2005

Deterministic dynamic element matching: an enabling technology for SoC built-in-self-test

Hanjun Jiang
Iowa State University

NOTE TO USERS

Deterministic dynamic element matching:

An enabling technology for SoC built-in-self-test

by

Hanjun Jiang

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Electrical Engineering

Program of Study Committee:
Degang Chen, Major Professor
Randall L. Geiger
Robert J. Weber
Chris Chong-Nuen Chu
Fritz Keinert

Iowa State University

Ames, Iowa

2005

Graduate College
Iowa State University

This is to certify that the doctoral dissertation of

Hanjun Jiang

has met the dissertation requirements of Iowa State University

Signature was redacted for privacy.

Major Professor

Signature was redacted for privacy.

For the Major Program

Table of Contents

List of Figures

List of Tables

Abstract

The analog-to-digital converter (ADC) is a key building block of today's high-volume systems-on-a-chip (SoCs). Built-in-self-test (BIST) is the most promising solution to testing deeply-embedded ADCs. Cost-effective stimulus source with on-chip integratability has been viewed as the bottleneck of ADC BIST, and consequentially the bottleneck of SoC BIST and BIST-based self-calibration. The deterministic dynamic element matching (DDEM) technique has been proposed as a solution to this problem.

In this work, rigorous theoretical analysis is presented to show the performance of a DDEM digital-to-analog converter (DAC) as an ADC linearity test stimulus source. Guided by the insight obtained this analysis, a systematic approach for cost-effective DDEM DAC design is proposed. Two generations of DDEM DACs have been designed, fabricated, and measured. 12-bit equivalent linearity was achieved from the first DDEM DAC with 8-bit apparent resolution and less than 5-bit raw linearity after systematic error compensation. The achieved 12-bit linearity outperforms any on-chip stimulus source in literature. Based on the first design, a new DDEM DAC with 12-bit apparent resolution, 10-bit raw linearity, and 9-bit DDEM switching was designed with improved design technique. This DAC was fabricated in standard 0.5-μm CMOS technology with a core die area of 2 mm^2. Clear ramp signals could be observed on an oscilloscope when the DDEM DAC was clocked at 100 MHz even though there were no instruments available that could capture the generated signal at such a high speed with sufficient resolution. Laboratory testing results confirmed that the new DDEM DAC achieved at least a 16-bit equivalent linearity; this was limited by the available instrumentation, which has 18-bit linearity. It outperforms any previously reported on-chip

stimulus source in terms of ADC BIST performance by 5 bits. The robust performance, low cost, and short design cycle for on-chip implementation make DDEM an enabling technology for SoC BIST and self-calibration.

Two new approaches based on DDEM are developed to further boost the die area efficiency, improving the basic DDEM approach. The first is termed segmented DDEM, and the second is dither-incorporated DDEM (DiDDEM). It has been shown through mathematical analysis and simulation that these can maintain the performance of the basic DDEM approach while greatly reducing the implementation cost.

Chapter 1 Introduction

1.1 ADC Test Challenge

System-on-a-chip (SoC) is one of the major trends in state-of-the-art semiconductor technology. In SoCs, multiple building components, such as microprocessors, embedded memory, radio frequency (RF) circuitry, and analog and mixed-signal (AMS) blocks, are integrated on a single chip into high-complexity, high-performance, and high-value products. Challenges are introduced by large-scale system integration on a single, tiny chip. One of the most daunting SoC challenges is development of SoC test methodology, with needs including test reusability and analog/digital built-in-self-test (BIST). [1]

Among those AMS circuits, analog-to-digital converter (ADC) is identified as the most prominent and widely used mixed-signal circuit in today's integrated mixed-signal circuit design. [1] With the development of SoCs, most of today's ADCs are deeply embedded in large system designs. In these high-volume SoC applications, ADC, together with the digital-to-analog converter (DAC), plays a significant role as the interface between the analog or physical world and the digital logic world. From the viewpoint of test, testing other AMS blocks requires ADCs that are already accurately characterized. Hence, the solution to testing embedded ADCs will also provide the key to SoC BIST.

Due to increasing resolution and conversion rates, the challenge of testing ADCs is also continuously growing. Extra difficulty is added when testing deeply-embedded ADCs in SoCs other than stand-alone ADCs. Test techniques that overcome those challenges while maintaining the test cost at an affordable level will have a significant impact in the final

product costs. [2] Specifically, for embedded ADCs, BIST is the most promising solution in terms of test performance and cost. For stand-alone ADCs, BIST is also an excellent option. More significantly, self-calibration based on BIST can lead to post-fabrication performance improvement.

The ubiquitous belief in IC test community is that to test ADCs, stimulus accuracy must be at least 10 times or 3 bits better than that of ADCs under test. However, even in production test environments, such high-accuracy stimulus source is not easy to obtain when the device under test (DUT) resolution goes high (i.e., 14 bits or more). In BIST environments, an affordable, sufficiently high-accuracy stimulus source that meets this rule is extremely difficult to implement on chip. For instance, the best published on-chip linear ramp generator [4] is only 11-bit linear, which can only be used to test 9-bit ADCs if the test error is expected to at the ¼ least-significant-bit (LSB) level. The test stimulus source accuracy has been the ADC test bottleneck, especially BIST, and, consequently, the bottleneck of SoC BIST. Test technology to solve this problem will be the enabling technology for SoC BIST.

The possible solutions to this challenge can include two major types of technologies. One is the post signal processing method that uses low-accuracy, low-linearity, and low-cost signals as the test stimulus source, and the other applies pre-processing on those low-cost stimulus sources to make their effective accuracy meet the test requirements. The technologies proposed in [5, 6] are representative of the first type in which the linearity requirements on test stimulus sources are relaxed by orders of magnitudes through the use of multiple correlated inputs and appropriate digital signal processing (DSP) techniques to accurately characterize the DUT. In this work, alternative approaches that follow the philosophy of the second type of technology are proposed and studied. The kernel technique is termed

Deterministic Dynamic Element Matching (DDEM), which was first introduced by Beatriz Olleta *et. al.* from our research group in 2003. [7] Collaboration research work since that has both theoretically and experimentally demonstrated that this DDEM method can be applied on low-resolution/low-linearity DACs to generate a set of voltage samples that can be used to test high-resolution ADCs affordably under BIST environments. [8-12] Although the BIST environment is targeted originally, the DDEM approach will also have remarkable impact on ADC production test when adequately high-performance stimulus sources are not available using traditional approaches to handle DUTs with resolution beyond 14-bits.

Before the DDEM method is explained, key issues in ADC test will be reviewed.

1.2 On-chip Stimulus Source for ADC BIST

The most important performance parameters for an ADC include static, such as integral nonlinearity (INL) and differential nonlinearity (DNL), and dynamic, such as spurious-free dynamic range (SFDR) and signal-to-noise and distortion ratio (SINAD). [3] In this work, we focus on ADC static linearity test.

An ADC's static linearity is completely characterized by the relative spacing of its transition voltages. [2, 3] For an n-bit ADC, we denoted T_k as the transition voltage from the $(k\text{-}1)^{th}$ code, C_{k-1}, to the k^{th} code, C_k. Then code width W_k for C_k can be defined as T_{k+1}-T_k. The DNL at code C_k can be defined as the difference between the actual W_k and the average code width W_0. The average code width is also known as the LSB step. The INL at C_k is defined as the difference between the actual transition voltage, T_k, and the ideal position for T_k. Equations (1.1) and (1.2) give the formulas to calculate the DNL[k] and INL[k] from the ADC transition voltage sequence. The INL[k] curve can be obtained by integrating the DNL[k]

curve, and the DNL[k] curve can be calculated by differentiating the INL[k] curve. The overall INL and DNL are the maximum absolute values of INL[k] and DNL[k], respectively.

$$DNL[k] = \frac{T_{k+1} - T_k}{(T_{N-1} - T_1)/(C_{N-1} - C_1)} - 1 \tag{1.1}$$

$$INL[k] = C_k - \left(\frac{T_k - T_1}{T_{N-1} - T_1}(C_{N-1} - C_1) + C_1 \right) \tag{1.2}$$

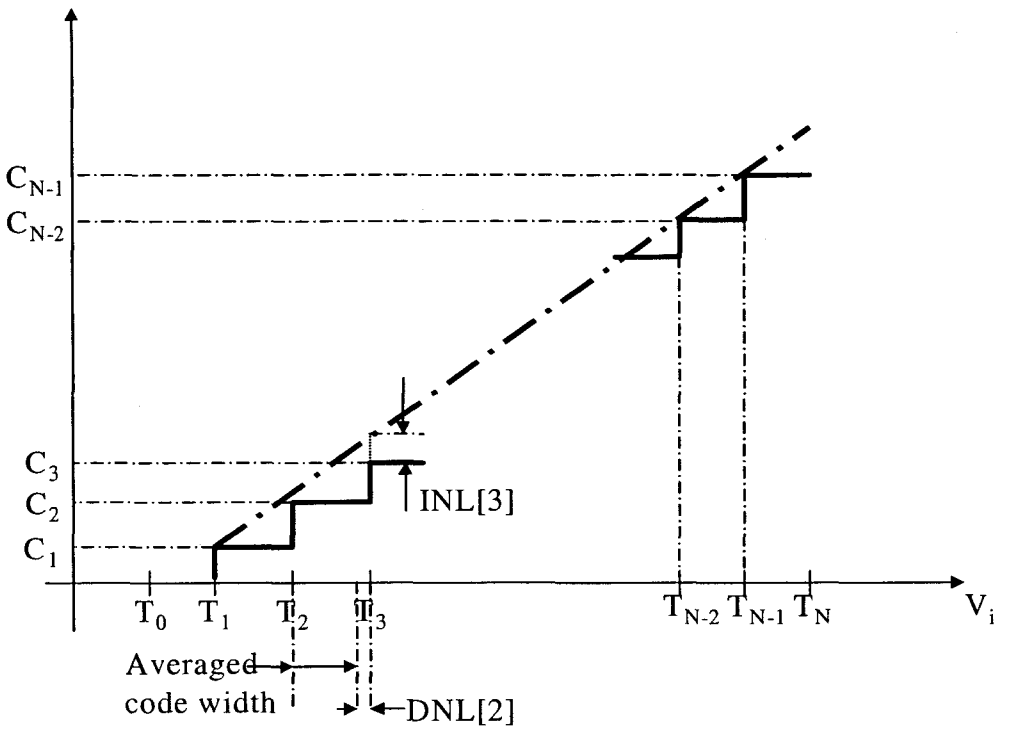


Figure 1.1 ADC fit line and DNL[k] and INL[k]

Since the INL[k] and DNL[k] curves can be easily calculated from each other, we will focus on the INL[k] curve characterization only.

