.

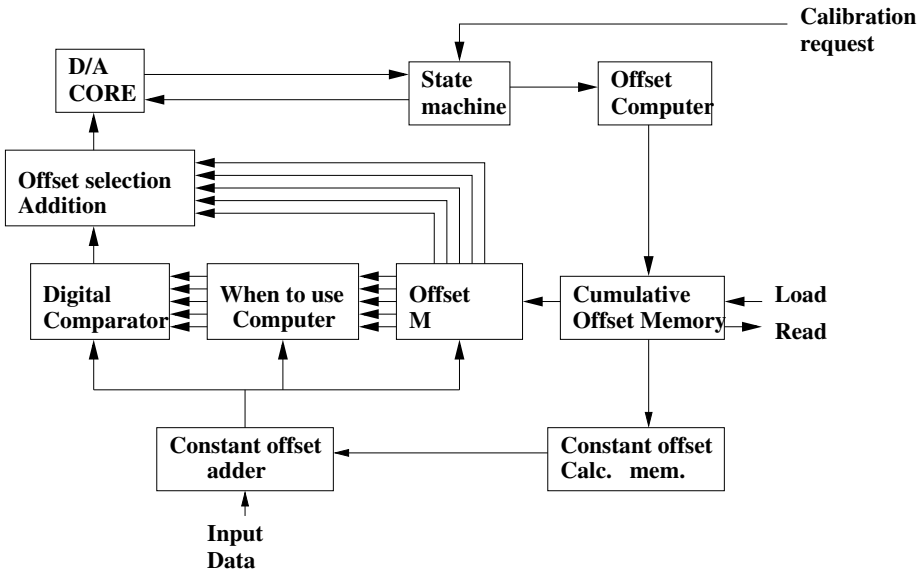

Figure 7.54 Block diagram of the calibration logic.

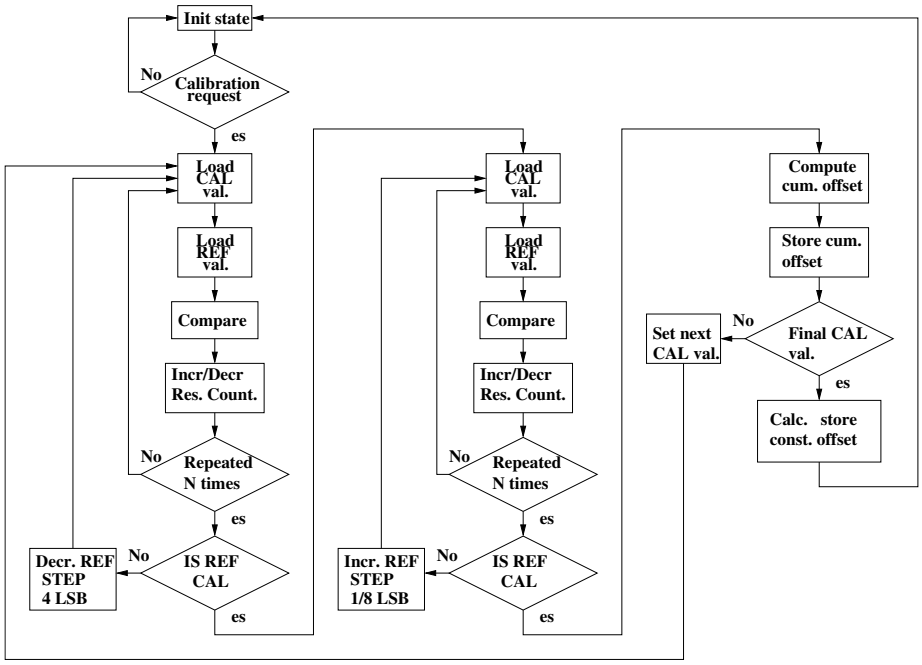

Figure 7.55 State diagram of the calibration.

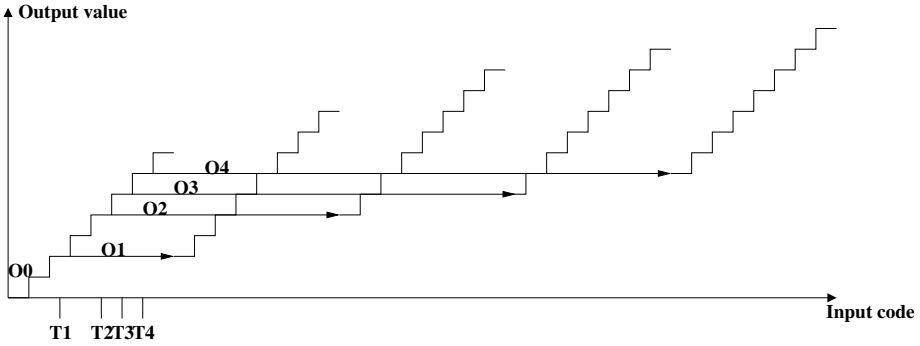

Figure 7.56 Offset possibilities for the 10-bit LSB segment.

Once the output corresponding to the REF-code is less than the output corresponding to the CAL code, the REF code is increased in steps of 1/8LSB, and the comparison is performed in a manner mentioned above. When the output corresponding to the REF value exceeds the output corresponding to the CAL, the offset is found with 1/8 LSB resolution, and the cumulative offset is computed by accumulating the current offsets with the previous ones.

For each of the CAL values corresponding to the MSB current source, the cumulative offset is stored to cumulative offset memory, which consists of 67 12-bit registers in series. RAM-memories could also be used, but the speed requirements could not be met with the memory elements provided by the silicon vendor, and the usage of the registers and multiplexers enabled the usage of pipelining and parallelism more freely.

After the offset for the last MSB source is found, a constant offset is computed and stored as

$$O_c = \frac{\left(2^{16} - 1 + 4 * 2^{10} + 0.875\right)\text{LSB} - O_{67}}{2}, \tag{7.27}$$

where O_{67} is the cumulative offset for the last MSB current source. The constant offset is used to align the input code range to the center of the analog output range, since the whole analog output range is not necessarily used.

After the calibration algorithm is executed, the state machine returns to initial state to wait for the calibration request, while the converter is in its normal operation mode, using stored cumulative offset values for the linearization.

The linearization of the conversion is performed in the normal operation mode as follows. For every input data value, the 67-to-5 offset MUX selects five possible offset values. This is due to the fact that the cumulative offset can be as large as $4\text{x}2^{10}\text{LSB} = 4\text{MSB}$, so for each 10-bit LSB segments there are five possible offset values, as presented in Fig. 7.56

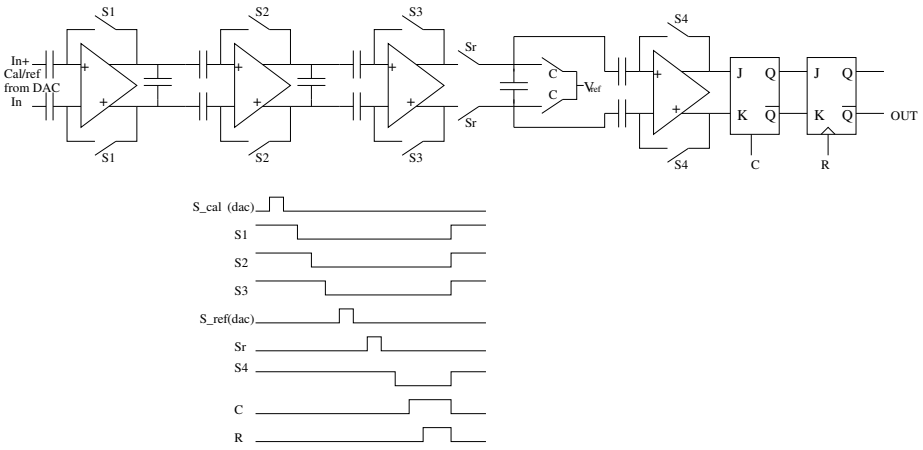

Figure 7.57 Comparator chain.

The when-to-use computer block is used to calculate the threshold values (T1-T4 in Fig. 7.56). The input code is compared to the when-to-use value, and the corresponding offset (O0-O4 in Fig. 7.56) is selected. It would be possible to also store the when-to-use values to memory, but since the memory is the speed bottleneck of the calibration system, it is justified to compute the when-to-use values separately for each input code rather than use memories, since the computation of when-to-use value requires only five subtracters and some registers for pipelining.

Finally, the digital comparator is used to compare the input to the when-to-use values, and the offset value is selected according to Fig. 7.56. The simulation result demonstrating the effect of the calibration was presented in Fig. 6.39.

7.2.6 DNL measurement for calibration

The key function of the digital calibration is the successive approximation A/D conversion of the error of the MSB current source, and the key component in the realization of the SAR A/D converter is the comparator. The comparator is realized as a chain of single-stage amplifiers with offset-compensation feedback (Fig. 7.57) In the design of this comparator chain, the following aspects are taken into account. What is required is a comparator that is able to provide a 3.3V output with a differential input voltage of 1.2μV, which corresponds to 1/16 LSB, half of the minimum step used in calibration. Thus, a gain of 128dB is needed.

The chain-of-amplifiers type of structure enables the offset voltage cancellation and amplification of the signal before the comparison in such a manner that the signal is amplified, but only the charge injection of the third amplifier stage in Fig. 7.57 affects the result of the comparison (offset and charge injection of the first and second stage

are canceled by the feedback of the third amplifier) [174]. Using multiple stages makes it also possible to use a simple single-stage differential amplifier as a comparator and obtain the required gain simultaneously. Since the single-stage amplifiers are inherently stable, no compensation is required, which would otherwise be the case due to the offset-cancellation unity feedback.

The one comparison cycle is performed as follows. First, the CAL value corresponding to the MSB current source to be calibrated (see Section 6.4 and Figs 7.55 7.57) is applied at the input of the comparator chain (i.e. output of the D/A converter) by triggering signal S_{CAL}. After that, the first three feedback loops are opened by opening the switches S_1, S_2, and S_3. Then the REF-value is applied to the input, triggered with signal S_{REF}. If the REF-value is larger than the CAL-value the output of the third comparator is positive, otherwise it is negative. The output of the third comparator is sampled with the signal S_r. Sampling of the difference of CAL and REF will alleviate the 1/f-noise requirements of the amplifiers, since it introduces a transmission zero on the noise transfer function at the zero frequency [175]. Furthermore the noise bandwidth is reduced by parallel capacitors at the amplifier outputs. Thermal noise is reduced by maximizing the *gm* of the input stage of the amplifier which also increases the gain. Increasing the dimensions of the input transistors above a certain point results in attenuation of the signal due to capacitive voltage division between the series capacitors and C_{gs} of the input stage of the amplifier.

After sampling, the feedback of the fourth comparator is opened, and the bottom-plate sampling is completed by closing the switch controlled by the signal C, which also enables the latching comparator. Finally, the differential output of the latch-comparator is read to JK-flip-flop with signal R.

The latch-comparator at the end of the chain is used to further boost the result of the comparison in order to ensure full rail-to-rail signal to be read to digital calibration circuitry. Circuit diagrams of the amplifier and the latch-comparator are presented in Fig. 7.58 and Fig. 7.59.

7.2.7 Experimental results

The circuit was fabricated on a 5 metal 0.35μm BiCMOS SiGE process with metal-insulator-metal capacitors. The photomicrograph of the circuit is presented in Appendix B. The chip was packed into a 160-pin TQFP package. The measurement board was a 4-layer FR4 PCB. Two inner layers were dedicated to ground and supply in such a manner that the ground plane was common to all analog and digital parts, whereas the positive supply voltages were distributed with separate supply planes. Decoupling capacitors were placed between the supplies and ground in order to stabilize the supply voltages. Tunable regulators were used to produce the supply voltages from a single

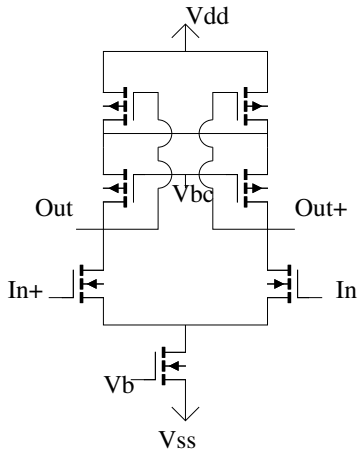

Figure 7.58 Single-stage differential amplifier.

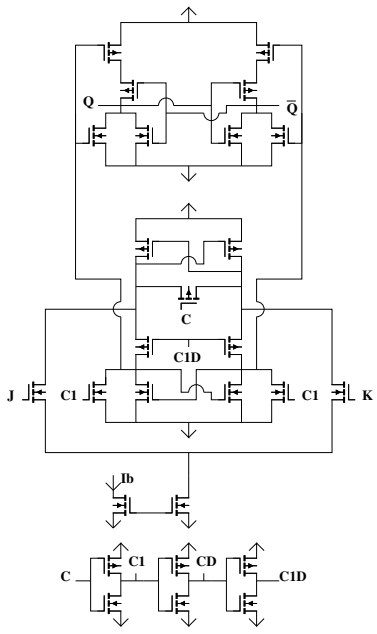

Figure 7.59 Comparator-latch.

supply voltage input. Area and power dissipation characteristics of the converter are presented in Tables 7.7 and 7.8.