The histogram-based test is widely used to test the ADC INL[k] curve. In [3], the linear ramp histogram and sinusoidal wave histogram tests are recommended. From these two types of histogram-based tests, the linear-ramp-based test is more efficient to implement and more

commonly adopted. The main challenge in a linear-ramp-based ADC test is generating high-accuracy linear ramp signal itself. In the BIST environment, this problem becomes more stringent due to the limited availability of high-performance stimulus sources. On the other hand, the sine-wave histogram test is not suitable for the BIST environment, because to provide high-purity sine-waves requires large die area overhead, [4] which conflicts with the die area constraints on SoCs. In this work, we seek stimulus sources for the linear ramp histogram test only.

The ways to generate linear ramps can traditionally be divided into two types. One uses high-performance DACs as the signal generators and the other, continuous-time analog linear ramp generators. Since the best available DACs usually have higher resolutions than contemporary ADCs, it will be adequate in terms of test performance to use high-performance DACs as ADC test stimulus sources. For example, the 16-bit, 400M sample-per-second DAC with 15-bit effective number of bits (ENOB), reported by W. Schofield *et. al.* in 2003 [13], will be adequate to test 12-bit ADCs, with the test error bounded by ¼ LSB. However, in the BIST environments, such high-performance DACs cannot be adopted due to high costs resulting from the large die area and long design cycle. As a contrast, it is more practical to use continuous-time analog ramp generators in BIST environments because of merits such as simple circuit structure and small circuit size.

Many research groups have investigated the ramp generator issue, and many linear ramp generator structures have been proposed and published. However, the structures published since the 1990s either lacked the experimental demonstration or the results did not show adequate accuracy to test high-resolution ADCs. [4, 14-17] As stated previously, the best linear ramp generator, reported was by B. Provost *et. al.* [4] and it is only 11-bit linear and

can only be used to test 9-bit ADCs with test errors bounded by ¼ LSB. To implement a time-continuous linear ramp generator on-chip with constrained die size has been demonstrated to be a difficult task indeed, and so far, there is not any effective way to build a linear ramp generator in the BIST environments for ≥12-bit ADCs test.

It is necessary to find out what is really required to perform the ramp histogram test. By examining carefully the ramp histogram test, it can be shown that what is really needed is not the actual linearity of the ramp signal itself but rather the uniform spacing of the collective set of voltage samples presented to the ADC under test. If multiple ramp-up and ramp-downs are to be used in testing an ADC, the linearity of individual ramps with respect to time in the test signals is actually unnecessary for this test. To achieve uniform spacing in the aggregate set of voltage samples without caring for the time-local signal linearity is drastically easier than achieving a high-accuracy linear ramp. This insight is the key to the DDEM approach proposed in this work, and the uniform output spacing will be the target for our stimulus sources presented in the following chapters.

It will be helpful to find the criterion for the stimulus sources designated for the ramp histogram ADC test. It is clear that with stimulus voltage samples that have ideal uniform spacing, the histogram bin height $H[k]$ for ADC output code k will be proportional to W_k. Under the ideal case, we can use $H[k]$ to characterize W_k. Assume the stimulus to the DUT (ADC) has a voltage range $[V_{smin}, V_{smax}]$ and the ADC input voltage range is $[V_{min}, V_{max}]$. The stimulus voltage range should cover the DAC input range. For an arbitrary voltage, V_t, let $h(V_t)$ represent the number of stimulus voltage samples that falls into $[V_{smin}, V_t]$. It is clear that $H[k]$ is determined by $h(T_k)$ and $h(T_{k+1})$. The linearity of $h(T_k)$ with respect to T_k will determine how accurately we can characterize W_k. Based on this analysis, we can evaluate

the ADC testing accuracy with a given stimulus by estimating the linearity of $h(V_t)$ with respect to V_t. We define the error expression $e(V_t)$ as

$$e(V_t) = h(V_t) - h(V_{\min}) - C_h \cdot (V_t - V_{\min}) \tag{1.3}$$

in which C_h is the ideal value of $(h(V_t) - h(V_{\min}))/(V_t - V_{\min})$. Equation (1.3) will be used as a criterion to evaluate the approaches proposed in this work.

In the following chapters, we will present the on-chip stimulus source to test high-resolution ADCs using the DDEM technique. In Chapter 2, the DDEM approach is described, followed by some preliminary simulation results. Chapter 3 gives some analysis to validate the DDEM approach by evaluating the "averaged" DAC. Optimization guided by this analysis will also be given in this chapter. A rigorous mathematical analysis on DDEM DAC is given in Chapter 4 by directly inspecting its output distribution, and a systematic approach for cost-effective DDEM DAC design is subsequently presented based on this analysis. Two DDEM DAC designs, together with experimental results, are shown in Chapter 5. In Chapters 6 and 7, two new techniques originated from DDEM will be proposed, respectively, with further improved performance/cost ratio for the high-resolution ADC test.

Chapter 2 Deterministic Dynamic Element Matching

In this chapter, the Deterministic Dynamic Element Matching (DDEM) technique will be reviewed, and some simulation results will be given. Before doing so, it is necessary to look at the general Dynamic Element Matching (DEM) technique first.

2.1 Dynamic Element Matching

DACs are commonly used to generate stimulus signals for an AMS circuit test. The performance of most useful DAC architectures is dependant on matching properties of critical elements. For the current steering DAC structure, depicted in Figure 2.1, the output linearity is greatly dependent on the matching performance of current source elements. Due to process variation and other reasons, element matching errors are inevitable. Although special layout techniques, special processes, and/or laser trimming can be used to reduce matching errors, these methods lead to significant cost increases and are difficult to use for ADC BIST.

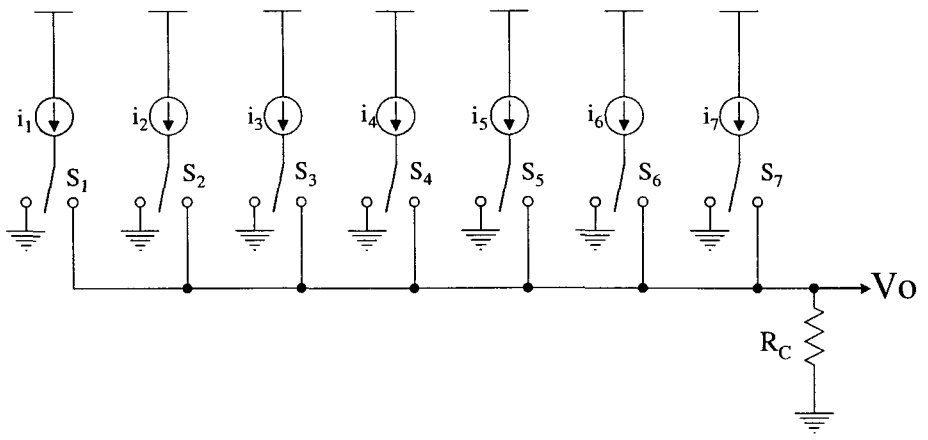
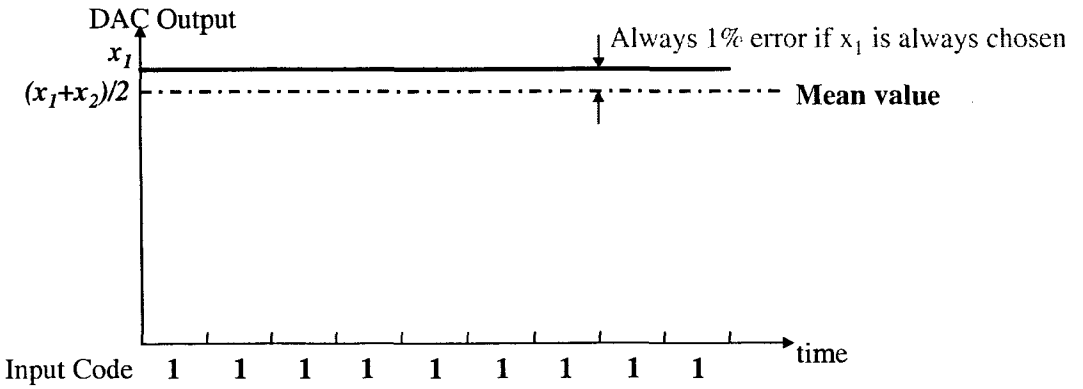
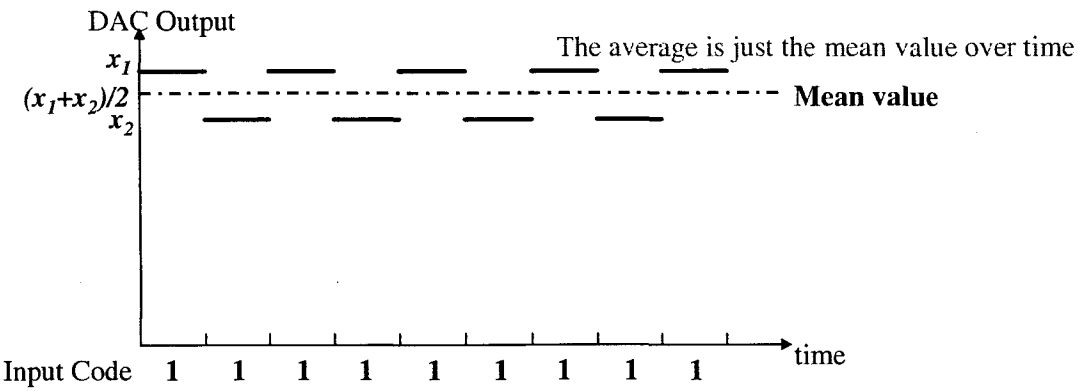


Figure 2.1 A 3-bit current mode thermometer-coded DAC



(a) Output for code 1 without DEM



(b) Output for code 1 with DEM

Figure 2.2 Averaging effect of DEM

The Dynamic Element Matching (DEM) technique [18, 19] accepts matching errors as inevitable and dynamically rearranges the interconnections of the mismatched elements so that on the average all element values are nearly equal. The averaging effect of DEM is shown in Figure 2.2, with a two-element DAC as the example. Assume the DAC has two elements, x_1 and x_2. In the ideal DAC case, all the DAC elements are equal to their mean value. When element mismatching exists, the DAC elements deviate from their mean value. In this case, assume that x_1 is 1% more than the mean value $(x_1+x_2)/2$ and that x_2 is 1% less

than the mean value. Without the DEM technique, when the input code is 1, x_1 is always chosen and the output is always 1% larger than the ideal value $(x_1+x_2)/2$. However, with the DDEM technique, x_1 and x_2 are chosen alternatively when the input code is 1 and the averaged output value over time for input code 1 is exactly the ideal value $(x_1+x_2)/2$. Although various switching sequences for different DEM approaches exist, all of them take advantage of the averaging effect to enhance the DAC overall output linearity.

The DEM method was first introduced by Van De Plassche in 1976 [18], and was used by H. T. Jensen and I. Galton [20, 21] to improve the effective specifications of DAC linearity performance. It has been demonstrated that DEM can be used to appreciably improve the SFDR performance of moderately low-linearity DACs [20] by spreading the errors in the DAC over a wide spatial frequency spectrum. This behavior is a direct consequence of the averaging effect over time provided by DEM.

Other researchers [22-27] have used DEM in Delta-Sigma Converters, and the high oversampling ratio inherent in these structures can either partially or totally remove the limitations associated with the time-local errors. B. H. Leung and S. Sutarja presented three different approaches to DEM [22]: conventional random averaging, clocked averaging, and individual level averaging. The DEM was applied on a 3-bit DAC in a Delta-Sigma ADC. R. T. Baird and T. S. Fiez [23] introduced and analyzed the Data Weighted Algorithm (DWA). Different modifications to the DWA are made [24-27] to improve its performance. Adams and his colleagues present a Data-Directed Scrambler for multi-bit noise shaping D/A converters. [27] An explanation of the DEM techniques is shown in Figure 2.3, with the DWA approach as an example.

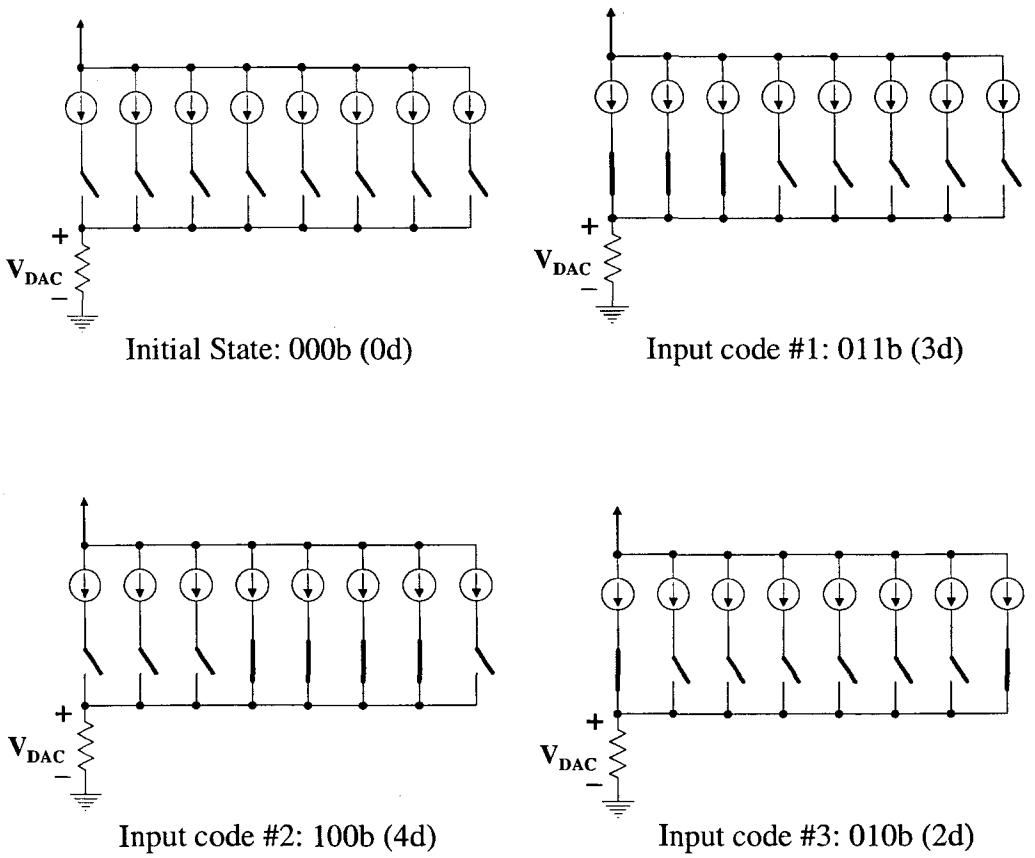


Figure 2.3 DWA DEM switching sequence

The element switching/selecting sequence is the most significant part of different DEM algorithms. In the previous DEM algorithms shown in the literature, the element switching sequences are either random or nearly random. Most are input data dependent and are variants of the Data-Dependant First-use-Next-use (DDFN) algorithm. Many problems exists in these algorithms, such as the spectral spurious components generated by DEM, the large sampling window required, and the time-local non-stationarity.

All of the above DEM algorithms are used in real-time signal paths, and none have been used for test purposes. When the DEM DACs are used as ADC static test stimulus sources, as shown in Figure 2.4, problems related to spectral performance or real-time performance no

longer matter. Instead, the output distribution is of the most significant interest, as stated in Chapter 1. It is clear that some problems should be resolved when the DEM DACs are used for the test purpose. First, the performance needs to be re-evaluated for testing. The averaging performance over time is no longer of any meaning in the ADC test; instead, the output distribution needs to be inspected. The second consideration is the implementation issue. It has been found to be too complicated to implement any random DEM control logic, even on a modest high-resolution DAC, as a large scrambler is required. [28] To apply the DEM DAC for the ADC BIST purpose, new DEM control logic that requires much less hardware to implement must be investigated.

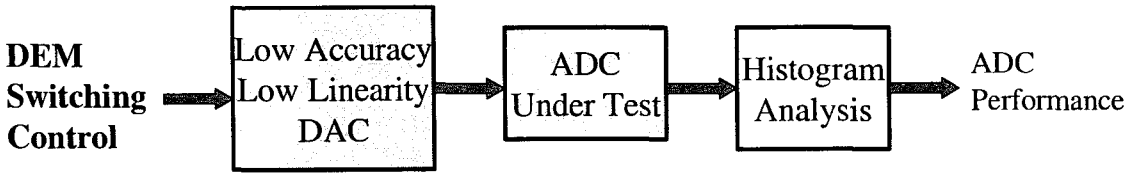


Figure 2.4 DEM DAC for ADC test

The DDEM algorithm was proposed as the solution. This algorithm overcomes the previously mentioned problems associated with random DEM algorithms while keeping the implementation cost adequately low for BIST.

2.2 DDEM Description

In this section, the Cyclic DDEM Switching Sequence for a current steering thermometer-coded DAC is reviewed. It will be shown that the Cyclic DDEM Switching Sequence is very easy to implement and gives an excellent performance.

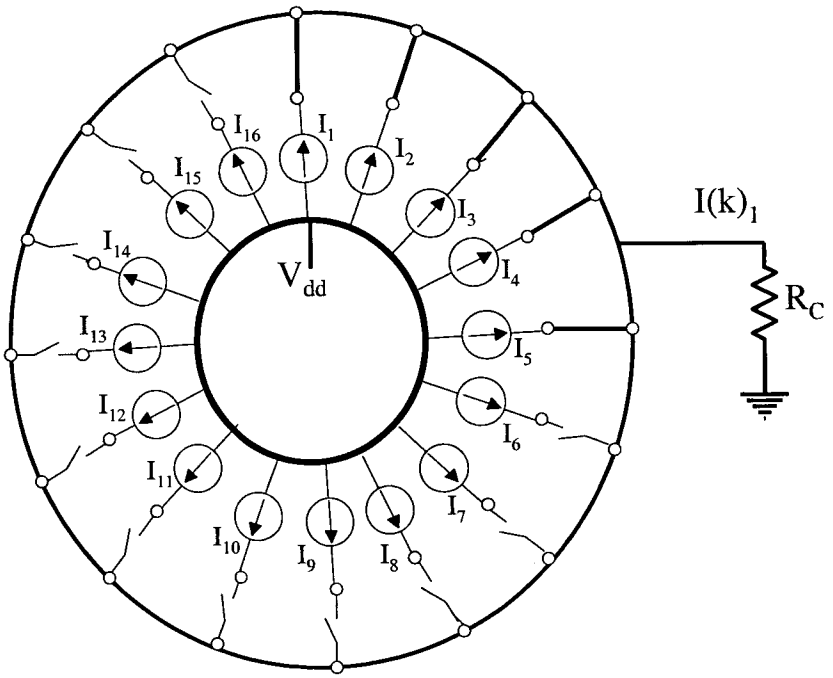
The DDEM switching sequence will be applied to a current steering thermometer-coded DAC, as shown in Figure 2.1, in which a 3-bit current DAC is depicted. In this DAC

structure, for a digital input word "k", k switches need to be connected to the output node. The resistor R_C is chosen so that when all of the current sources are on, the voltage output is at the desired maximum. The DDEM approach controls the DAC switching with a certain pattern to generate stimulus voltage samples for the ADC test. It is evident that the switching sequence will determine the output distribution and, therefore, the test performance.