Table 7.7 Area characteristics

	Calibration logic	D/A-core	Total
Area	5.83mm^2	5.77mm^2	14.40mm^2

Table 7.8 Power characteristics

Power dissipation:	Logic bypassed	Logic active
Digital@65MS/s	142mW	406mW
D/A core@65MS/s	95mW	95mW
Total 65MS/s	237mW	501mW
Digital@400MS/s	320mW	1449mW
D/A core@400MS/s	306mW	306mW
Total 400MS/s	626mW	1755mW

7.2.7.1 Static linearity

The static linearity of the uncalibrated converter was first measured in order to discover the linearity achieved by transistor dimensioning and layout techniques. The DNL and INL curves in the uncalibrated case are presented in Fig. 7.60. The maximum and minimum of the *DNL* are 2.018 LSB and -1.62 LSB, respectively. *INL* varies between 5.32 LSB and -7.51 LSB.

In order to discover the effect of calibration, offset is applied between the binary-weighted LSB and thermometer-coded MSB current source matrices. The static linearity of the uncalibrated D/A converter with -14 LSB offset between the LSB matrix and MSB matrix is presented in Fig. 7.61. In this case, *DNL* varies between 1.73 LSB and -15.29 LSB, and *INL* between 4.26 LSB and -18.93 LSB, respectively.

The effect of the calibration on the DNL can be clearly seen in Fig. 7.62. The variation of *DNL* is reduced to vary from 1.47 LSB to -1.41 LSB, while *INL* varies from 3.79 LSB to -10.96 LSB.

It seems that, even though *DNL* is reduced due to calibration, the *INL* is increased when compared to the results shown in Fig. 7.60. This is due to fact that *DNL* errors in Fig. 7.60 seem to compensate the accumulation of the *DNL* from the LSB matrix, whereas in Fig. 7.62, this does not happen. When it comes to the reduction of *DNL*, the prototype seems to function as designed; however, due to the limited resolution of the measurement, the absolute accuracy of the calibration cannot be determined.

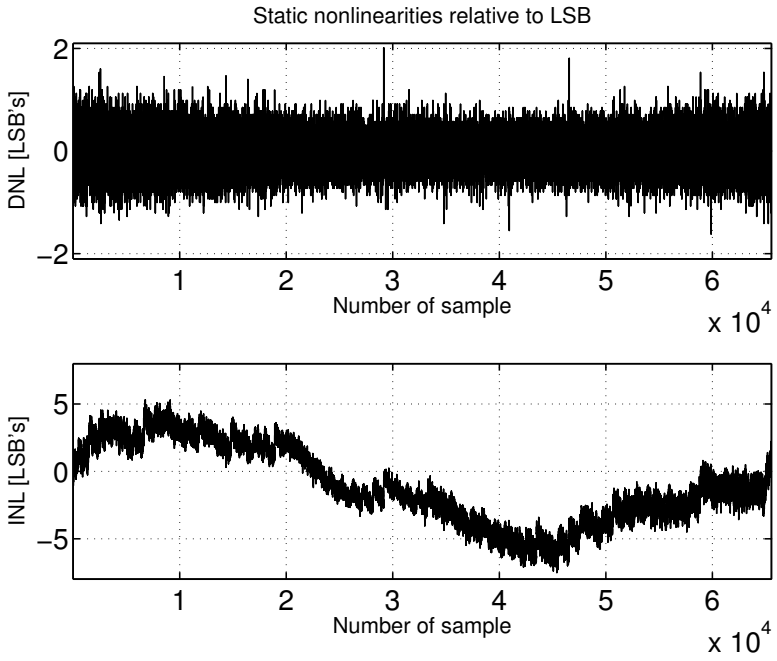

Figure 7.60 Static linearity of the uncalibrated D/A converter.

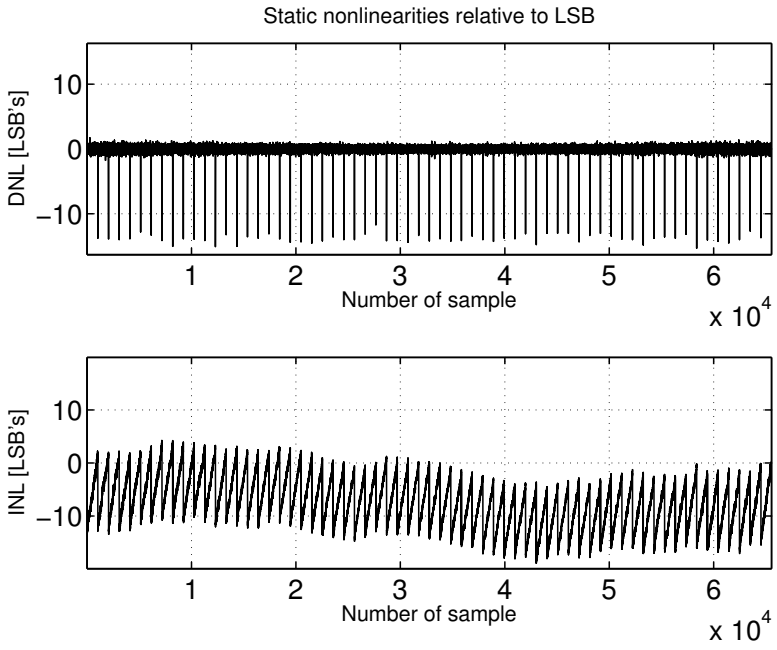

Figure 7.61 Static linearity of the uncalibrated D/A converter with -14 LSB offset between current source matrices.

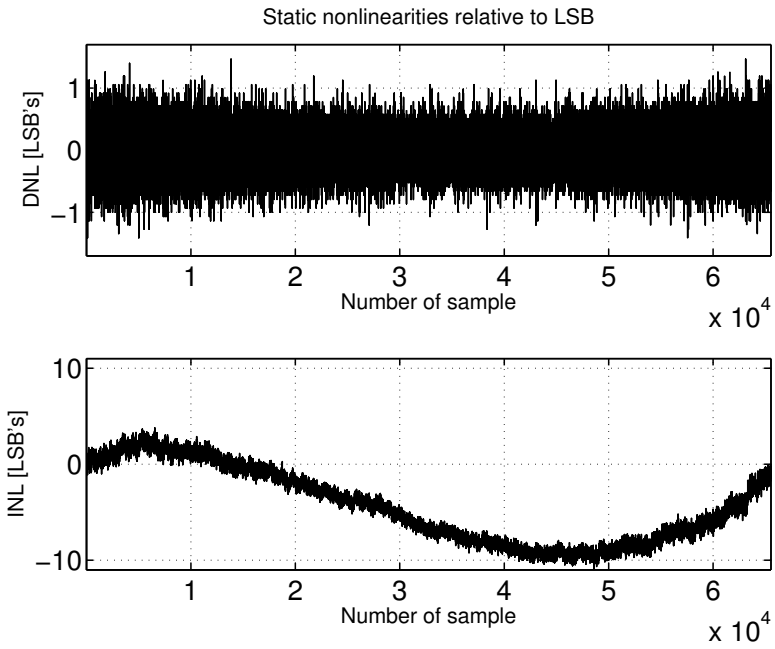

Figure 7.62 Static linearity of the calibrated D/A converter.

7.2.7.2 Dynamic performance

Fig. 7.63 and Fig. 7.64 represents the output spectrum of the converter at sample rate 65Ms/s and signal frequency 22.75MHz with the differential-quad switching disabled and enabled, respectively.

It can be observed that the level image of the third harmonic component is reduced 23.18dB , whereas the level of the second harmonic component remains the same. Spurious free dynamic range at a sample rate of 65MS/s as a function of signal frequency are presented in Figs 7.65 and 7.65.

Similar measurements were carried out at a 400Ms/s sample rate. Fig. 7.67 and Fig. 7.68 represent the output spectrum of the 140MHz sine signal with the differential-quad switching disabled and enabled, respectively.

It can be observed that the effect of the differential-quad switching on the third harmonic is less than in the case of the 65Ms/s sample rate. Enabling the toggler also increases the other spurious tones. This is mainly due to the increased activity of the switch driver circuitry, which, together with the too-narrow power supply rails, causes an increase in the power-rail interference. The effect of the power-supply-rail interference-induced jitter was analyzed in Section 6.7.4. Parasitic resistance of the power supply rails explains a large part of the degradation of the $SFDR$ as can be observed from the simulation results with estimated parasitic resistances presented in

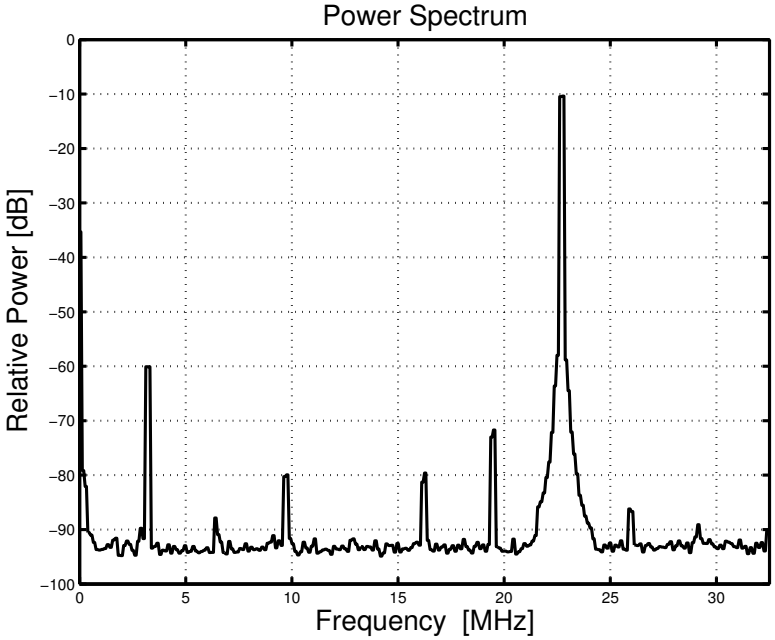

Figure 7.63 Output spectrum of a 22.75MHz sine signal with sampling frequency of 65MS/s and differential-quad switching disabled.

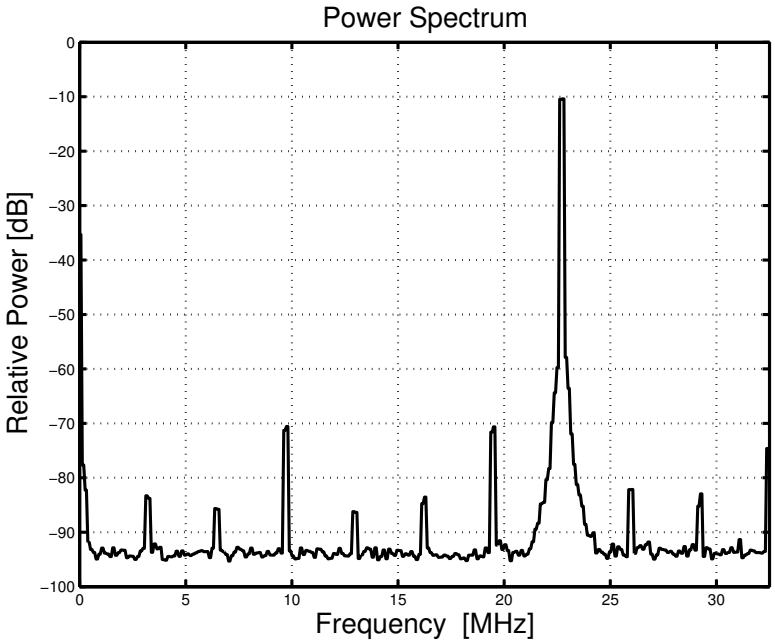

Figure 7.64 Output spectrum of a 22.75MHz sine signal with sampling frequency of 65MS/s and differential quad switching enabled.

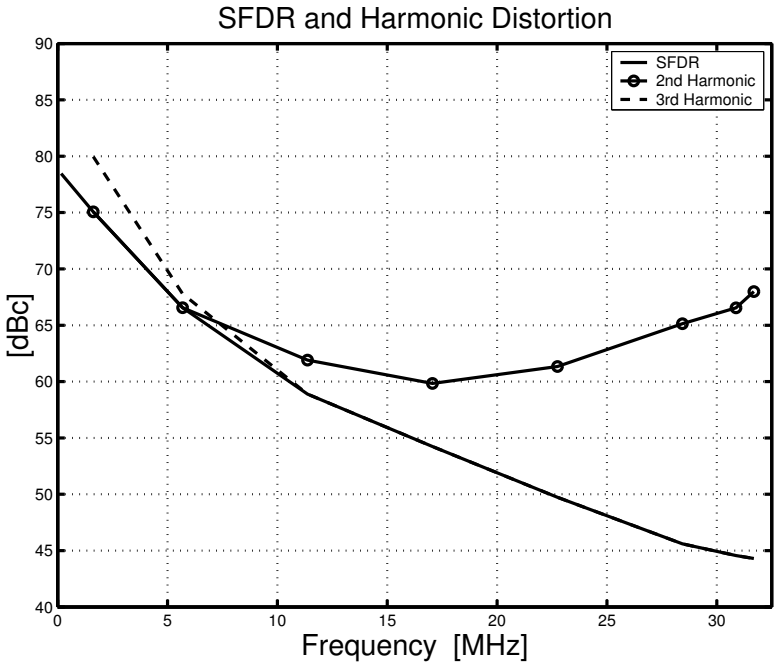

Figure 7.65 Spurious free dynamic range of the converter sampling frequency of 65MS/s and differential quad switching disabled.

Fig. 7.69 and Fig. 7.70.

The measured *SFDR* curves at 400MS/s sample rate are presented in Fig. 7.71 and Fig. 7.72.

7.2.7.3 Summary

The measured performance of the digital calibration algorithm demonstrates that relatively good static performance can be achieved by digital predistortion, even though the matching of the MSB current sources is initially poor. With the process used, the speed requirement of 400MHz is hard to achieve. Extensive parallelism, which increases the power dissipation of the circuit, had to be used. Parallelism also increases the area required for the calibration. However, in the future, the scaling of the silicon processes will increase the speed and decrease the size of the digital circuitry, whereas the area required for the current sources will scale down slower. Therefore the digital calibration may become an interesting option with the deep sub-micron silicon processes, in which fast and small memories and digital circuitry are available. Also scaling down the supply voltages along the scaling of the device size will make it more complicated to implement high performance analog circuitry, whereas it alleviates the implementation of the digital part. This work reveals that the proposed digital calibration algorithm

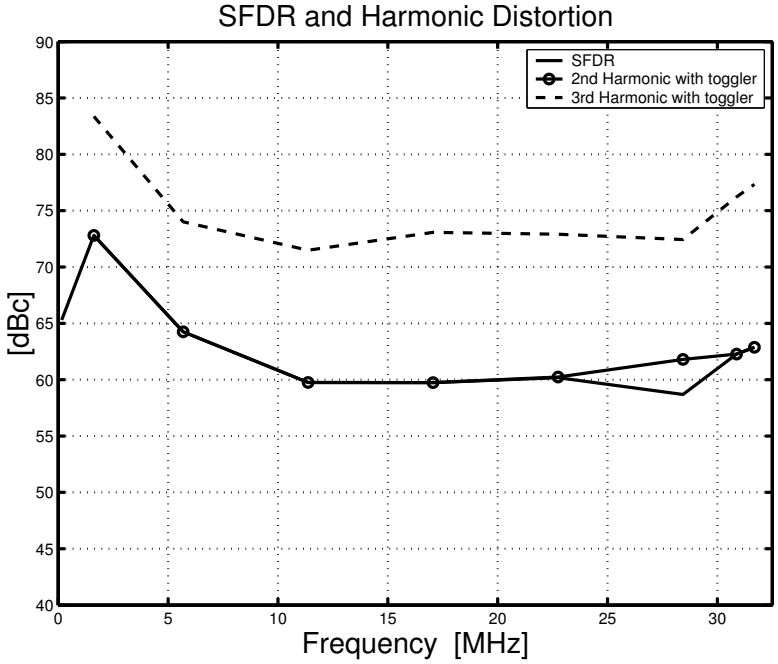

Figure 7.66 Spurious free dynamic range of the converter with sampling frequency of 65MS/s and differential quad switching enabled.

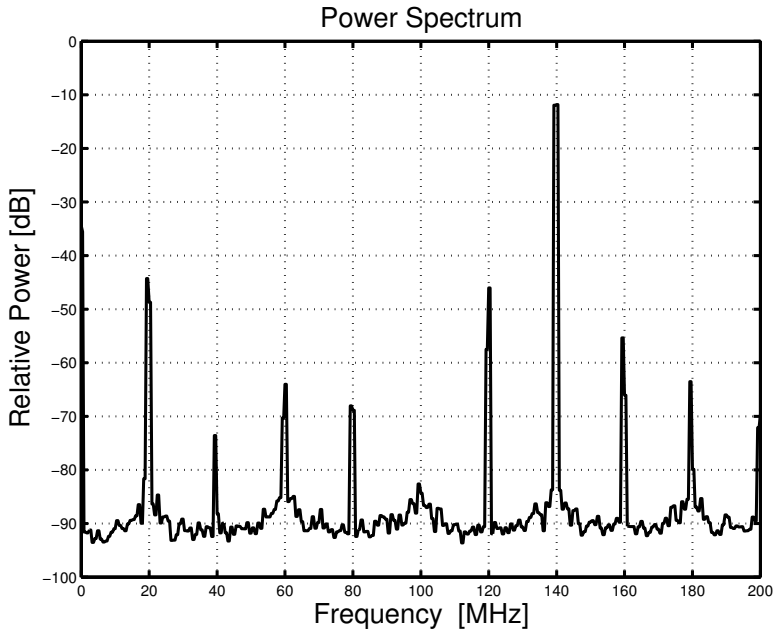

Figure 7.67 Output spectrum of 140MHz sine signal with sampling frequency of 400MS/s and differential-quad switching disabled.

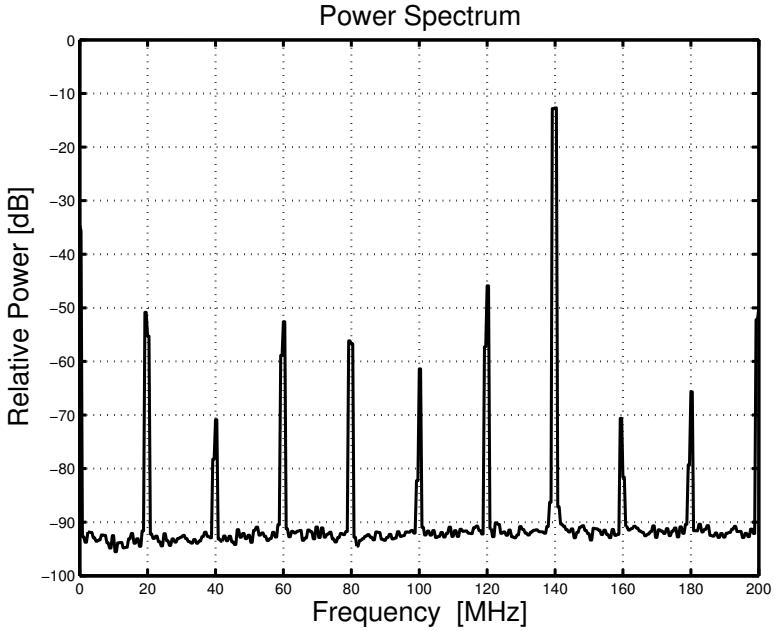

Figure 7.68 Output spectrum of 140MHz sine signal with sampling frequency of 400MS/s and differential quad switching enabled.

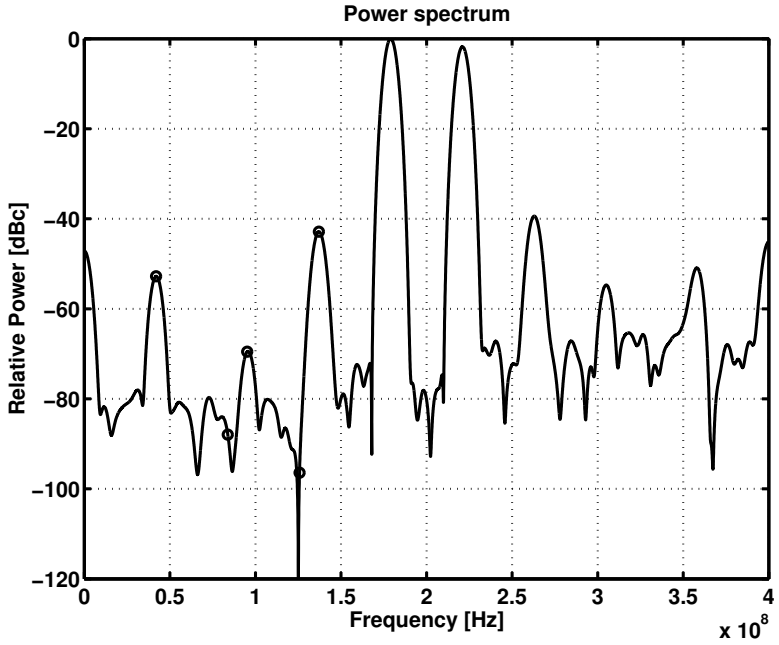

Figure 7.69 Simulated spectrum with estimated parasitic power supply resistances and differential-quad switching disabled.

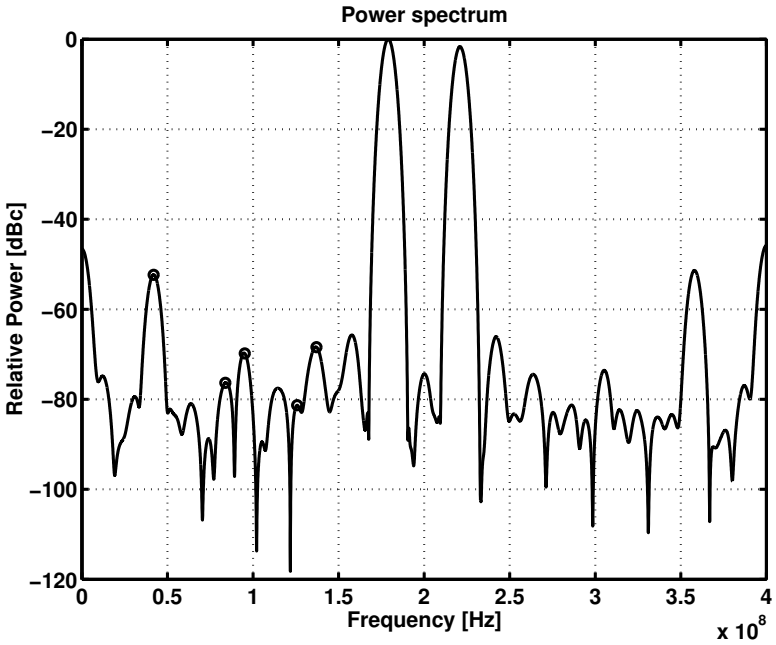

Figure 7.70 Simulated spectrum with estimated parasitic power supply resistances and differential-quad switching enabled.

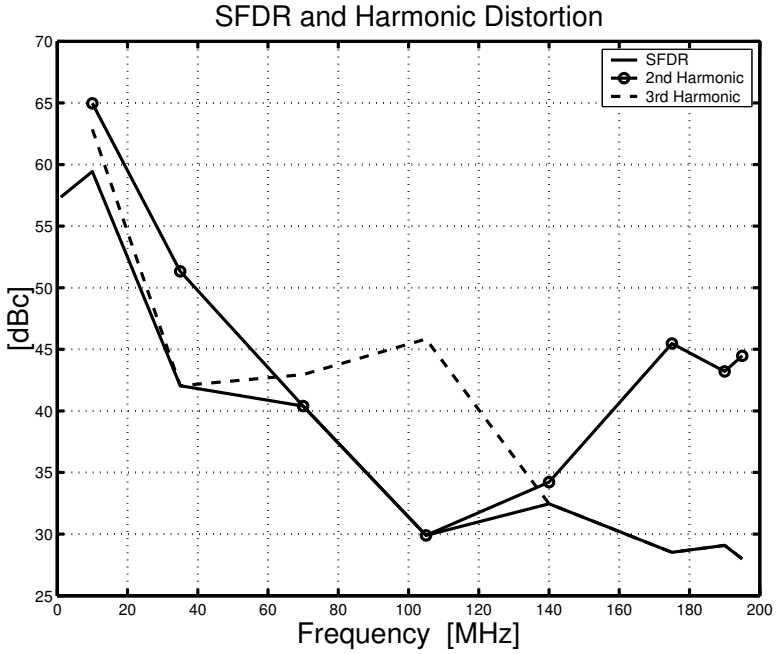

Figure 7.71 Spurious free dynamic range of the converter with sampling frequency of 400MS/s and differential quad switching disabled.

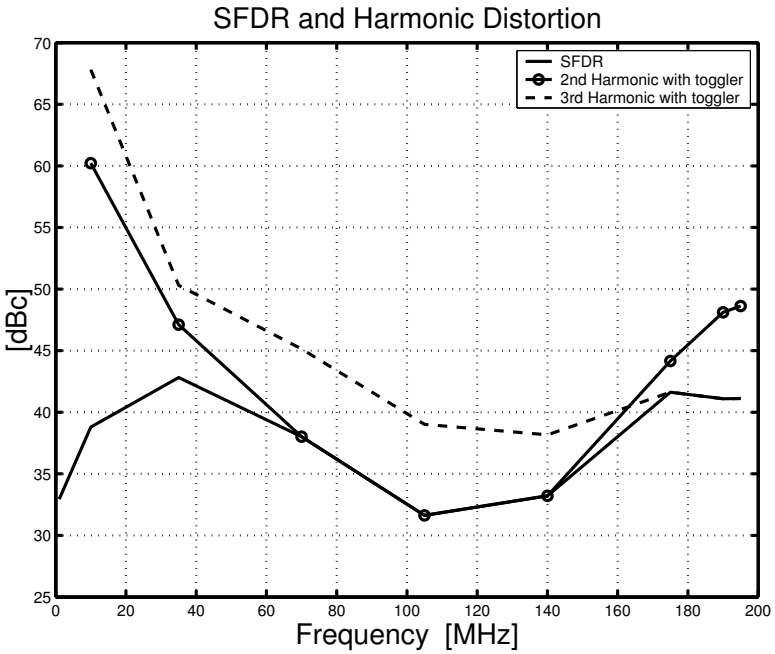

Figure 7.72 Spurious free dynamic range of the converter with sampling frequency of 400MS/s and differential quad switching enabled.

is a feasible way to perform the calibration of a current steering D/A converter.