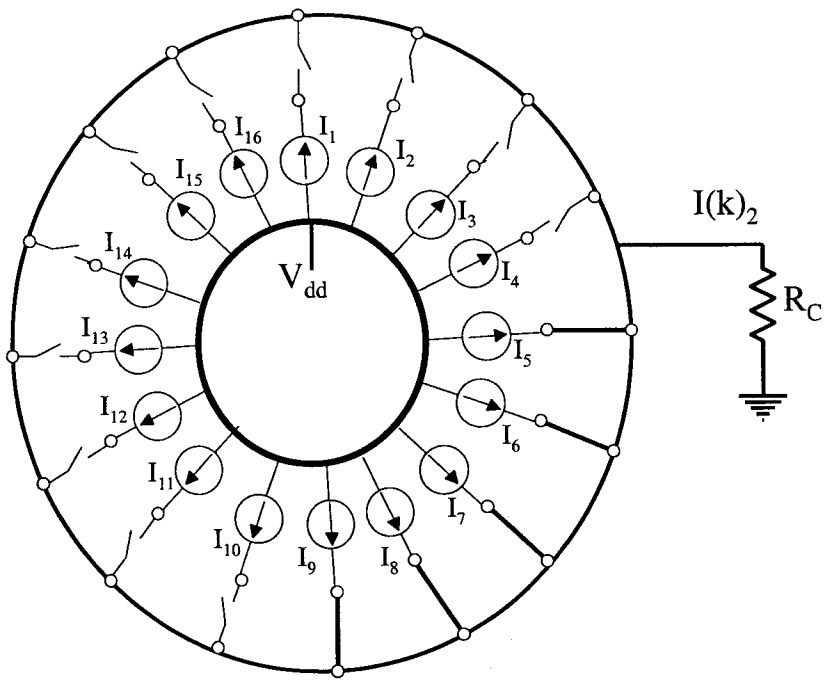
A normal n-bit thermometer-coded DAC has 2^n-1 current source elements, as shown in Figure 2.1. In the DDEM DAC, one extra current source element has been added to the DAC in Figure 2.1 so that an n-bit DDEM DAC has a total of $N = 2^n$ current sources. We use $i_j \quad (j = 1,...,N)$ to represent the j^{th} current source element out of the total N elements.

The conventional random DEM idea for generating an output voltage for a digital word "k" is to pick k switches randomly to be turned on each time an output corresponding to word k is desired. The DDEM method deterministically picks the k current sources to be switched on. Multiple outputs are generated for each digital word "k" with different deterministically selected current sources. The number of outputs per DAC input code is denoted as p. p is also termed as the DDEM iteration number. An integer q is defined by the expression $q = N/p$. The iteration number p should be selected such that q is an integer.

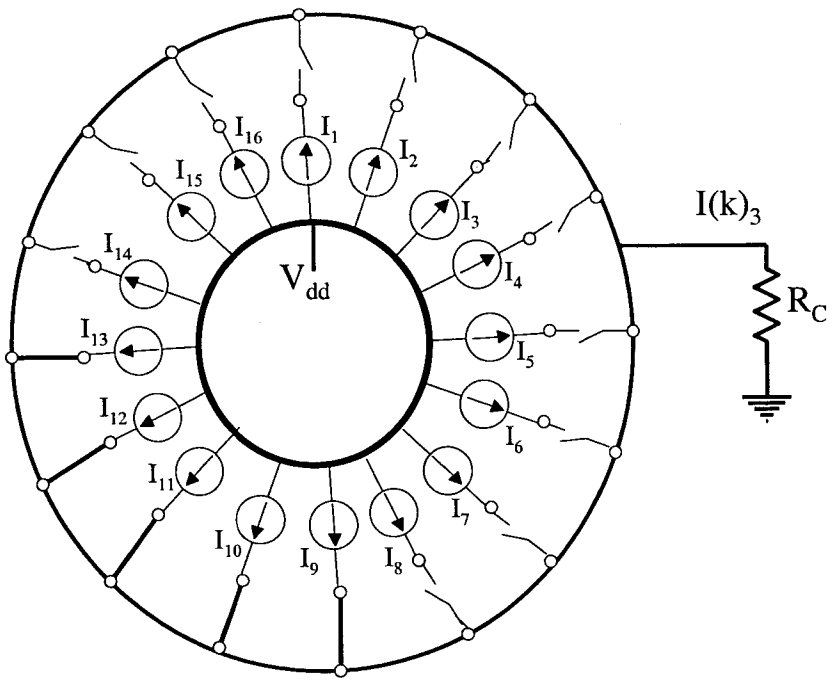
To show the switching sequence, the current sources are arranged conceptually and sequentially around a circle, as seen in Figure 2.5, to visualize a wrapping effect whereby the N^{th} current source is adjacent to the first current source. The physical layout of the current sources does not need to have any geometric association with this cyclic visualization.



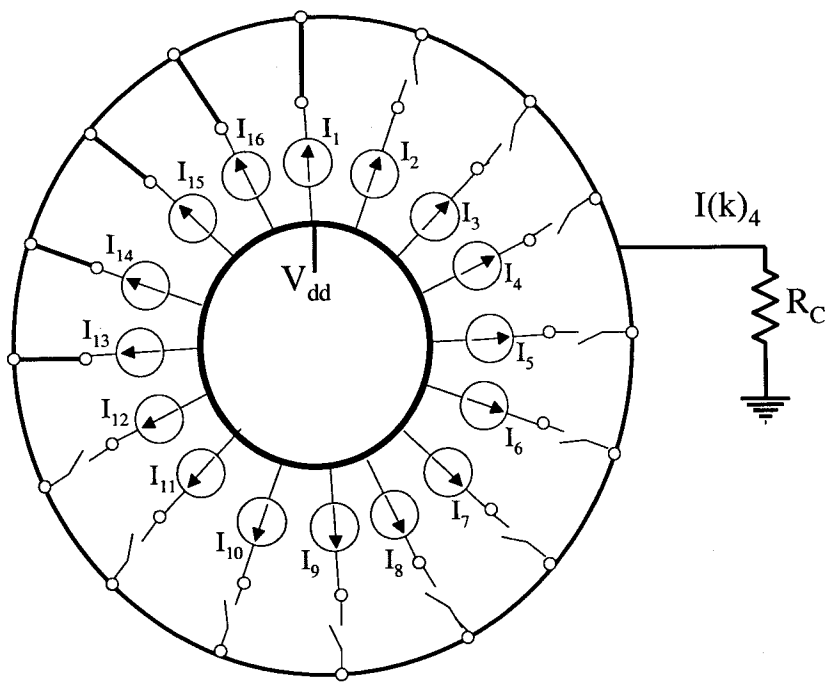
a) 1st output sample when *k*=5



b) 2nd output sample when *k*=5



c) 3rd output sample when k=5



d) 4th output sample when k=5

Figure 2.5 Cyclic DDEM switching of a 4-bit DAC

All the N current sources are divided into p groups, with each group starting from an index source from the sequence given by i_1, i_{1+q}, i_{1+2q},..$i_{1+(p-1)q}$. These index current sources are uniformly spaced around the circle. For each input code $k, 1 \leq k \leq p$, the DAC generates p output voltages. Each output voltage is obtained by switching on k current sources consecutively starting with one of the p index current sources. Thus, the d^{th} voltage sample ($1 \leq d \leq p$) is obtained by switching on k current sources consecutively starting with $i_{1+(d-1)q}$ and continuing around the circle in the clock-wise direction. We term this the Cyclic DDEM Switching Sequence. The Cyclic DDEM Switching Sequence is not data-dependent and is completely deterministic. It can be shown that the logic needed to implement the Cyclic DDEM approach is quite simple, and a shift register can be used to drive the switches that select the current sources. An example of this Cyclic Switching Sequence is shown in Figure 2.5, in which $k = 5$, $n = 4$, $N = 16$, $p = 4$, and $q = 4$.

The DDEM approach takes advantage of the fact that for the INL test, the ADC needs to be tested from the static viewpoint, where the output of a DAC is used as the input to the ADC. The DAC's output for the same input digital word will be sent to the ADC p times using different deterministically chosen current sources. The ADC's outputs corresponding to all the p input samples are then stored for calculating the ADC INL later. In this way, the real-time limitations are eliminated, and a stimulus signal containing a set of uniformly spaced voltage samples can be generated, as shown, and is verified in the following chapters. Some preliminary simulation results are given in the next section to show the performance of DDEM DAC as the ADC static test stimulus source.

2.3 Simulation Results

Numeric simulations were first carried out to verify the DDEM DAC performance as the ADC static linearity test stimulus source. In the simulation, current steering DACs with randomly generated current source values were adopted. These DAC current source elements have totally random mismatching errors only. They were controlled by the DDEM Cyclic Switching Sequence, and the generated voltages samples were sent to the ADCs under test. The INL of these simulated ADCs was then estimated with a standard histogram method. The DDEM DAC performance was evaluated by inspecting how accurately the ADCs INL were estimated.

It will first be shown that ADC test capability of the DDEM DAC can far exceed the DAC resolution. In the first simulation example, we show how an 8-bit resolution DAC with 3 LSB INL was used to test 12-bit and 11-bit ADCs, with DDEM iteration number p equal to 256 (named the full-p test, since p is equal to the DAC current element number). The DAC has current source mismatching modeled by a Gaussian distribution, with σ = 0.3, and is truncated at the 50% variation level. In the simulation, the DDEM approach is applied to the DAC, and the DAC's output serves as the test stimulus to simulated ADCs. We can calculate the true INL[k] of the simulated ADCs and compare the true INL[k] curve to the estimated INL[k] curve by using DDEM DAC as the test stimulus source.