The main shortcoming of the prototype is the poor $SFDR$ with a 400MS/s sampling rate. An adequate level of $SFDR$ is achieved with 65 MS/s. The measured dynamic performance of the converter differs from the simulated initial performance (Fig. 7.51 and Fig. 7.50) quite a lot due to the fact that the parasitic resistances were underestimated in the layout design phase, resulting in an excessive amount of signal-dependent jitter.

The performance degradation was analyzed with simulations including estimates of post-layout parasitic resistances and capacitances. The post-measurement simulations and simulation results match quite well, indicating that the parasitics are the source of the degradation.

Also, the effect of differential-quad switching was also verified with simulations and measurements revealing its effectiveness in reducing odd-order harmonic distortion.

The quality of performance of the implemented prototype and the recently published converters are presented in Table 7.9.

Table 7.9 Comparison of the prototype and recently published D/A converters

Publication	Process	Bits	Sampl. freq. [MHz]	SFDR [dBc] @ [MHz]	DNL max/ min	INL max/ min	Comment
Nakamura [154]	1μm CMOS	10	70	?	0.5/ -0.5	0.5/ -0.5	
Mercer [160]	2μm Bi-CMOS	16	40	80 @ 1.23	3/ -3	4/ -4	SFDR @ 10MS/s
Lin [126]	0.35μm CMOS	10	500	51 @ 240	0.05/ -0.1	0.2/ -0.2	
Bastos [155]	0.5μm CMOS	12	300	40 @ 60	0.3/ ?	0.5/ -0.6	No minmax DNL
Bugeja1 [161]	0.5μm CMOS	14	100	74 @ 8.5	0.5/ -0.5	0.5/ -0.5	
Van der Plas [158]	0.5μm CMOS	14	150	61 @ 5	0.15/ -0.1	0.3/ -0.25	
Bugeja2 [162]	0.35μm CMOS	14	100	72 @ 42.5	0.25/ -0.25	0.35/ -0.35	
Van den Bosch1 [164]	0.35μm CMOS	10	1000	61.2 @ 490	0.14/ 0.08	0.18/ -0.15	
Van den Bosch2 [163]	0.25μm CMOS	12	500	62 @ 125	0.2/ -0.25	0.3/ -0.25	
Tiilikainen [137]	0.18μm CMOS	14	100	64 @ 1	0.5/ -0.5	0.5/ -0.5	
Cong [138]	0.13μm CMOS	14	180	50 @ 63	0.4/ -0.2	0.2/ -0.4	
Schofield [139]	0.25μm CMOS	16	400	73 @ 190	0.2/ -0.3	0.2/ -0.7	
Hyde [165]	0.18μm CMOS	14	300	71 @ 120	0.4/ -0.3	0.3/ -0.3	
O'Sullivan [166]	0.18μm CMOS	12	320	60 @ 60	0.3/ -0.3	0.3/ -0.3	
Huang1 [167]	0.18μm CMOS	14	200	60 @ 90	0.6/ -0.4	0.6/ -0.6	RZ-mode
Schafferer [168]	0.18μm CMOS	14	1400	67 @ 260	0.8	1.8	No minmax DNL/INL
Doris [146]	0.18μm CMOS	12	500	60 @ 220	0.6/ -0.6	1/ -1	
This work	0.35μm CMOS	16	65	62.9 @ 31.68	1.47/ -1.41	3.79/ -10.96	Calibrated DNL/INL
This work	0.35μm CMOS	16	400	41.11 @ 195	2.02/ -1.62	5.32/ -7.51	Uncalibrated DNL/INL
This work					1.73/ -15.89	4.26/ -18.93	Worst case DNL/INL

Conclusions

Wireless communications have been the driving force of the development of the integrated circuits during the last decade. The trend has been to move from analog to digital signal processing and increase the bandwidth. It started with analog, while nowadays the first digital wireless systems like GSM are at the end of the trail, making way for 3G systems. The wireless digital communications have evolved through services such as GPRS and EDGE towards the WCDMA and 3G, which is capable of handling both the narrow voice band and wide data bands. Simultaneously, the wireless data transmission systems such as WLAN/WiFi and Wimax have gained popularity. In the world of multiple standards, it is beneficial if a single system is capable of handling multiple standards, or if the system is reconfigurable. This kind of flexibility can be obtained by using digital signal processing in transmitters and receivers. In this book, the research of digital signal processing and D/A converters for digital-IF transmitters is presented. The first part of the book represents the research results obtained during the design of the multi-carrier digital IF transmitter prototype for a WCDMA base-station transmitter. The theory of the transmitters and the signal processing, and efficient DSP realizations, was studied. The knowledge gained was then applied to the practical design, and finally the experimental results were presented.

The transmitter prototype consists of an interpolate-by-16 interpolation filter chain including a root-raised cosine pulse shaping filter and three half-band filters. The root raised cosine filter was designed with the Langrange-optimization algorithm in order to achieve adequate EVM and $ALCR$ for the 3G WCDMA system. Polyphase structures, CSD-multipliers, common subexpression sharing and time interleaving was used in order to reduce the amount of hardware in the filters.

Upconversion to the digital IF frequency was performed with a digital modulator based on a CORDIC vector rotation algorithm, which is very suitable for VLSI implementation and performs the modulation without digital multipliers, which are usually considered area consuming. After upconversion, the multicarrier signal was formed by summation of eight independent datapaths on four carrier frequencies, and the result was converted to the analog domain with a 14-bit current-steering D/A converter.

The SINC-attenuation due to the sample-and-hold function of the D/A converter is canceled with an inverse-SINC predistortion filter. Optimization of the dynamic range of the signal was also considered briefly. Further research into sophisticated signal clipping methods has been carried out at ECDL by Olli Väänänen, and published in [176].

Experimental results from the transmitter prototype indicate that the selected architecture is suitable for integrated digital IF multicarrier modulator, resulting in reasonable area and power dissipation, and that the quality requirements of the 3G WCDMA transmitter can be met. The performance of the transmitter is limited by the D/A converter. In the future, the requirement for high-quality high-speed D/A converters will increase, since more-and-more signal processing will be performed digitally.

The second part of the book is about the design of the current-steering D/A converters. The current-steering D/A converter is a complex entity composed of various details, all of which have to be considered and designed carefully in order to achieve good performance. The theory part of this book represents the problem scope related to static linearity, dynamic linearity and timing jitter. The model for the static linearity was developed and the effect of the output impedance was analyzed. A digital calibration method based on predistortion was developed and implemented.

It was demonstrated with the prototype that digital predistortion is one possible way to realize the calibration of the current-steering D/A converter. This method has the advantage of having practically no matching requirement for the additional analog current sources, as is the case when a dedicated calibration DAC is used to tune the values of the MSB current source. The main difficulty is the measurement of the nonlinearity and the realization of the DSP required for the predistortion, whereas the requirements for the matching (and thus area) of the MSB sources are alleviated. In the future, the size of the digital part will scale down and the speed will increase, whereas the analog signal processing will become more-and-more challenging making digital predistortion technique more-and-more attractive.

The future of the IC's seems to be mainly digital, due to the decreasing cost of digital signal processing. However, the real world is analog and the conversion between the two worlds cannot be avoided. This will increase the demand for the various digital enhancement algorithms of analog-signal processing.

Bibliography

[1] J. Vankka, M. Kosunen, and K. Halonen, "A multicarrier QAM modulator," in *Proc. Int. Symp. Circuits Syst.*, vol. 4, 30 May-2 Jun. 1999, pp. 415–418.

[2] J. Vankka, M. Kosunen, J. Hubach, and K. Halonen, "A CORDIC based multicarrier QAM modulator," in *Proc. Global Telecommunications Conference*, vol. 1A, 1999, pp. 173–177.

[3] M. Kosunen, J. Vankka, and K. Halonen, "A multicarrier QAM modulator for WCDMA basestation," in *Proc. Int. Symp. Circuits Syst.*, vol. 3, 28-31 May 2000, pp. 105–108.

[4] ——, "A multicarrier QAM-modulator for WCDMA basestation using interleaved FIR-filter structure," in *Proc. International Symposium on Intelligent Signal Processing and Communication Systems*, 5-8 Nov. 2000, pp. 1000–1005.

[5] ——, "A digital carrier synthesizer and modulator for WCDMA basestation," in *Proc. Nordic Signal Processing Symposium*, 6-7 Nov. 2000, pp. 93–96.

[6] J. Vankka, M. Kosunen, I. Sanchis, and K. A. I. Halonen, "A multicarrier QAM modulator," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 1, pp. 1–10, Jan. 2000.

[7] L. Koskinen, M. Kosunen, S. Lindfors, and K. Halonen, "Low-power decimation and matched channel selection filter for a WCDMA receiver," in *Proc. International Symposium on Intelligent Signal Processing and Communication Systems*, vol. 2, 5-8 Nov. 2000, pp. 819–822.

[8] ——, "Matched filter for a low-power oversampling WCDMA receiver architecture," in *Proc. Nordic Signal Processing Symposium*, 6-7 Nov. 2000, pp. 437–440.

[9] ——, "Truncation DC-error elimination in FIR filters," in *IEEE Midwest Symposium on Circuits and Systems*, vol. 3, 8-11 Aug. 2000, pp. 1292–1295.

[10] M. Kosunen, J. Vankka, M. Waltari, and K. Halonen, "A multicarrier QAM-modulator for WCDMA basestation with on-chip D/A converter," in *Proc. Custom Integrated Circuits Conf.*, vol. 1, 6-9 May 2001, pp. 301–304.

[11] ——, "Integrated digital modulator and D/A converter for WCDMA base-station," in *Proc. Nordic Signal Processing Symposium*, 12-13 Nov. 2001, pp. 291–296.

[12] ——, "A multicarrier QAM -modulator for WCDMA base-station with on-chip D/A converter," *IEEE Trans. VLSI Syst.*, vol. 13, no. 2, pp. 181–190, Feb. 2005.

[13] M. Kosunen, J. Vankka, I. Teikari, and K. Halonen, "DNL and INL yield models for a current-steering D/A converter," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 25-28 May 2003, pp. 969–972.

[14] J. Pirkkalaniemi, M. Kosunen, M. Waltari, and K. Halonen, "A digital calibration for a 16-bit, 400-MHz current-steering DAC," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 23-26 May 2004, pp. 297–300.

[15] M. Kosunen and K. Halonen, "Sampling jitter and power supply interference in current-steering D/A converters," in *Proc. European Conf. on Circuit Theory and Design*, vol. 1, 28 Aug.-1 Sep. 2005, pp. 305–308.

[16] J. Vankka, J. Ketola, O. Väänänen, J. Sommarek, M. Kosunen, and K. Halonen, "A GSM/EDGE/WCDMA modulator with on-chip D/A converter for base station," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, vol. 1, 3-7 Feb. 2002, pp. 236–463.

[17] J. Vankka, J. Ketola, J. Sommarek, O. Väänänen, M. Kosunen, and K. A. I. Halonen, "A GSM/EDGE/WCDMA modulator with on-chip D/A converter for base stations," *IEEE Trans. Circuits Syst. II*, vol. 49, no. 10, 2002.

[18] R. Peterson, R. Ziemer, and D. Borth, *Introduction to Spread Spectrum Communications*. Englewood Cliffs, New Jersey: Prentice-Hall, 1995.

[19] A. B. Carson, *Communication Systems*. Singapore: McGraw-Hill, 1986.

[20] 3^{rd} Generation Partnership Project; Technical Specification Group Radio Access Networks, UTRA (BS) FDD; Radio Transmission and Reception, *3G TS 25.104 Version 3.2.0*, 3^{rd} Generation Partnership Project (3GPP), 1999.

[21] E. Lee and D. Messerschmitt, *Digital Communication*. USA: Kluwer academic publishers, 1994.

[22] B. Razavi, *RF Microelectronics*. New Jersey, USA: Prentice-Hall PTR, 1998.

[23] R. Jain, H. Samueli, P. T. Yang, C. Chien, G. G. Chen, L. K. Lau, B.-Y. Chung, and E. G. Cohen, "Computer-aided design of a BPSK spread-spectrum chip set," *IEEE J. Solid-State Circuits*, vol. 27, no. 1, pp. 44–58, Jan. 1992.

[24] B.-Y. Chung, C. Chien, H. Samueli, and R. Jain, "Performance analysis of an all-digital BPSK direct-sequence spread-spectrum if receiver architecture," *IEEE J. Select. Areas Commun.*, vol. 11, no. 7, pp. 1096–1107, Sep. 1993.

[25] A. Abidi, "Direct-conversion radio transceivers for digital communications," *IEEE J. Solid-State Circuits*, vol. 30, no. 12, pp. 1399–1410, Dec. 1995.

[26] R. Joshi, P. Yang, H. Liu, K. Kindsfater, K. Cameron, D. Gee, H. Vu, G. Gorman, S. Tsai, A. Hung, R. Khan, O. Lee, S. Tollefsrud, E. Berg, J. Lee, T. Kwan, C. Lin, A. Buchwald, D. Jones, and H. Samueli, "A 52 Mb/s universal DSL transceiver IC," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 15-17 Feb. 1999, pp. 250–251, 465.

[27] *AD9853 Programmable Digital QPSK/16-QAM Modulator*, Analog Devices Inc., 1999.

[28] H. Suzuki and Y. Yamao, "Design of quadrature modulator for digital FM signaling with digital signal processing," *Electronics and Communications in Japan*, vol. 65-B, no. 9, pp. 66–73, 1982.

[29] K. Cho and H. Samueli, "A 8.75-MBaud single-chip digital QAM modulator with frequency-agility and beamforming diversity," in *Proc. Custom Integrated Circuits Conf.*, 21-24 May 2000, pp. 27–30.

[30] R. Marchesani, "Digital precompensation of imperfections in quadrature modulators," *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 552–556, Apr. 2000.

[31] A. Rofougaran, G. Chang, J. J. Rael, J. Y.-C. Chang, M. Rofougaran, P. J. Chang, M. Djafari, M.-K. Ku, E. W. Roth, A. A. Abidi, and H. Samueli, "A single-chip 900-MHz spread-spectrum wireless transceiver in 1-μm CMOS-Part I: Architecture and transmitter design," *IEEE J. Solid-State Circuits*, vol. 33, no. 4, pp. 515–534, Apr. 1998.

[32] Y. Chong, S. Kang, I. Lee, B. Kim, and H. Hong, "The implementation of a base station transceiver for WLL system based on wideband CDMA," in *Conference Record Int. Conf. on Communications*, vol. 1, 1-7 Jun. 1998, pp. 43–47.

[33] J.-S. Wu, M.-L. Liou, H.-P. Ma, and T.-D. Chiueh, "A 2.6-V, 44-MHz all-digital QPSK direct-sequence spread-spectrum transceiver IC," *IEEE J. Solid-State Circuits*, vol. 32, no. 10, pp. 1499–1510, Oct. 1997.

[34] X. H. Wang, X. Z. Qiu, J. Codenic, J. Vandewege, K. de Meyer, and W. Trog, "Digital if modulator and demodulator design for transmission over band-limited channels," in *Proc. Int. Conf. on Signal Processing*, vol. 2, 14-18 Oct. 1996, pp. 1242–1245.

[35] K. Seki, T. Sakata, and S. Kato, "A digitalized quadrature modulator for fast frequency hopping," *IEICE Transactions on Communications*, vol. E77-B, no. 5, pp. 656–662, May 1994.

[36] B. C. Wong and H. Samueli, "A 200-MHz all-digital QAM modulator and demodulator in 1.2-μm CMOS for digital radio applications," *IEEE J. Solid-State Circuits*, vol. 26, no. 12, pp. 1970–1980, Dec. 1991.

[37] H. Samueli and B. C. Wong, "A VLSI architecture for a high-speed all-digital quadrature ' modulator and demodulator for digital radio applications," *IEEE J. Select. Areas Commun.*, vol. 8, no. 8, pp. 1512–1519, Oct. 1990.

[38] G. Irvine, S. Herzinger, R. Schmidt, D. Kubetzko, and J. Fenk, "An up-conversion loop transmitter ic for digital mobile telephones," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 5-7 Feb. 1998, pp. 364–365, 465.

[39] W. T. Bax and M. A. Copeland, "A GSM modulator using $\Delta\Sigma$ frequency discriminator based synthesizer," in *Proc. Int. Symp. Circuits Syst.*, vol. 4, 31 May3 Jun. 1998, pp. 498–501.

[40] T. A. D. Riley and M. A. Copeland, "A simplified continuous phase modulator technique," *IEEE Trans. Circuits Syst. II*, vol. 41, no. 5, pp. 321–328, May 1994.

[41] T. Yamawaki, M. Kokubo, K. Irie, H. Matsui, K. Hori, T. Endou, H. Hagisawa, T. Furuya, Y. Shimizu, M. Katagishi, and J. R. Hildersley, "A 2.7-V GSM RF transceiver IC," *IEEE J. Solid-State Circuits*, vol. 32, no. 12, pp. 2089–2096, Dec. 1997.

[42] C. S. Kokourlis, P. H. Houlis, and J. N. Sahalos, "A general purpose differential digital modulator implementation incorporating a direct digital synthesis method," *IEEE Trans. Broadcast.*, vol. 39, no. 4, pp. 383–389, Dec. 1993.

[43] J. Haspelagh, D. Sallaerts, P. Reusens, A. Vanwelsenaers, R. Granek, and D. Rabaey, "A 270-kb/s 35-mW modulator IC for GSM cellular radio hand-held terminals," *IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1450–1457, Dec. 1990.

[44] L. R. Kahn, "Single-sideband transmission by envelope elimination," *Proc. IRE*, vol. 40, pp. 803–806, Jul. 1952.

[45] D. C. Cox, "Linear amplification with nonlinear components," *IEEE Trans. Commun.*, vol. 22, no. 12, pp. 1942–1945, Dec. 1974.

[46] P. B. Kenington, R. J. Wilkinson, and J. D. Marvill, "Broadband linear amplifier design for a PCN base-station," in *Proc. IEEE Vehicular Technology Conference*, 19-22 May 1991, pp. 155–160.

[47] J. Cavers, "The effect of quadrature modulator and demodulator errors on adaptive digital predistorters for amplifier linearization," *IEEE Trans. Veh. Technol.*, vol. 46, no. 2, pp. 456–466, May 1997.

[48] B. Razavi, "Challenges in the design of frequency synthesizers for wireless applications," in *Proc. Custom Integrated Circuits Conf.*, 5-8 May 1997, pp. 395–402.

[49] S. Wilingham, M. Perrot, B. Setterberg, A. Grzegorek, and B. McFarland, "An integrated 2.5 GHz $\Sigma\Delta$ frequency synthesizer with $5\mu s$ settling and 2 Mb/s closed loop modulation," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 7-9 Feb. 2000, pp. 200–201, 457.

[50] A. Linz and A. Hendricson, "Efficient implementation of an I-Q GMSK modulator," *IEEE Trans. Circuits Syst. II*, vol. 43, no. 1, pp. 14–23, Jan. 1996.

[51] L. Sundström, "Digital RF power amplifiers linearizers: Analysis and design," Ph.D. dissertation, Department of Applied Electronics, Lund University, Lund, Sweden, 1995.

[52] D. K. Su and W. J. McFarland, "An IC for linearizing RF power amplifiers using envelope elimination and restoration," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 2252–2258, Dec. 1998.

[53] B. Shi and L. Sundström, "A 200-MHz IF BiCMOS signal component separator for linear LINC transmitters," *IEEE J. Solid-State Circuits*, vol. 35, no. 7, pp. 987–993, Jul. 2000.

[54] H. Chireix, "High power outphasing modulation," *Proc. IRE*, vol. 23, pp. 1370–1392, Nov. 1935.

[55] T. W. Parks and J. H. McClellan, "Chebyshev-approximation for nonrecursive digital filters with linear phase," *IEEE Trans. Circuit Theory*, vol. 19, no. 2, pp. 189–194, Mar. 1972.

[56] ——, "A program for design of linear phase finite impulse response digital filters," *IEEE Trans. Audio. Electroacoust*, vol. 20, pp. 195–199, Aug. 1972.

[57] H. Samueli, "On the design of optimal equiripple FIR digital filters for data transmission applications," *IEEE Trans. Circuits Syst.*, vol. 35, no. 12, pp. 1542–1546, Dec. 1988.

[58] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing*, 3rd ed. Upper Saddle River, New Jersey: Prentice-Hall, 1996.

[59] G. Medlin, J. Adams, and C. Leondes, "Lagrange multiplier approach to the design of FIR filters for multirate applications," *IEEE Trans. Circuits Syst.*, vol. 35, no. 10, pp. 1210–1219, Oct. 1988.

[60] E. Hogenauer, "An economical class of digital filters for decimation and interpolation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 29, no. 2, pp. 155–162, Apr. 1981.

[61] S. Chu and C. S. Burrus, "Multirate filter design using comb filters," *IEEE Trans. Circuits Syst.*, vol. 31, no. 11, pp. 405–416, Nov. 1984.

[62] P. P. Vaidyanathan and T. Q. Nguyen, "'Trick' for the design of FIR half-band filters," *IEEE Trans. Circuits Syst.*, vol. 34, no. 3, pp. 297–300, Mar. 1987.

[63] L. R. R. Crochiere, "Interpolation and decimation of digital signals - a tutorial review," *Proc. IEEE*, vol. 69, no. 3, pp. 417–448, Mar. 1981.

[64] A. Avizienis, "Signed-digit number representation for fast parallel arithmetic," *IEEE Trans. Electron. Comput.*, pp. 389–400, Sep. 1961.

[65] C.-L. Chen and A. N. Willson, Jr, "A trellis search algorithm for the design of FIR filters with signed-powers-of-two coefficients," *IEEE Trans. Circuits Syst. II*, vol. 46, no. 1, pp. 29–39, Jan. 1999.

[66] D. Li, J. Song, and Y. C. Lim, "Polynomial-time algorithm for designing digital filters with power-of-two coefficients," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 3-6 May 1993, pp. 84–87.

[67] H. Samueli, "An improved search algorithm for the design of multiplierless fir filters with powers-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. 36, no. 7, pp. 1044–1047, Jul. 1989.

[68] M. Yagyu, A. Nishihara, and N. Fujii, "Design of FIR digital filters using estimates of error function over CSD coefficient space," *IEICE Transactions on Fundamentals*, vol. E79-A, no. 3, pp. 283–290, Mar. 1996.

[69] Q. Zhao and Y. Tadokoro, "A simple design of FIR filters with powers-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. 35, no. 5, pp. 566–570, May 1988.

[70] J. Laskowski and H. Samueli, "A 150 MHz 43-tap half-band FIR digital filter in $1.2-\mu m$ CMOS generated by silicon compiler," in *Proc. Custom Integrated Circuits Conf.*, 3-6 May 1992, pp. 11.4.1–11.4.4.

[71] Z. Jiang and A. N. Willson, Jr, "Efficient digital filtering architectures using pipelining/interleaving," *IEEE Trans. Circuits Syst. II*, vol. 44, no. 2, pp. 110–119, Feb. 1997.

[72] R. A. Hawley, B. C. Wong, T. Lin, J. Laskowski, and H. Samueli, "Design techniques for silicon compiler implementations of high-speed FIR digital filters," *IEEE J. Solid-State Circuits*, vol. 31, no. 5, pp. 656–667, May 1996.

[73] R. I. Hartley, "Subexpression sharing in filters using canonic signed digit multipliers," *IEEE Trans. Circuits Syst. II*, vol. 43, no. 10, pp. 677–688, Oct. 1996.

[74] A. G. Dempster and M. D. Mcleod, "Use of minimum-adder multiplier blocks in FIR digital filters," *IEEE Trans. Circuits Syst. II*, vol. 42, no. 9, pp. 569–577, Sep. 1995.

[75] B. P. Brandt and B. A. Wooley, "A low-power, area-efficient digital filter for decimation and interpolation," *IEEE J. Solid-State Circuits*, vol. 29, no. 6, pp. 679–687, Jun. 1994.

[76] L. Jackson, *Digital Filters and Signal Processing*, 2nd ed. Kluwer Academic Publishers, 1989.

[77] S. Y. Hwang, "Dynamic range constraint in state-space digital filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 23, no. 6, pp. 591–593, Dec. 1975.

[78] ——, "On monotonicity of l_p and l_p norms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 23, no. 6, pp. 593–594, Dec. 1975.

[79] H. T. Nicholas and H. Samueli, "An analysis of the output spectrum of direct digital frequency synthesizer in the presence of phase-accumulator truncation," in *Proc.* 41^{st} *Annual Frequency Control Symposium*, 1987, pp. 495–502.

[80] H. T. Nicholas, H. Samueli, and B. Kim, "The optimization of direct digital frequency synthesizer performance in the presence of finite word length effects," in *Proc.* 42^{nd} *Annual Frequency Control Symposium*, 1988, pp. 357–363.

[81] J. Tierney, C. M. Rader, and B. Gold, "A digital frequency synthesizer," *IEEE Trans. Audio. Electroacoust*, vol. 19, no. 1, pp. 48–57, Mar. 1971.

[82] F. Lu, H. Samueli, J. Yuan, and C. Svensson, "A 700-MHz 24-b pipelined accumulator in 1.2 *mum* CMOS for application as a numerical controlled oscillator," *IEEE J. Solid-State Circuits*, vol. 28, no. 8, pp. 878–885, Aug. 1993.

[83] D. A. Sunderland, R. A. Strauch, S. S. Wharfield, H. T. Peterson, and C. R. Cole, "CMOS/SOS frequency synthesizer LSI circuit for spread spectrum communications," *IEEE J. Solid-State Circuits*, vol. 19, no. 4, pp. 497–505, Aug. 1984.

[84] P. H. Saul and M. S. J. Mudd, "A direct digital synthesizer with 100-MHz output capability," *IEEE J. Solid-State Circuits*, vol. 23, no. 3, pp. 819–821, Jun. 1988.

[85] L. K. Tan and H. Samueli, "A 200 MHz quadrature digital synthesizer/mixer in 0.8 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 193–200, Mar. 1995.

[86] H. T. Nicholas and H. Samueli, "A 150 -MHz direct digital frequency synthesizer in 1.25 μm CMOS with -90-dBc spurious performance," *IEEE J. Solid-State Circuits*, vol. 26, no. 12, pp. 1959–1969, Dec. 1991.