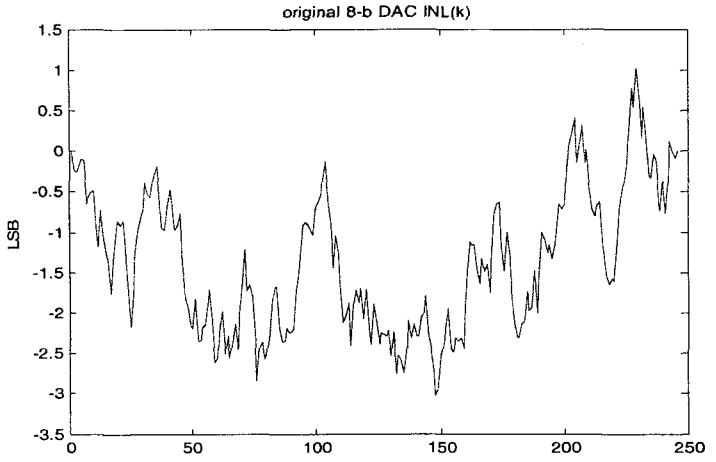
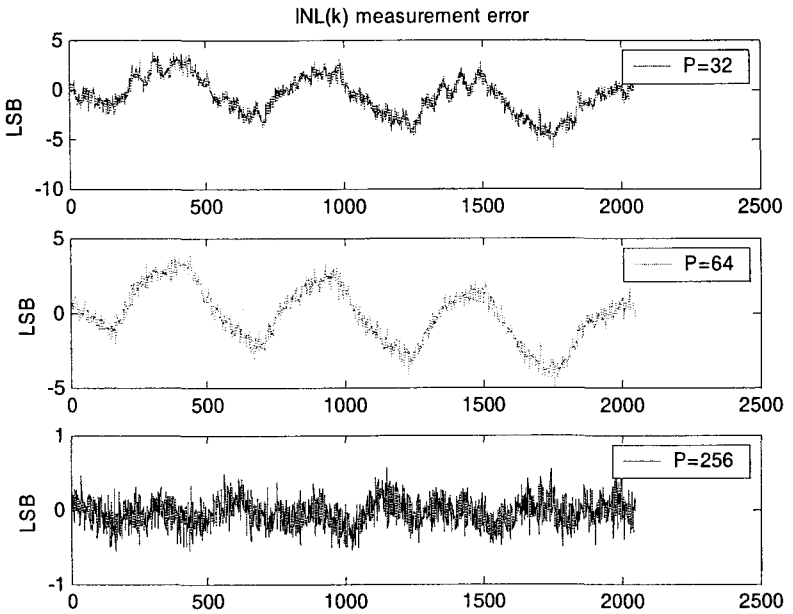
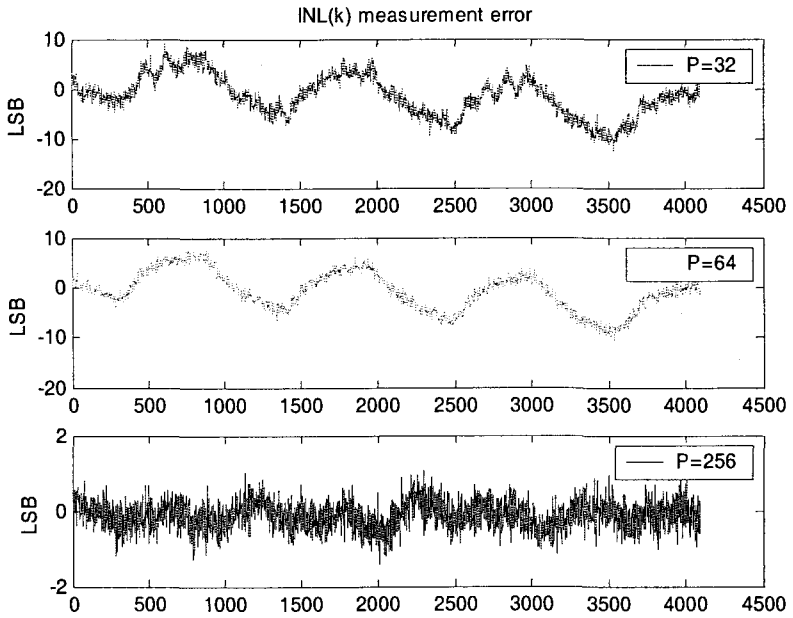


Figure 2.6 INL[k] of the original 8-bit DAC (simulated)

The INL[k] plot of the original 8-bit DAC without DDEM is shown in Figure 2.6. In Figure 2.7 (a) and (b), the ADC INL[k] testing errors using the DDEM DAC with different p for 11-bit and 12-bit ADCs are depicted, respectively. For the 11-bit ADC, the maximum INL[k] error was 0.65 LSB, with p=256 (32 samples per ADC code), while for the 12-bit ADC, the maximum INL[k] error was 1.52 LSB, with p=256 (16 samples per ADC code). It should be pointed out that if a simulated, ideal 8-bit DAC is used to test the 12-bit ADCs, the average maximum INL[k] error is up to 16.5 LSB.



(a) INL[k] test error for the 11-bit ADC



(b) INL[k] test error for the 12-bit ADC

Figure 2.7 ADC INL[k] test error with varying p (simulated)

The following results show how a 12-bit DDEM DAC was simulated to test 14-bit ADCs. The current element-mismatching error amount is chosen so that the original DAC has INL of approximately 2 LSB, which means that the DAC ENOB is about 10 bits. Figure 2.8 shows the INL[k] plot of the original 12-bit DAC, which is 2.1 LSB. Figure 2.9 shows the comparison between the true ADC INL[k] curve and the estimated INL[k] curve by using this DDEM DAC, with p=512. The maximum estimation error in ADC INL[k] is bounded by ±0.15 ADC LSB.

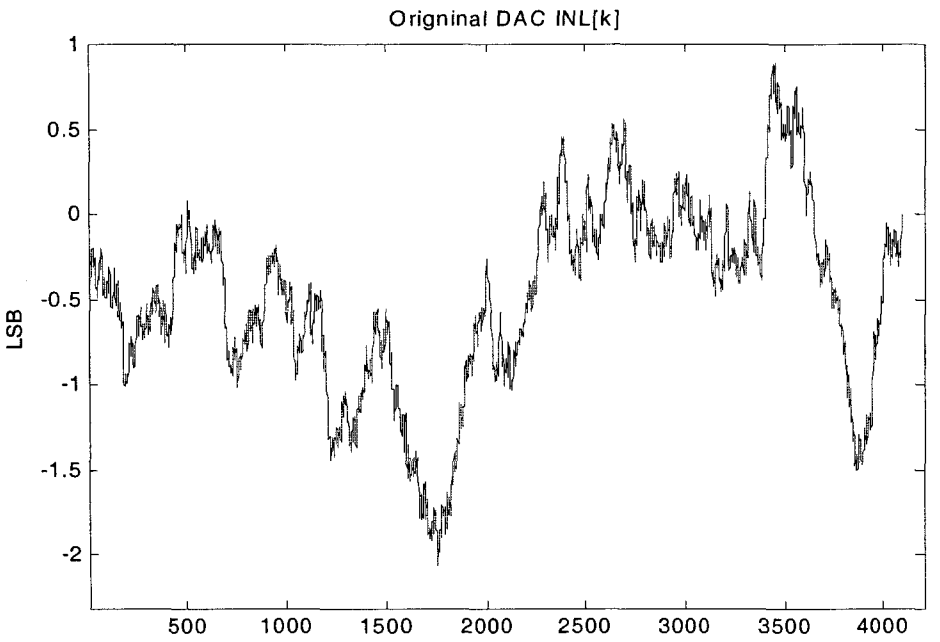


Figure 2.8 INL[k] of the original 12-bit DAC (simulated)

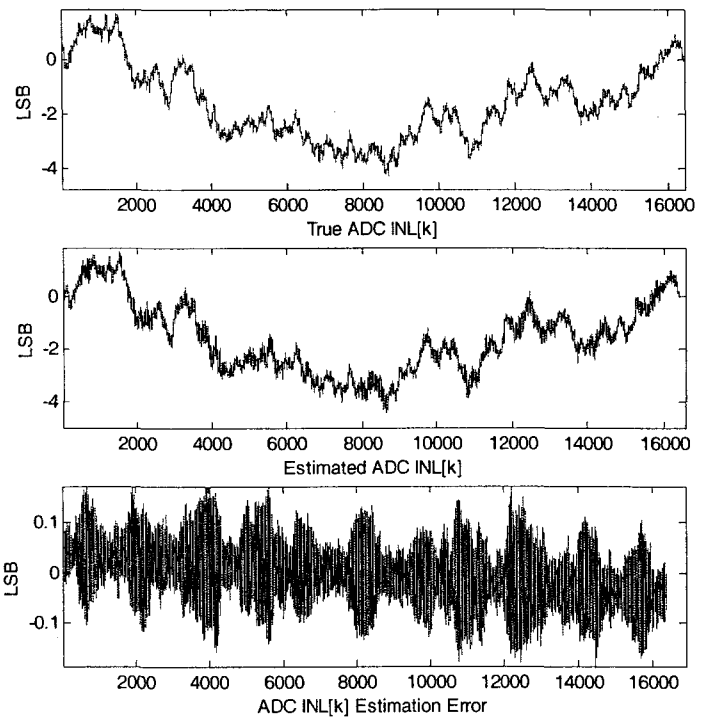


Figure 2.9 ADC INL[k] test error using a 12-bit DDEM DAC (simulated)

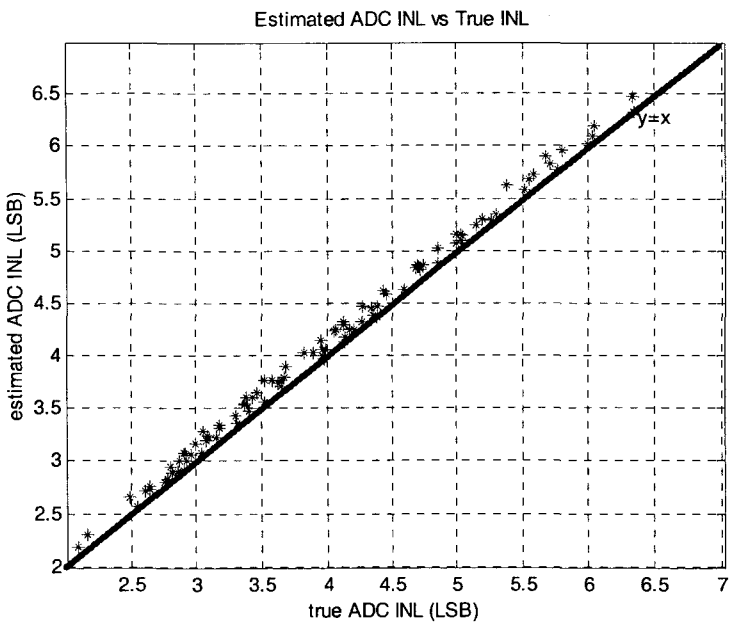


Figure 2.10 ADC true INL vs estimated INL

To verify the robustness of the DDEM DAC performance in the case that a large amount of ADCs are under test, the simulated 12-bit DDEM DAC was used to test 100 14-bit ADCs. Figure 2.10 depicts the estimated ADC INL versus the true ADC INL. The ADC INL estimation error (estimated INL – true INL) varies from -0.0141 to 0.25 LSB. It is clear that the ADC INL is almost never underestimated, which helps avoid sending bad parts to the customer; this is critical for the actual test.

In the following simulation, it will be shown how the DDEM algorithm outperforms the random DEM (RDEM). In Figure 2.11, we compare the performance of estimating the INL for 100 11-bit ADCs using a 10-bit DAC with RDEM and DDEM, respectively. In the comparison, the same 10-bit DAC with an INL of about 10 LSB was used, and p was chosen to be 128. From Figure 2.11, one important observation can be made that the DDEM method offers substantial improvements in testing performance over the RDEM approach for a given DEM iteration number. In the following chapters, it will also be shown that DDEM requires much less hardware for implementation than RDEM.