[87] L. K. Lau, R. Jain, H. Samueli, H. T. Nicholas, and E. G. Cohen, "DDFSGEN a silicon compiler for direct digital frequency synthesizers," *Kluwer Academic Publishers, Journal of VLSI Signal Processing*, vol. 4, pp. 213–224, 1992.

[88] A. Yamagishi, M. Ishikawa, T. Tsukahara, and S. Date, "A 2-V 2-GHz low-power direct digital frequency synthesizer chip set for wireless communication," in *Proc. Custom Integrated Circuits Conf.*, 1-4 May 1995, pp. 319–321.

[89] S. Liao and L. Chen, "A low-power low-voltage direct digital frequency synthesizer," in *Proc. International Symposium on VLSI Technology, Systems, and Applications*, 1997, pp. 265–269.

[90] A. M. Sodagar and G. R. Lahiji, "Mapping from phase to sine-amplitude in direct digital frequency synthesizers using parabolic approximation," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 12, pp. 1452–1457, Dec. 2000.

[91] B.-D. Yang, J.-H. Choi, S.-H. Han, L.-S. Kim, and H.-K. Yu, "An 800-MHz low-power direct digital frequency synthesizer with an on-chip D/A Converter," *IEEE J. Solid-State Circuits*, vol. 39, no. 5, pp. 761–774, May 2004.

[92] F. Curticăpean, K. I. Palomäki, and J. Niittylahti, "Direct digital frequency synthesizer with high memory compression ratio," *IEE Electronics Letters*, vol. 37, no. 21, pp. 1275–1276, Oct. 2001.

[93] A. Bellaouar, M. S. O'Brecht, A. M. Fahim, and M. I. Elmasry, "Low-power direct digital frequency synthesis for wireless communications," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 385–390, Mar. 2000.

[94] J. M. P. Langlois and D. Al-Khalili, "Novel approach to the design of direct digital frequency synthesizers based on linear interpolation," *IEEE Trans. Circuits Syst. II*, vol. 50, no. 9, pp. 567–578, Sep. 2003.

[95] L. S. J. Chimakurthy, M. Ghosh, F. F. Dai, and R. C. Jaeger, "A novel DDS using nonlinear ROM addressing with improved compression ratio and quantization noise," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, vol. 53, no. 2, pp. 274–283, Feb. 2006.

[96] J. Vankka, M. Waltari, M. Kosunen, and K. A. I. Halonen, "A direct digital synthesizer with an on-chip D/A-converter," *IEEE J. Solid-State Circuits*, vol. 33, no. 2, pp. 218–227, Feb. 1998.

[97] L. K. Tan, E. W. Roth, G. E. Yee, and H. Samueli, "An 800-MHz quadrature digital synthesizer with ECL-Compatible output drivers in 0.8 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 30, no. 12, pp. 1463–1473, Dec. 1995.

[98] J. E. Volder, "The CORDIC trigonometric computing technique," *IRE Trans. Electron. Comput.*, pp. 330–334, Sep. 1959.

[99] N. Takagi, T. Asada, and S. Yajima, "Redundant CORDIC methods with a constant scale factor for sine and cosine computation," *IEEE Trans. Comput.*, vol. 40, no. 9, pp. 989–995, Sep. 1991.

[100] D. Timmermann, H. Hahn, and B. J. Hosticka, "Low latency time CORDIC algorithms," *IEEE Trans. Comput.*, vol. 41, no. 8, pp. 1010–1015, Aug. 1992.

[101] G. L. Haviland and A. A. Tuszynski, "A CORDIC arithmetic processor chip," *IEEE Trans. Comput.*, vol. 29, no. 2, pp. 68–79, Feb. 1980.

[102] J.-M. Muller, "Discrete basis and computation of elementary functions," *IEEE Trans. Comput.*, vol. 34, no. 9, pp. 857–862, Sep. 1985.

[103] G. Gielis, R. van de Plassche, and J. van Valburg, "A 540-MHz polar-to-cartesian converter," *IEEE J. Solid-State Circuits*, vol. 26, no. 11, pp. 1645–1650, Nov. 1991.

[104] A. Madisetti, A. Kwentus, and A. N. Willson, Jr, "A sine/cosine direct digital frequency synthesizer using an angle rotation algorithm," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 15-17 Feb. 1995, pp. 262–263.

[105] Y. H. Hu, "CORDIC-based VLSI architectures for digital signal processing," *IEEE Signal Processing Mag.*, vol. 9, no. 3, pp. 16–35, Jul. 1992.

[106] R. Sarmiento, F. Tobajas, V. de Armas, R. Esper-Chaín, J. F. López, J. A. Montiel-Nelson, and A. Núñez, "A CORDIC processor for FFT computation and its implementation using gallium arsenide technology," *IEEE Trans. VLSI Syst.*, vol. 6, no. 1, pp. 18–30, Mar. 1998.

[107] C. Mazenc, X. Merrheim, and J.-M. Muller, "Computing functions cos^{-1} and sin^{-1} using CORDIC," *IEEE Trans. Comput.*, vol. 42, no. 1, pp. 118–122, Jan. 1993.

[108] S. Nahm, K. Han, and W. Sung, "A CORDIC digital quadrature mixer: Comparison with a ROM-based architecture," in *Proc. Int. Symp. Circuits Syst.*, vol. 4, 31 May3 Jun. 1998, pp. 385–388.

[109] A. Torosyan, D. Fu, and A. N. Willson, Jr, "A 300-MHz quadrature direct digital synthesizer/mixer in 0.25μm CMOS," *IEEE J. Solid-State Circuits*, vol. 38, no. 6, pp. 875–887, Jun. 2003.

[110] Y. Song and B. Kim, "A quadrature digital synthesizer/mixer architecture using fine/coarse coordinate rotation to achieve 14-b input, 15-b output, and 100dbc SFDR," *IEEE J. Solid-State Circuits*, vol. 39, no. 11, pp. 1853–1861, Nov. 2004.

[111] S. Wang, V. Piuri, and E. E. Swartzlander, Jr, "Hybrid CORDIC algorithms," *IEEE Trans. Comput.*, vol. 46, no. 11, pp. 1202–1207, Nov. 1997.

[112] I. Janiszewski, B. Hoppe, and H. Meuth, "Numerically controlled oscillators with hybrid function generators," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, vol. 49, no. 7, pp. 995–1004, Jul. 2002.

[113] K. Kota and J. R. Cavallaro, "Numerical accuracy and hardware tradeoffs for CORDIC arithmetic for special-purpose processors," *IEEE Trans. Comput.*, vol. 42, no. 7, pp. 769–779, Jul. 1993.

[114] Y. H. Hu, "The quantization effects of the CORDIC algorithm," *IEEE Trans. Signal Processing*, vol. 40, no. 4, pp. 834–844, Apr. 1992.

[115] J. Vankka, "Direct digital synthesizers: Theory, design and applications," Ph.D. dissertation, Helsinki University of Technology, Espoo, Finland, Nov. 2000.

[116] D. De Caro, E. Napoli, and A. G. M. Strollo, "Direct digital frequency synthesizers with polynomial hyperfolding technique," *IEEE Trans. Circuits Syst. II*, vol. 51, no. 7, pp. 337–344, Jul. 2004.

[117] A. Madisetti, A. Y. Kwentus, and A. N. willson, Jr, "A 100-MHz, 16-b, direct digital frequency synthesizer with a 100-dBc Spurious-Free Dynamic Range," *IEEE J. Solid-State Circuits*, vol. 34, no. 8, pp. 1034–1043, Aug. 1999.

[118] F. Curticăpean, K. I. Palomäki, and J. Niittylahti, "Quadrature direct digital frequency synthesizer using an angle rotation algorithm," in *Proc. Int. Symp. Circuits Syst.*, vol. 2, 25-28 May 2003, pp. 81–84.

[119] Y. Song and B. Kim, "Quadrature direct digital frequency synthesizers using interpolation-based angle rotation," *IEEE Trans. VLSI Syst.*, vol. 12, no. 7, pp. 701–710, Jul. 2004.

[120] J. Vankka, J. Sommarek, J. Ketola, I. Teikari, and K. Halonen, "A digital quadrature modulator with on-chip D/A converter," *IEEE J. Solid-State Circuits*, vol. 38, no. 10, pp. 1635–1642, Oct. 2003.

[121] H. Scheuermann and H. Göckler, "A comprehensive survey of digital transmultiplexing methods," *Proc. IEEE*, vol. 69, no. 11, pp. 1419–1450, Nov. 1981.

[122] S. Mortezapour and E. K. F. Lee, "Design of low-power ROM-less direct digital frequency synthesizer using nonlinear digital-to-analog converter," *IEEE J. Solid-State Circuits*, vol. 34, no. 10, pp. 1350–1359, Oct. 1999.

[123] P. Hendriks, "Specifying communication DACs," vol. 34, no. 7, pp. 58–69, Jul. 1997.

[124] K. R. Lakshmikumar, R. A. Hadaway, and M. A. Copeland, "Characterization and modeling of mismatch in mos transistors for precision analog design," *IEEE J. Solid-State Circuits*, vol. 12, no. 6, pp. 1057–1066, Dec. 1986.

[125] M. J. M. Pelgrom, A. C. J. Duinmaijer, and A. P. G. Welbers, "Matching properties of MOS transistors," *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1440, Oct. 1989.

[126] C.-H. Lin and K. Bult, "A 10-b, 500-MSample/s CMOS DAC in 0.6 mm^2," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1948–1958, Dec. 1998.

[127] M. J. Schervish, "Multivariate normal probabilities with error bound," *Applied Statistics*, vol. 33, no. 1, pp. 81–94, 1984.

[128] ——, "Corrections: Algorithm AS 195:multivariate normal probabilities with error bound," *Applied Statistics*, vol. 34, no. 1, pp. 103–104, 1985.

[129] Z. Drezner, "Computation of the multivariate normal integral," *ACM Trans. on Mathematical Software*, vol. 18, no. 4, pp. 470–480, Dec. 1992.

[130] Y. Cong and R. L. Geiger, "Formulation of INL and DNL yield estimation in current-steering D/A converters," in *Proc. Int. Symp. Circuits Syst.*, vol. 3, 26-29 May 2003, pp. 149–152.

[131] K. R. Lakshmikumar, M. A. Copeland, and R. A. Hadaway, "Reply to "Comment on 'Characterization and modeling of mismatch in MOS transistors for precision analog design'"," *IEEE J. Solid-State Circuits*, vol. 23, no. 1, p. 296, Feb. 1988.

[132] A. Van den Bosch, M. Steyaert, and W. Sansen, "An accurate statistical yield model for CMOS current-steering D/A converters," in *Proc. Int. Symp. Circuits Syst.*, vol. 4, 28-31 May 2000, pp. 105–108.

[133] C. S. G. Conroy, W. A. Lane, and M. A. Moran, "A comment on "Characterization and modeling of mismatch in MOS transistors for precision analog design"," *IEEE J. Solid-State Circuits*, vol. 23, no. 1, pp. 294–296, Feb. 1988.

[134] D. W. J. Groeneveld, H. J. Schouvenaars, H. A. H. Termeer, and C. A. A. Bastiaansen, "A self-calibration technique for monolithic high-resolution D/A converters," *IEEE J. Solid-State Circuits*, vol. 24, no. 6, pp. 1517–1522, Dec. 1989.

[135] A. T. K. Tang and C. Toumazou, "Novel self-calibrated high-speed D/A converter using trimmable current sources," in *Proc. Int. Symp. Circuits Syst.*, vol. 5, 30 May2 Jun. 1994, pp. 469–472.

[136] Y. Manoli, "A self-calibration method for fast high-resolution A/D and D/A converters," *IEEE J. Solid-State Circuits*, vol. 24, no. 3, pp. 603–608, Jun. 1989.

[137] M. P. Tiilikainen, "A 14-bit 1.8-V 20-mW 1-mm^2 CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 36, no. 7, pp. 1144–1147, Jul. 2001.

[138] Y. Cong and R. L. Geiger, "A 1.5V 14b 100MS/s self-calibrated DAC," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 9-13 Feb. 2003, pp. 128–129.

[139] W. Schofield, D. Mercer, and L. S. Onge, "A 16b 400MS/s DAC with <-80dBc IMD to 300MHz and <-160dBm/Hz noise power spectral density," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 9-13 Feb. 2003, pp. 126–127.

[140] B. Razavi, *Principles of Data Conversion System Design*. USA: John Wiley & Sons, Inc, 1997.

[141] M. Gustavsson, J. Wikner, and N. Tan, *CMOS Data Converters for communications*. Kluwer Academic Publishers, 2000.

[142] A. Van den Bosch, M. Steyaert, and W. Sansen, "Sfdr-bandwidth limitations for high speed high resolution current steering CMOS D/A converters," in *Proc. Intl. Conf. on Electronics, Circuits and Systems*, vol. 3, 5-8 Sep. 1999, pp. 1193–1196.

[143] S. Luschas and H.-S. Lee, "Output impedance requirements for DACs," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 25-28 May 2003, pp. 861–864.

[144] P. E. Allen and D. R. Holberg, *CMOS Analog Circuit Design*. USA: Holt, Rinehart and Winston, 1987.

[145] T. Chen and G. Gielen, "Analysis of the dynamic SFDR property of high-accuracy current-steering D/A converters," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 25-28 May 2003, pp. 973–976.

[146] K. Doris, J. Briaire, D. Leenaerts, M. Vertregt, and A. van Roermund, "A 12b 500MS/s DAC with >70dB SFDR up to 120MHz in 0.18μm CMOS," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 6-10 Feb. 2005, pp. 116–117.

[147] K. Doris, A. van Roermund, and D. Leenaerts, "A general analysis on the timing jitter in D/A converters," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 26-29 May 2002, pp. 117–120.

[148] ——, "Mismatch-based timing errors in current steering DACs," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 25-28 May 2003, pp. 977–980.

[149] J. L. González and E. Alarcón, "Clock-jitter induced distortion in high speed CMOS switched-current segmented digital-to-analog converters," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 6-9 May 2001, pp. 512–515.

[150] K. Doris, D. Leenaerts, and A. van Roermund, "Time non linearities in D/A converters," in *Proc. European Conf. on Circuit Theory and Design*, vol. 3, 28-31 Aug. 2001, pp. 353–356.

[151] S. Park, G. Kim, S.-C. Park, and W. Kim, "A digital-to-analog converter based on differential-quad switching," *IEEE J. Solid-State Circuits*, vol. 37, no. 10, pp. 1335–1338, Oct. 2002.

[152] J. Bastos, M. S. J. Steyaert, A. Pergrot, and W. M. Sansen, "Influence of die attachment on MOS transistor mismatch," *IEEE Trans. Semiconduct. Manufact.*, vol. 10, no. 2, pp. 209–218, May 1997.

[153] T. Miki, Y. Nakamura, M. Nakaya, S. Asai, Y. Akasaka, and Y. Horiba, "An 80-MHz 8-bit CMOS D/A converter," *IEEE J. Solid-State Circuits*, vol. 12, no. 6, pp. 983–988, Dec. 1986.

[154] Y. Nakamura, T. Miki, A. Maeda, H. Kondoh, and N. Yawaza, "A 10-b 70-MS/s CMOS D/A converter," *IEEE J. Solid-State Circuits*, vol. 26, no. 4, pp. 637–642, Apr. 1991.

[155] J. Bastos, A. M. Marques, M. S. J. Steyaert, and W. Sansen, "A 12-bit intrinsic accuracy high-speed CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1959–1969, Dec. 1998.

[156] A. Van den Bosch, M. Borremans, J. Vandenbussche, G. Van der Plas, A. Marques, J. Bastos, M. Steyaert, G. Gielen, and W. Sansen, "A 12-bit 200 MHz low glitch CMOS D/A converter," in *Proc. Custom Integrated Circuits Conf.*, vol. 4, 11-14 May 1998, pp. 249–252.

[157] J. Deveugele, G. Van der Plas, M. Steyaert, G. Gielen, and W. Sansen, "A gradient-error and edge-effect tolerant switching scheme for a high-accuracy DAC," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 1, pp. 191–195, Jan. 2004.

[158] G. A. M. Van der Plas, J. Vandenbussche, W. Sansen, M. S. J. Steyaert, and G. G. E. Gielen, "A 14-bit intrisinc accuracy Q^2 *Random Walk* CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 34, no. 12, pp. 1708–1718, Dec. 1999.

[159] Y. Cong and R. L. Geiger, "Switching sequence optimization for gradient error compensation in thermometer-decoded DAC arrays," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 7, pp. 585–595, Jul. 2000.

[160] D. Mercer, "A 16-b D/A converter with increased spurious free dynamic range," *IEEE J. Solid-State Circuits*, vol. 29, no. 10, pp. 1180–1185, Oct. 1994.

[161] A. R. Bugeja, B.-S. Song, P. L. Rakers, and S. F. Gillig, "A 14-b, 100-MS/s CMOS DAC designed for spectral performance," *IEEE J. Solid-State Circuits*, vol. 34, no. 12, pp. 1719–1732, Dec. 1999.

[162] A. R. Bugeja and B.-S. Song, "A self-trimming 14-b 100-MS/s CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1841–1852, Dec. 2000.

[163] A. Van den Bosch, M. Borremans, M. Steyaert, and W. Sansen, "A 12-b 500MSample/s current-steering CMOS D/A converter," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 5-7 Feb. 2001, pp. 366–367,466.

[164] A. Van den Bosch, M. A. F. Borremans, M. S. J. Steyaert, and W. Sansen, "A 10-bit 1-GSample/s nyquist current-steering CMOS D/A converter," *IEEE J. Solid-State Circuits*, vol. 36, no. 3, pp. 315–324, May 2001.

[165] J. Hyde, T. Humes, C. Diorio, and M. Thomas, "A 300-MS/s 14-bit digital-to-analog coonverter in logic CMOS," *IEEE J. Solid-State Circuits*, vol. 38, no. 5, pp. 734–740, May 2003.

[166] K. O'sullivan, C. Gorman, and M. Hennessy, "A 12-bit 320-MSample/s current-steering CMOS D/A converter in 0.44 mm^2," *IEEE J. Solid-State Circuits*, vol. 39, no. 7, pp. 1064–1072, Jul. 2004.

[167] Q. Huang, P. A. Francese, C. Martelli, and J. Nielsen, "A 200MS/S 14b 97mw DAC in 0.18μm CMOS," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 15-19 Feb. 2004, pp. 364–365.

[168] B. Schafferer and R. Adams, "A 3v CMOS 400mW 14b 1.4GS/s DAC for multi-carrier applications," in *Proc. IEEE Intl. Solid-State Circuits Conf., Digest of Technical Papers*, 15-19 Feb. 2004, pp. 360–361.

[169] T. Lin and H. Samueli, "A 200-MHz CMOS x/sin(x) digital filter for compensating D/A converter frequency response distortion," *IEEE J. Solid-State Circuits*, vol. 26, no. 9, pp. 1278–1285, Sep. 1991.

[170] O. Väänänen, "Clipping in wideband CDMA base station transmitter," Master's thesis, Helsinki University of Technology, Espoo, Finland, Jun. 2001.

[171] O. Väänänen, J. Vankka, and K. Halonen, "Simple algorithm for peak windowing and its application in GSM, EDGE and WCDMA systems," *Proc. IEE Communications*, vol. 152, no. 3, pp. 357–362, Jun. 2005.

[172] J. Feltes, "Lisää käyttöä PC:n rinnakkaisportille," *Prosessori*, no. 5, pp. 60–63, 1996.

[173] J. A. Starzyk and R. P. Mohn, "Cost-oriented design of a 14-bit current steering DAC macrocell," in *Proc. Int. Symp. Circuits Syst.*, vol. 1, 25-28 May 2003, pp. 965–968.

[174] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*. New Jersey, USA: IEEE Press, 1995.

[175] C. C. Enz and G. C. Temes, "Circuit techniques for reducing the effects of op-amp imperfections: Autozeroing, correlated double sampling, and chopper stabilization," *Proc. IEEE*, vol. 84, no. 11, pp. 1584–1614, Nov. 1996.

[176] O. Väänänen, "Digital modulators with crest factor reduction techniques," Ph.D. dissertation, Helsinki University of Technology, Espoo, Finland, Mar. 2006.

Appendix A

Photomicrograph of the WCDMA transmitter

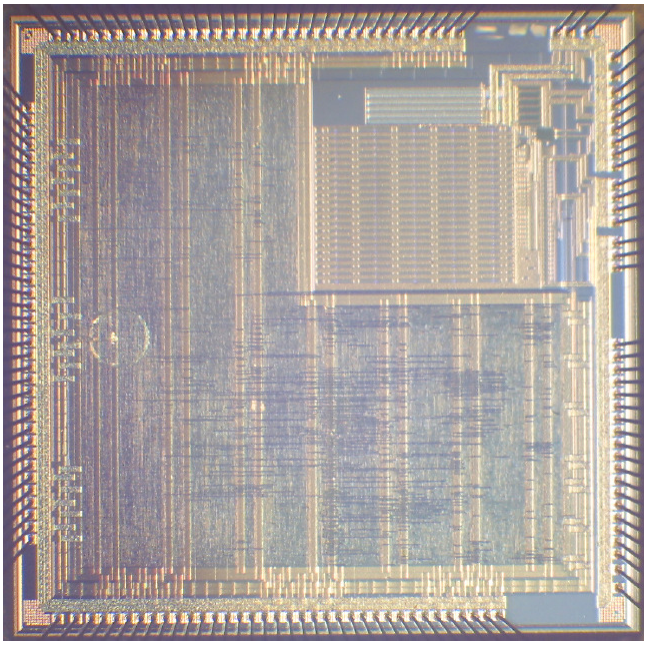

Figure A.1 Photomicrograph of the circuit.

Appendix B

Photomicrograph of the D/A converter with digital calibration

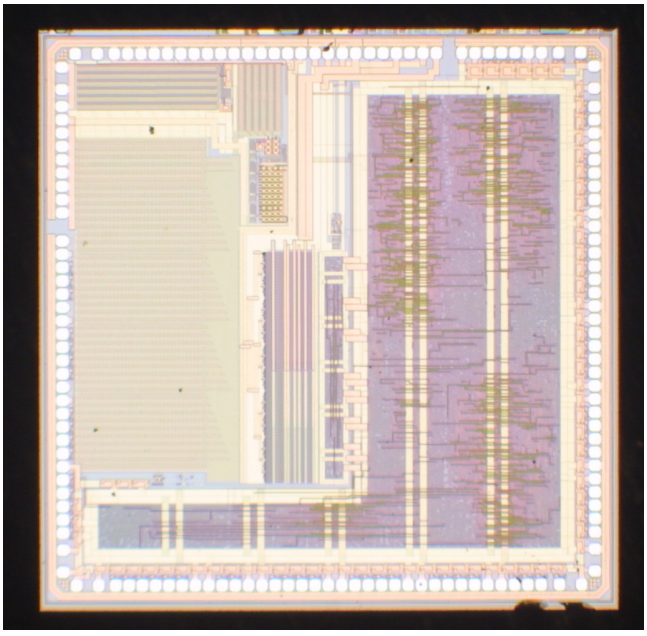

Figure B.1 Photomicrograph of the circuit.

Appendix C

Jitter energy as a function of signal frequency and jitter amplitude

The continuous time error signal $e(t)$ due to timing jitter is defined by the jitter signal $w(n)$ and the discrete-time derivative $\Delta x(n)$ of the signal $x(n)$ as

$$\begin{aligned} e(t) &= \sum_{n=-\infty}^{\infty} \Delta x(n)\, g(t) \\ &= \sum_{n=-\infty}^{\infty} [x(nT) - x((n-1)T)] \\ &\quad \times [u(t - nT - w(n))) - u(t - nT)], \end{aligned} \tag{C.1}$$

in which $\Delta x(n) = x(n) - x(n-1)$ and $u(t)$ is a unit step function. $w(n)$ has the following properties.

$$E[w(n)] = 0 \tag{C.2}$$

$$E[w(n) - w(n-1)] = 0, \tag{C.3}$$

since $E[w(n)] \neq 0$ results in time shift of the signal and $E[w(n) - w(n-1)] \neq 0$ is equal to change in sampling period T, therefore not affecting the energy of the signals. The power of the error signal due to jitter can be computed as

$$P_j = \lim_{C \to \infty} \frac{1}{C} \int_0^C \sum_{n=-\infty}^{\infty} \Delta x(n)^2 \times (u(t - nT - w(n)) - u(t - nT))^2 \, \mathrm{d}t \tag{C.4}$$

$$= \lim_{N\to\infty} \sum_{n=0}^{N} \frac{1}{NT} \Delta x(n)^2 \left| \int_{nT}^{nT+w(n)} (u(t-nT-w(n)) - u(t-nT))\,\mathrm{d}t \right| \quad \text{(C.5)}$$

$$= \frac{1}{T} E\left[\Delta x(n)^2 |w(n)|\right] \quad \text{(C.6)}$$

in which $E[]$ means expectation. By substituting

$$y = \Delta x(n)^2 - \mu_d \quad \text{(C.7)}$$

$$z = |w(n)| - \mu_a, \quad \text{(C.8)}$$

in which $\mu_d = E\left[\Delta x(n)^2\right]$ and $\mu_a = E[|w(n)|]$, Eq. C.6 can be written as

$$P_j = \rho\sigma_y\sigma_z + \mu_d\mu_a, \quad \text{(C.9)}$$

$$\rho = \frac{E[yz]}{\sigma_y\sigma_z}. \quad \text{(C.10)}$$

In case correlation factor $\rho = 0$, $\Delta x(n)^2$ and $|w(n)|$ are independent resulting in

$$P_j = \mu_d\mu_a = E\left[\Delta x(n)^2\right] E[|w(n)|]. \quad \text{(C.11)}$$

Let us assume, that $x(n)$ is a sinusoidal signal

$$x(n) = A\sin(\Delta\phi n) \quad \text{(C.12)}$$

$$\Delta\phi = \frac{2\pi F_{sig}}{F_s}, \quad F_{sig} \le F_s/2, \quad \text{(C.13)}$$

and $w(n)$ is an arbitrary jitter signal independent of $x(n)$ In this case the power of the error signal due to jitter can be computed as

$$P_j = E\left[\Delta x^2\right] E[|x|] \quad \text{(C.14)}$$

$$= A^2(1-\cos(\Delta\phi)) E\left[\frac{|w(n)|}{T}\right] \quad \text{(C.15)}$$

$$= A^2\left(1-\cos\left(\frac{2\pi F_{sig}}{F_s}\right)\right) E\left[\frac{|w(n)|}{T}\right], \quad \text{(C.16)}$$

in which the value of the $E\left[\frac{|w(n)|}{T}\right]$ is defined by the probability distribution and is linearly dependent on the standard deviation.