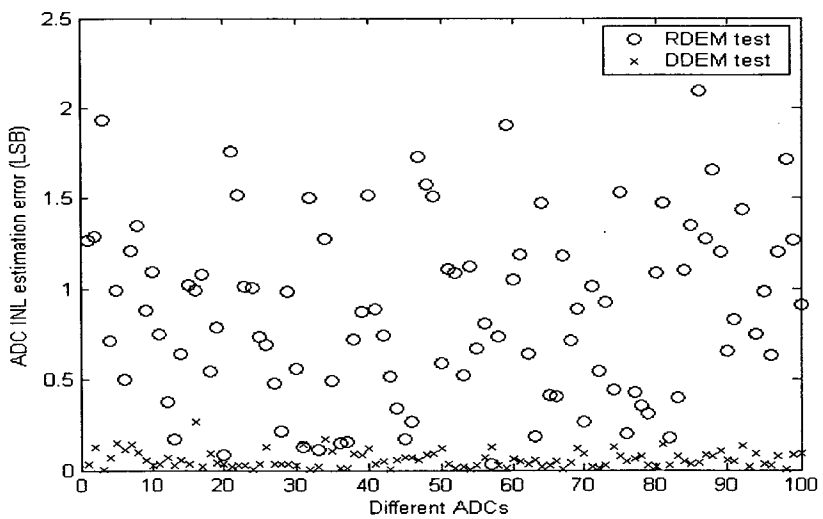


Figure 2.11 Comparison of RDEM and DDEM for estimating 100 ADCs' INL

Chapter 3 Averaged DAC

The performance of the DDEM DAC as the ADC static test stimulus source has been shown in Chapter 2 by simulation results. In this chapter, some preliminary analysis is presented to validate the DDEM performance analytically. Guided by this analysis, DDEM performance optimization will be given with the aid of numerical simulations.

3.1 Averaged DAC

The averaging effect is the most important feature of the existing DEM algorithms. It is natural to start the DDEM DAC performance evaluation from the averaging effect. However, the averaging effect over time is no longer important for test purposes. Instead, we will inspect the averaged DDEM DAC output for each DAC code by defining the "averaged" DAC and evaluating the linearity of the "averaged" DAC.

Before doing so, it is necessary to define the DDEM DAC outputs. For the structure shown in Figure 2.5, an n-bit DDEM DAC has $N = 2^n$ current source elements, denoted as $i_j \quad (j = 1,...,N)$. Suppose that the expected value of all current elements in this DAC is i_0, which is unknown and may be different from wafer to wafer or from die to die. Due to random process variations, the actual value of each current source is given by:

$$i_j = i_0(1+\varepsilon_j) \quad (j = 1,...,N) \tag{3.1}$$

We assume that the variations follow a Gaussian distribution and that ε_j $i.i.d.$ $\sim N(0,\sigma^2)$ in which σ^2 is determined by the allocated area to each current element, layout strategies, and

process variations. This Gaussian assumption is reasonable and is frequently adopted in literature.

For each input code k, the DAC outputs p samples. p is termed the DDEM iteration number. Each output is the summation of the selected k current elements scaled by R_C, in which R_C is the output resistance. The d^{th} current summation is given by:

$$I_d[k] = \sum_{j=1}^{k} i_{(d-1)q+j} \qquad d = 1,\ldots, p \tag{3.2}$$

The average of the p samples for code k is given by:

$$\bar{I}[k] = \frac{1}{p}\sum_{d=1}^{p} i_d[k] = \frac{1}{p}\sum_{d=1}^{p}\sum_{j=1}^{k} i_{(d-1)q+j} \tag{3.3}$$

The "averaged" DAC is defined to be an n-bit DAC with the output sequence given by $\bar{I}[k] \quad (k = 0,1,\ldots,N)$, with the weighting fact R_C. By definition, $I_d[0]$ and $I_d[N]$are always fixed, and we denote the fixed values by $I[0]$ and $I[N]$ for codes 0 and N, respectively. In the following analysis, the scaling factor R_C is disregarded since it plays no role in the linearity analysis, and only the output current is inspected.

It is obvious that the overall output range is given by $I[N] = N \cdot i_0 + i_0 \sum_{j=1}^{N} \varepsilon_j$. Due to uncertainties in $I[N]$, the actual output range may not reach the nominal level. To make sure that DAC output range covers the DUT input range, it is necessary to expand the DDEM DAC nominal output range a little bit beyond the DUT nominal input range as what we have already done in the simulations shown in Chapter 2.

We will first look at the linearity of this "averaged" DAC, and then inspect the actual output of the DDEM DAC. To evaluate the linearity of the "averaged" DAC, we define an end-point

fit line that connects (0, $I[0]$) and (N, $I[N]$). Therefore, the LSB of the "averaged" DAC is given by:

$$\overline{LSB} = \frac{I[N]-I[0]}{N} = i_0 + i_0 \frac{1}{N}\sum_{j=1}^{N}\varepsilon_j \tag{3.4}$$

It can be verified that $\frac{1}{N}\sum_{j=1}^{N}\varepsilon_j$ is Gaussian and has the following standard deviation:

$$\frac{1}{N}\sum_{j=1}^{N}\varepsilon_j \sim N(0,\frac{1}{N}\sigma^2) \tag{3.5}$$

Since N is a very large number, the variation of $\overline{LSB}$ from i_0 is adequately small.

The fit-line of the DDEM DAC for each input code k is then given by:

$$\bar{I}_{fit}[k] = k \cdot \overline{LSB} = ki_0 + i_0 \frac{k}{N}\sum_{j=1}^{N}\varepsilon_j \tag{3.6}$$

Once we have the fit line, we can now compute the INL of the "averaged" DAC. This is done by calculating the distance from the "averaged" DAC output current value to the corresponding fit line point for each DAC input code. Let's denote this distance at code k by $\overline{INL}[k]$. For any code k, write it as

$$k = t \cdot q + s \,(\mathrm{s} = 0,1,...,\mathrm{q}-1;\ \mathrm{t} = 0,1,...,\mathrm{p}) \tag{3.7}$$

The "averaged" DAC output current derived from (3.3) and (3.7) is given by:

$$\bar{I}[k] = k \cdot i_0 + i_0 \cdot \frac{1}{p} \cdot \left(t \cdot \sum_{d=1}^{p}\sum_{j=s+1}^{q}\varepsilon_{(d-1)q+j} + (t+1) \cdot \sum_{d=1}^{p}\sum_{j=1}^{s}\varepsilon_{(d-1)q+j} \right)$$

$$= k \cdot i_0 + i_0 \cdot \frac{1}{p} \cdot \left(t \cdot \sum_{j=1}^{N}\varepsilon_j + \sum_{d=1}^{p}\sum_{j=1}^{s}\varepsilon_{(d-1)q+j} \right) \tag{3.8}$$

Subtracting it by the corresponding fit line point yields:

$$\begin{aligned}\overline{INL}[k] &= \bar{I}[k] - \bar{I}_{fit}[k] \\ &= i_0 \frac{1}{p} \cdot \left(t \cdot \sum_{j=1}^{N} \varepsilon_j + \cdot \sum_{d=1}^{p} \sum_{j=1}^{s} \varepsilon_{(d-1)q+j} \right) - i_0 \frac{k}{N} \sum_{j=1}^{N} \varepsilon_j \\ &= i_0 \left[\left(\frac{1}{p} - \frac{s}{N} \right) \sum_{d=1}^{p} \sum_{j=1}^{s} \varepsilon_{(d-1)q+j} - \frac{s}{N} \sum_{d=1}^{p} \sum_{j=s+1}^{q} \varepsilon_{(d-1)q+j} \right] \end{aligned} \tag{3.9}$$

It can be proven that the distribution of the normalized $\overline{INL}[k]$ is given by:

$$\frac{\overline{INL}[k]}{i_0} \sim N\left(0, A\sigma^2\right) \;\&\; A = \left(\frac{1}{p} - \frac{s}{N} \right)^2 ps + \left(\frac{s}{N} \right)^2 p(q-s) = \frac{s(q-s)}{pq} \tag{3.10}$$

Equation (3.9) shows that whenever s= 0, that is, k is a multiple of q, $\overline{INL}[k]$ is equal to zero, meaning that the DDEM DAC output is exactly on the fit line. This is correct because when $k = tq\,(t = 1,\ldots, p)$, each and every current source in the DAC will be used exactly t times to create the p output samples. Hence, their averaged value will be exactly $k \cdot \overline{LSB}$, with no uncertainty.

From (3.10), we can estimate the location and amount of the maximum deviation from the fit line. The variance of $\overline{INL}[k]$ reaches its maximum value at $s = q/2$. Using this value for s and with i_0 equal to 1 LSB, the largest standard deviation of $\overline{INL}[k]$ is:

$$\sqrt{\frac{q}{4p}}\sigma = \sqrt{\frac{N}{4p^2}}\sigma \tag{3.11}$$

It is known [29] that the distribution of the $INL[k]$ for a non-DEM current steering DAC is given by:

$$\frac{INL[k]}{i_0} \sim N(0, \frac{(N-k)k}{N}\sigma^2) \tag{3.12}$$

The largest standard deviation of $INL[k]$ is approximately $\frac{\sqrt{N}}{2}\sigma$. A numeric example is given to compare this maximum $INL[k]$ deviation to the maximum $\overline{INL}[k]$ deviation given in (3.12). Assume n=18, p=2^7=128, and q=2^{11}. The averaged DDEM DAC's maximum $\overline{INL}[k]$ deviation is $\sqrt{\frac{q}{4p}}\sigma = 2\sigma$. Compare this against the standard DAC's maximum $INL[k]$ deviation of $\frac{\sqrt{N}}{2}\sigma = 2^8\sigma$. We can see that the averaged DDEM DAC's maximum $\overline{INL}[k]$ deviation is reduced by a factor of 128 = p.

The conclusion is that the $\overline{INL}$ of the "averaged" DAC is greatly improved as compared to the INL of a non-DEM DAC with the same current source elements. On the other hand, for the DDEM DAC, we can tolerate large current element variation while ensuring that the maximum $\overline{INL}[k]$ standard deviation remains adequately low. For this 18-bit DAC, the current element standard deviation can be up to 5%, while the maximum $\overline{INL}[k]$ standard deviation is still within 0.1 LSB, with p=128. However, the maximum $INL[k]$ standard deviation of a normal DAC without DEM can reach 12.8 LSB if σ=5%. That means the DDEM approach can achieve an 18-bit linear "averaged" DAC out of an originally 13-bit linear DAC.

The above proves that the "averaged" DAC output samples are almost uniformly distributed. Thus, the DDEM DAC can be used to virtually generate an almost ideal

ramp $\{\bar{I}[k]\}(k=0,...,N-1)$ by increasing the DAC input code sequentially from 0 to N-1. The variation of the DAC output current among p samples for the same code can be treated as an additive noise to the ideal ramp at the input of the ADC. Since all the p output samples for code k are the summation of k elements from the same Gaussian distribution, they also have Gaussian distribution with center at $\bar{I}[k]$. The distribution is described as:

$$N\left(\bar{I}[k], \frac{(N-k)k}{N}\sigma^2 \cdot I_0^2\right) \tag{3.13}$$

With proper approximation, when p is large, all the output samples of the DDEM DAC obey a distribution with the following probability distribution function (PDF).

$$f(x) = \sum_{k=1}^{N} f(x \mid k) \cdot P(k) \tag{3.14}$$

Here, $f(x|k)$ is the PDF corresponding to (3.13), and $P(k)$ is the probability of each input code k and $P(k)=1/N$. For a DAC with given number of bits n, p and σ are the two key parameters to determine the distribution in (3.14). Although (3.14) is too complicated to simplify analytically, we can draw the overall PDF as a combination of Gaussian PDFs with the aid of MATLAB. Figure 3.1 depicts the output PDF of a 10-bit DDEM DAC. For this example, σ is chosen to be 0.1, and p is set to 64. From this figure, the output PDF is very flat except near the end points. Actually, near the end points, due to the small variances that can be calculated from (3.13), the PDF is discontinuous and fluctuates. The histogram of a given DAC is a realization of such PDF and must also be uniform except near the end points. When using the DDEM DAC for ADC test, the DAC output range should be scaled so that the two ends fall outside the ADC input range.

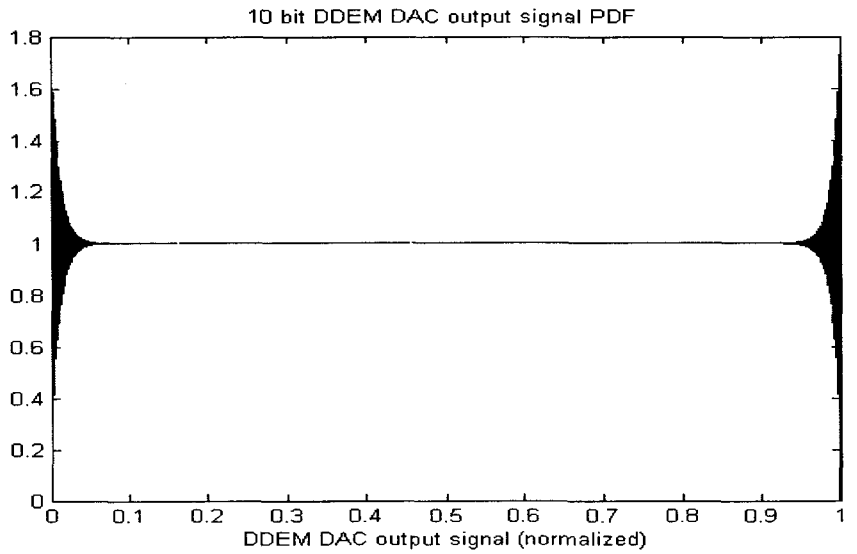


Figure 3.1 Output PDF of 10 bit DDEM DAC

The PDF flatness comes from two essential facts. First, the center (the "averaged" DAC output value $\bar{I}[k]$) of a cluster of p samples, corresponding to each code k, is almost uniformly distributed. Secondly, the individual samples within each cluster are suitably spread out (not too wide and not too narrow). The combined distribution of all these clusters becomes nearly continuous and flat. These two facts are controlled by two key parameters of the DDEM DAC, one is the DDEM iteration number p and the other is the DAC element mismatching standard deviation σ. Generally speaking, larger p helps to achieve a more uniform histogram. On the other hand, to ensure proper spread among each cluster of p samples, extreme values of σ should be avoided. A fairly large range of σ values satisfy this requirement, including typical mismatches of near-minimum-sized current cells. Because of this, an ideal DAC (or a very well designed DAC) is actually not a good candidate for using DDEM. Near-minimum-sized DACs whose elements suffer from significant variations can actually lead to better performance after DDEM is applied.

These resulted conclusions from the analysis based on the "averaged" DAC have already been partly observed and demonstrated in the simulation results shown in Chapter 2. There, we have shown how the DDEM DAC performance improves when the DDEM iteration number, p, increases. It will also be shown in the following section that it will not help to improve the DDEM DAC performance if the current element mismatching error is either too large or too small.

3.2 Simulation-Based DDEM Optimization

More numeric simulations were carried out to further verify the DDEM DAC performance as the ADC static linearity test stimulus source. As guided by the analysis shown previously, the DDEM iteration number, p, and the current element mismatching variation, σ, are the two key parameters that affect the performance of a given DDEM DAC. In addition, several other parameters were also inspected to investigate the DDEM performance. All these parameters are listed as follows.

- Original DAC number of bits (NOB)
- DAC element matching, quantized by the normalized standard deviation, σ
- DDEM iteration number p
- DDEM DAC output range
- Linearity of the ADC under test

Furthermore, the hardware cost and computational complexity are also greatly affected by these parameters. Optimizing the DDEM DAC parameters can help achieve the required ADC test accuracy with a minimum cost. The analysis given in Section 3.1 has pointed out

the basic direction to control the DDEM performance. Guided by this direction, a number of simulations were done to provide data with statistical meaning that can help to find the optimized parameters. In the simulations, the listed parameters were varied, and the DDEM DAC performance was checked. These simulations have provided us some preliminary rules to design an optimal DDEM DAC.

In these simulations, a 14-bit ADC is under test. Initially, a 14-bit thermometer-coded current steering DAC is simulated to send a stimulus to the ADC. The standard deviation of the current element mismatches is set to be σ=0.1. For each input code, the DAC sends p=64 analog samples to the ADC. To make sure DAC output has a uniform histogram that covers the ADC input range, the DAC nominal output range is set to be larger than the ADC input range by 2%. The ADC INL[k] is estimated using a standard histogram method.

To evaluate the INL estimation accuracy, two error parameters, *E1* and *E2*, are defined. *E1* is used to justify the error in INL[k] estimation, and *E2* to determine the error in overall INL estimation.

$$E1 = \max_k \left| INL_{true}[k] - INL_{est}[k] \right| \tag{3.15}$$

$$E2 = \left| \max_k \left| INL_{true}[k] \right| - \max_k \left| INL_{est}[k] \right| \right| \tag{3.16}$$

One simulation result that includes the true INL[k], estimated INL[k], and their difference is given in Figure 3.2. For this result, *E1*=0.2694 and *E2*=0.0714. The estimation errors are adequately low.

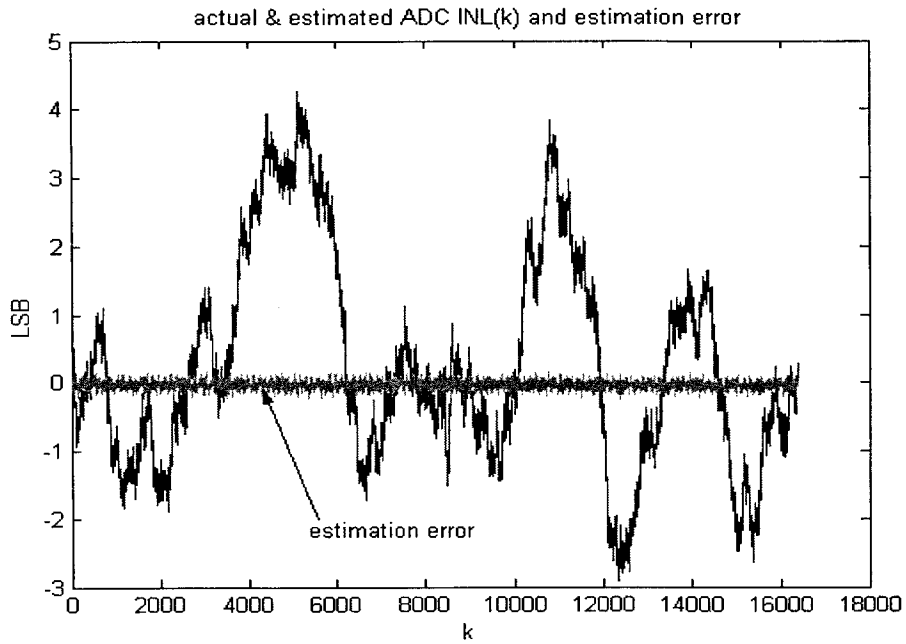


Figure 3.2 True INL[*k*], estimated INL[*k*], and estimation error

Now we will try to optimize the DDEM parameters with both test performance and cost in consideration. In the simulations, we vary the parameters and find the optimal values that minimize *E1* and *E2*.

1) DAC output range expansion

We must make the DAC nominal output expanded to exceed the ADC input range by a certain percentage to guarantee that the DUT input range is totally covered. Denote this percentage as *EXP*. *EXP* cannot be too large; otherwise, the effective resolution to the DUT is reduced. In simulation, *EXP* is varied, and *E1* and *E2* are observed, as shown in Figure 3.3. Though not displayed, *E1* and *E2* are 14 and 10 LSBs respectively when *EXP*=0. We can see that when *EXP* is either larger than 10% or less than 0.1%, *E1* and *E2* get large. It is safe to select *EXP*=2%.

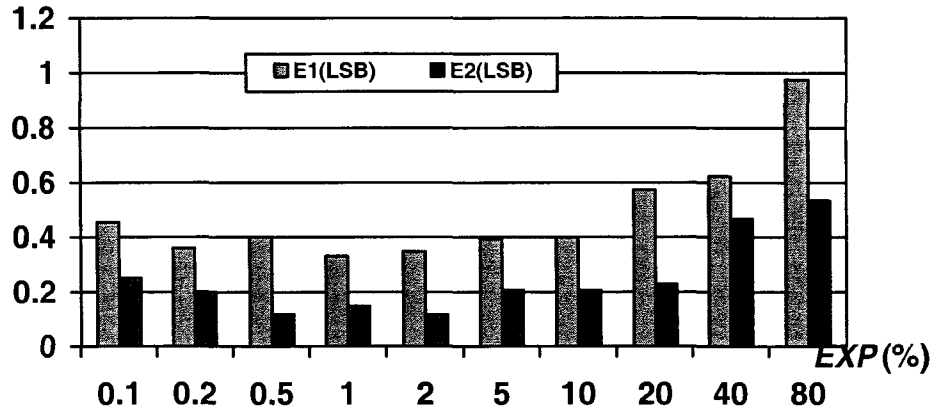


Figure 3.3 Estimation errors versus output range expansion

2) DDEM iteration number

Fix all the other parameters, and change p. The simulation results are shown in Figure 3.4. It is obvious that increasing p can reduce the estimation error. Also notice that both *E1* and *E2* are approximately halved each time when p is doubled, up to p = 64. On the other hand, increasing p also means increasing test time and computation complexity. Normally, p=64 is an acceptable value for both accuracy and cost. When p=64, the INL estimation error is only about 0.1 LSB.