It is possible to calculate $E\left[\frac{|w(n)|}{T}\right]$ for some typical jitter signals.

For sine signal

$$E\left[\left|a\sin(2\pi nT_j)\right|\right] = \frac{2a}{\pi}. \quad \text{(C.17)}$$

For Gaussian white noise

$$E\left[\left|N\left(0,\sigma\right)\right)\right|]=\sqrt{\frac{2}{\pi}}\sigma. \tag{C.18}$$

For uniformly distributed noise $-\frac{a}{2}\geq w\left(n\right)\leq\frac{a}{2}$

$$E\left[\left|w\left(n\right)\right|\right]=\frac{a}{4}=\frac{\sqrt{3}}{2}\sigma\quad,\sigma=\frac{a}{2\sqrt{3}}. \tag{C.19}$$

The power spectral density (PSD) function of error signal due to $N\left(\sigma_w,0\right)$ Gaussian jitter can be derived as follows. The Fourier transform of the jitter signal can be written as

$$E\left(j\omega\right)=\sum_{-\infty}^{\infty}\frac{\Delta x\left(n\right)}{-j\omega}e^{-j\omega nT}\left(1-e^{j\omega\left(nT\right)}\right) \tag{C.20}$$

$$\tag{C.21}$$

and

$$\begin{aligned}\left|E\left(j\omega\right)\right|^2&=Re\left\{E\left(j\omega\right)\right\}^2+Im\left\{E\left(j\omega\right)\right\}^2 \qquad \text{(C.22)}\\&=\sum_{n=-\infty}^{\infty}\sum_{k=-\infty}^{\infty}\frac{\Delta x\left(n\right)\Delta x\left(k\right)}{\omega^2}e^{-j\omega nT}e^{j\omega kT}\left(1-e^{j\omega w\left(nT\right)}\right)\left(1-e^{-j\omega w\left(kT\right)}\right)\\&=\sum_{n=-\infty}^{\infty}\sum_{k=-\infty}^{\infty}\frac{\Delta x\left(n\right)\Delta x\left(k\right)}{\omega^2}\cos\left(\omega\left(n-k\right)T\right)\\&\quad\times\left(1-\cos\left(\omega w\left(nT\right)\right)-\cos\left(\omega w\left(kT\right)\right)+\cos\left(\omega\left(w\left(nT\right)-w\left(kT\right)\right)\right)\right)\end{aligned}$$

Assuming that there is no correlation between $\Delta x\left(n\right)$, $\cos\left(\omega\left(n-k\right)T\right)$ and $w\left(nT\right)$, the only combination of $n-k$ resulting in nonzero value is when $n=k$, and $\left|E\left(j\omega\right)\right|^2$ becomes

$$\left|E\left(j\omega\right)\right|^2=\sum_{n=-\infty}^{\infty}\frac{\Delta x\left(n\right)^2}{\omega^2}\left(2-2\cos\left(\omega w\left(nT\right)\right)\right). \tag{C.23}$$

The PSD function of the error signal is obtained by computing the expectation of $\left|E\left(j\omega\right)\right|^2$. Since the expectation of $\cos\left(\omega w\left(nT\right)\right)$ with Gaussian $N\left(\sigma_w,0\right)$ distributed $w\left(nT\right)$ is

$$E\left[\cos\left(\omega w\left(nT\right)\right)\right]=e^{\frac{-\omega^2\sigma_w^2}{2}}, \tag{C.24}$$

$$\tag{C.25}$$

and

$$E\left[\Delta x\left(n\right)^2\right]=VAR\left[\Delta x\left(n\right)\right] \tag{C.26}$$

(C.27)

when $\Delta x(n)$ has zero mean, the PSD function can be expressed as

$$S(f) = VAR[\Delta x(n)] \frac{2-2e^{-\frac{4\pi^2 f^2 \sigma_w^2}{2}}}{4*\pi^2*f^2}. \tag{C.28}$$

For a sinusoidal signal $x(n)$ at frequency F_{sig}, the PSD function is

$$S(f) = A^2 \left(1-\cos\left(2\pi\frac{F_{sig}}{F_s}\right)\right) \frac{2-2e^{-\frac{(2\pi f)^2 \sigma_w^2}{2}}}{4*\pi^2*f^2}. \tag{C.29}$$

The power of the error signal on a certain frequency band $-F_b$ to F_b can be obtained by integrating $S(f)$ resulting in

$$P_j(F_b) = A^2 \left(1-\cos\left(2\pi\frac{F_{sig}}{F_s}\right)\right) \tag{C.30}$$

$$\times \frac{e^{-2\pi^2 F_b^2 s^2} + \sqrt{\pi}\sqrt{2}\pi F_b \sigma_w \text{erf}\left(\sqrt{2}\pi F_b \sigma_w\right) - 1}{\pi^2 F_b}, \tag{C.31}$$

where

$$\text{erf}\left(\sqrt{2}\pi F_b \sigma_w\right) = \int_0^{\sqrt{2}\pi F_b \sigma_w} e^{-x^2} \text{d}x. \tag{C.32}$$

The first-order Taylor series approximation of $P_j(F_b)$ is

$$P_j(F_b) \approx 2A^2 \left(1-\cos\left(2\pi\frac{F_{sig}}{F_s}\right)\right) F_b \sigma_w^2 \tag{C.33}$$

The total error power is obtained by

$$\lim_{F_b \to \infty} P_j(F_b) = A^2 \left(1-\cos\left(2\pi\frac{F_{sig}}{F_s}\right)\right) \sqrt{\frac{2}{\pi}}\sigma_w, \tag{C.34}$$

which is equal to Eq. (C.16) with Gaussian jitter, thus following Parceval's theorem.

Appendix D

Fourier transforms of some functions used in this book

D.1 Elementary relations

Notation for transformation

$$\mathcal{F}\{x(t)\} = X(f) \tag{D.1}$$

Time shift

$$\mathcal{F}\{x(t+\tau)\} = X(f)e^{-j\omega\tau} \tag{D.2}$$

Derivative

$$\mathcal{F}\{x'(t)\} = j\omega X(f) \tag{D.3}$$

D.2 Elementary functions and operations

Sampling function

$$\mathcal{F}\left\{\sum_{n=-\infty}^{\infty}\delta(t-nT_s)\right\} = \sum_{n=-\infty}^{\infty}e^{-j\omega nT_s} \tag{D.4}$$

According to Poisson's sum formula

$$\sum_{n=-\infty}^{\infty}e^{-j\omega nT_s} \leftrightarrow F_s\sum_{n=-\infty}^{\infty}\delta(f-nF_s),\ F_s=\frac{1}{T_s} \tag{D.5}$$

Sampling

$$\mathcal{F}\left\{x(t)\sum_{n=-\infty}^{\infty}\delta(t-nT_s)\right\} = F_s\sum_{n=-\infty}^{\infty}X(f-nF_s), F_s=\frac{1}{T_s} \tag{D.6}$$

Discrete time derivative

$$\begin{aligned}&\mathcal{F}\left\{(x(t)-x(t-T_s))\sum_{n=-\infty}^{\infty}\delta(t-nT_s)\right\}\\ &=\left(1-e^{j2\pi\frac{f}{F_s}}\right)F_s\sum_{n=-\infty}^{\infty}X(f-nF_s), F_s=\frac{1}{T_s}\end{aligned} \tag{D.7}$$

Sine

$$\mathcal{F}\{A\sin(\omega_0 t)\} = \frac{A}{j2}(\delta(f-f_0)-\delta(f+f_0)), f_0=\frac{\omega_0}{2\pi} \tag{D.8}$$

Cosine

$$\mathcal{F}\{A\cos(\omega_0 t)\} = \frac{A}{2}(\delta(f-f_0)+\delta(f+f_0)), f_0=\frac{\omega_0}{2\pi} \tag{D.9}$$

Trapezoidal pulse (Figure D.1)

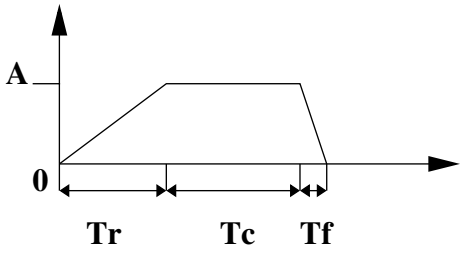

Figure D.1 Trapezoid pulse.

$$\begin{aligned}x(t) &= \frac{A}{T_r}t(u(t)-u(t-T_r))\\ &\quad +T_c(u(t-T_r)-u(t-(T_r+T_c)))\\ &\quad +\left(T_c-\frac{(t-T_r-T_c)T_c}{T_f}\right)\\ &\quad \times(u(t-(T_r+T_c))-u(t-(T_r+T_c+T_f)))\end{aligned} \tag{D.10}$$

$$\begin{aligned}\mathcal{F}\{x(t)\} &= \frac{A}{\omega^2 T_r}\left(e^{-j\omega T_r}-1\right)-\frac{A}{\omega^2 T_f}\left(e^{-j\omega T_l}-1\right)e^{-j\omega(T_r+Tc)}\\ &\quad -\frac{2jA}{\omega^2 T_r}e^{-j\omega\frac{T_r}{2}}\sin\omega\frac{T_r}{2}+\frac{2jA}{\omega^2 T_f}e^{-j\omega\left(T_r+Tc+\frac{T_f}{2}\right)}\sin\omega\frac{T_f}{2}.\end{aligned} \tag{D.11}$$

If $T_r = T_f = T_T$ we get

$$\mathcal{F}\{x(t)\} = \frac{4A}{\omega^2 T_T} e^{-j\omega\left(T_T+\frac{T_c}{2}\right)} \sin\frac{\omega T_T}{2} \sin\frac{\omega(T_T+T_c)}{2} \tag{D.12}$$

The Fourier transform of a triangular pulse with equal rise and fall times is obtained by setting $T_c = 0$ in Eq.D.12 resulting in

$$\mathcal{F}\{x(t)\} = 4AT_T e^{-j\omega T_T} \frac{\sin^2 \pi f T_T}{(\pi f T_T)^2} \tag{D.13}$$

Ramp pulse (Figure D.2)

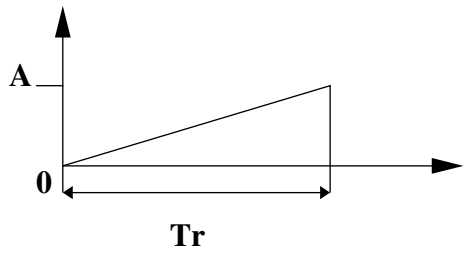

Figure D.2 Ramp pulse.

$$x(t) = \frac{A}{T_r} t\,(u(t) - u(t-T_r)) \tag{D.14}$$

$$\mathcal{F}\{x(t)\} = \frac{A}{\omega^2 T_r}\left(e^{-j\omega T_r} - 1\right) - \frac{A}{j\omega} e^{-j\omega T_r} \tag{D.15}$$

Rectangular pulse (Figure D.3)

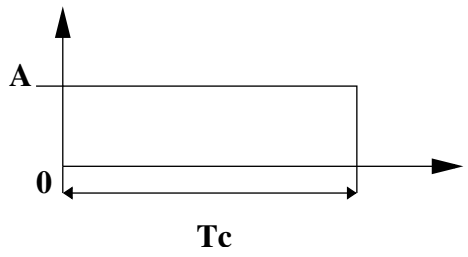

Figure D.3 Rectangular pulse.

$$x(t) = A\left(u(t) - u(t - T_c)\right) \tag{D.16}$$

$$\mathcal{F}\left\{x(t)\right\} = \frac{A}{j\omega}\left(1 - e^{-j\omega T_c}\right) \tag{D.17}$$

$$= T_c A e^{j\omega T_c} \frac{\sin \pi f T_c}{\pi f T_c} \tag{D.18}$$

Signum function of a zero-mean periodic signal $x(t)$ with 50% duty cycle (Figure D.4)

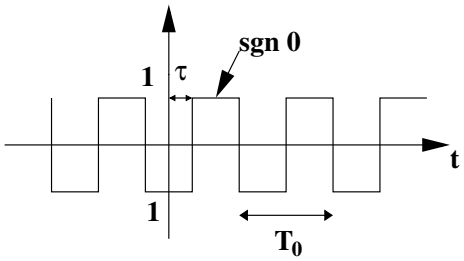

Figure D.4 Signum function of signal with 50% duty cycle and time shift τ.

$$\mathcal{F}\left\{\operatorname{sgn}(x(t))\right\} = \frac{-\operatorname{sgn}(0_+)\, e^{-j\omega\tau} f_0 \sin^2\left(\frac{\pi}{2f_0}\right)}{j\pi f} \sum_{n=-\infty}^{\infty} \delta(f - nf_0) \tag{D.19}$$

Absolute value function of a zero-mean periodic signal $x(t)$ with 50% duty cycle

$$\mathcal{F}\left\{|x(t)|\right\} = \left(\frac{-\operatorname{sgn}(0_+)\, e^{-j\omega\tau} f_0 \sin^2\left(\frac{\pi}{2f_0}\right)}{j\pi f} \sum_{n=-\infty}^{\infty} \delta(f - nf_0)\right) \otimes X(f) \tag{D.20}$$