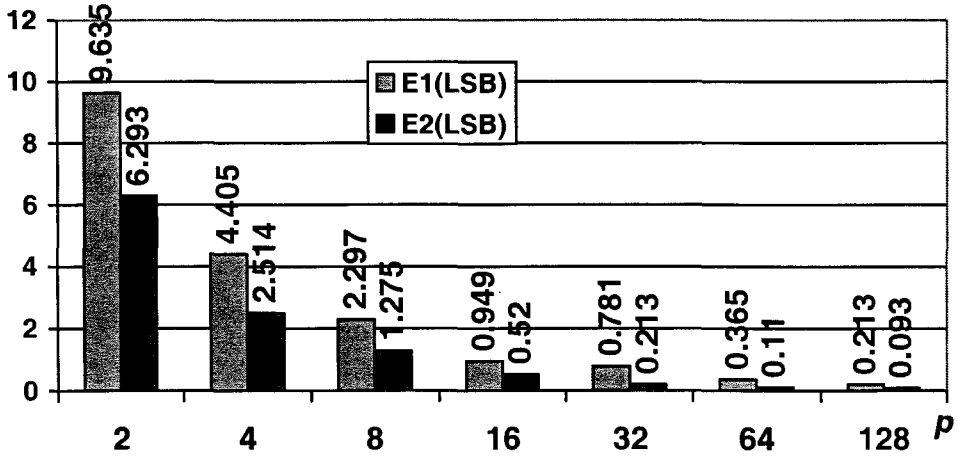


Figure 3.4 Estimation errors versus p

3) DAC current element mismatching

It is obvious that the DAC current element mismatching affects the estimation accuracy greatly. Figure 3.5 shows the estimation errors as the current element standard deviation σ varies. Small estimation errors are achieved when σ is 0.03 to 0.1, implying that 3σ current source mismatches in the 10% to 30% range. Such current sources are some of the easiest to achieve with minimum hardware overhead.

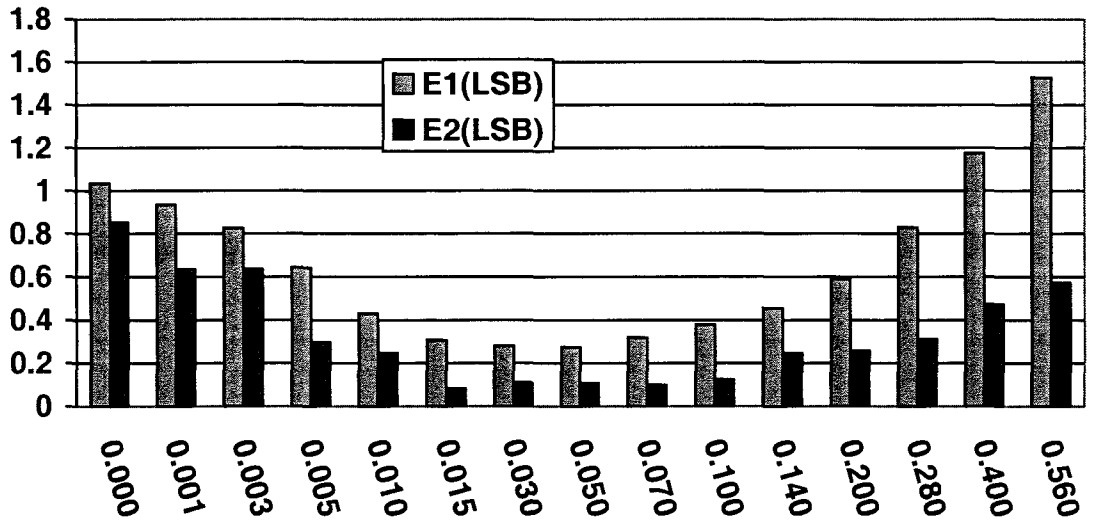


Figure 3.5 Estimation errors versus σ

4) DAC number of bit (NOB)

Increasing the DAC number NOB while maintaining other conditions will increase the DAC accuracy and, therefore, increase the test accuracy. This is verified in simulation by changing DAC NOB while keeping other parameters (σ=0.05, p=64). The result is shown in Figure 3.6. We can also see that using a 13-bit DDEM DAC to test a 14-bit ADC, the INL estimation error is about 0.25 LSB with hardware reduction by half.

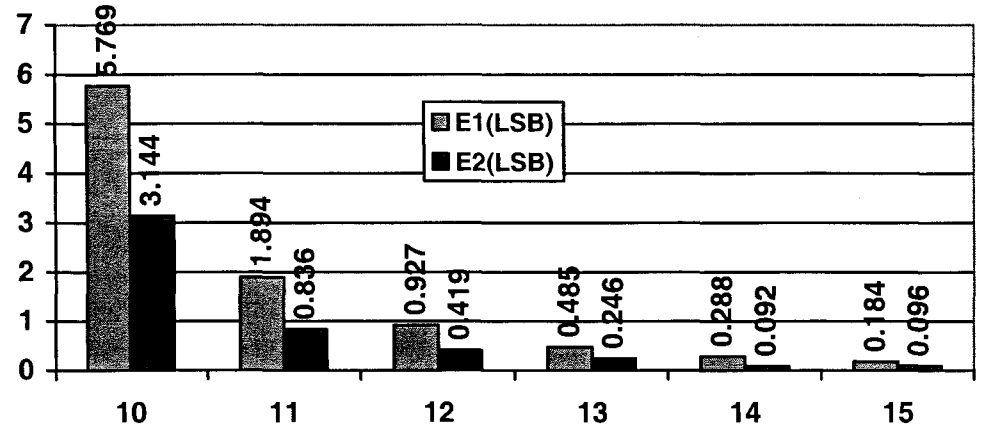


Figure 3.6 Estimation errors versus DAC NOB

5) ADC (DUT) error amount

The previous discussion suggests that to test a 14-bit ADC with 14-bit DAC using DDEM, the following parameter values can be chosen: *EXP*=2%, p=64 and σ=0.05. In this part, 6 ADCs with large INL variation are simulated, and the DDEM approach with these selected parameters is applied. Results given in Figure 3.7 Estimation errors for ADC with large INL variation show that the INL estimation error is maintained at a low level, no matter how the true INL of the DUT varies. This validates the robustness of the DDEM approach.

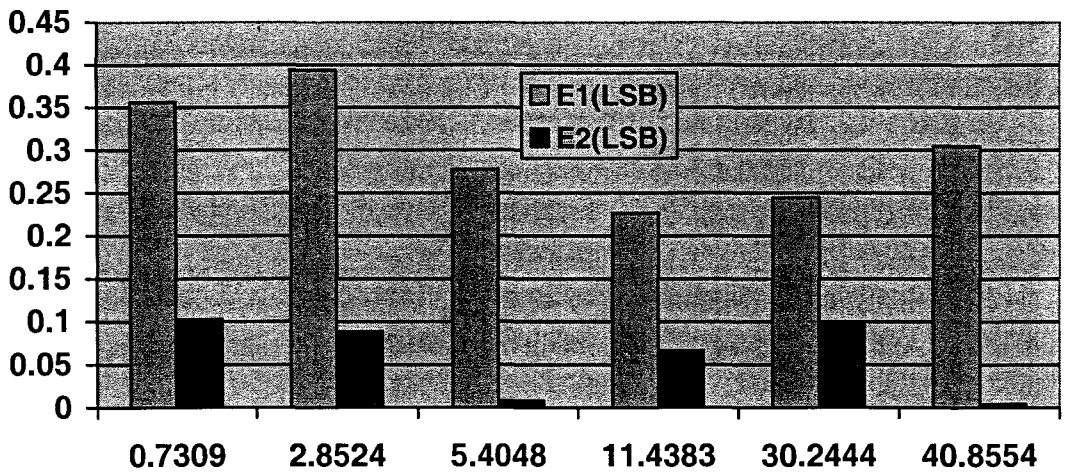


Figure 3.7 Estimation errors for ADC with large INL variation

The previous simulation results have suggested the optimal values for p, σ, and the DAC output range expansion percentage based on the consideration of both test cost and accuracy. The simulation results also show that a 13-bit DAC with DDEM can be used to test a 14-bit ADC with acceptable accuracy and one-half hardware reduction. The DDEM approach is robust for ADCs with various linearity errors. To test ADCs with resolution other than 14 bits, the same simulation can be carried out to find the optimal DDEM DAC parameter settings.

Although such optimization is quite effective, it requires a large amount of simulation data. This is limited by the analysis given in this chapter, since it can only point out the direction of optimal DDEM DAC design. It would be necessary to analyze the DDEM DAC output distribution more rigorously so that we can directly quantify the DDEM DAC performance with respect to the key parameters. A rigorous analysis will be given in the next chapter.

Chapter 4 DDEM Output Distribution

The analysis in Section 3.1 answers the question of why greatly improved performance as the ADC test stimulus source can be achieved from a low-linearity/low-resolution DAC by using the DDEM technique. However, the actual DDEM DAC output histogram has not been directly characterized yet. It is still necessary to directly inspect the distribution of DDEM DAC output voltage samples with the criterion set in (1.3). With the more rigorous analysis shown in the following section, we should be able to tell explicitly how much performance can be expected from a given DDEM DAC. Consequently, a systematic approach for cost-effective DDEM DAC design will be developed. The analytical results are given in Section 4.1, followed by the discussion on DDEM DAC design in Section 4.2.

4.1 DDEM Output Distribution

As stated in Section 1.2, the ADC test performance using a DDEM DAC as the stimulus source will be determined by how the DDEM DAC output voltage samples are distributed along the ADC input voltage range $[V_{\min}, V_{\max}]$. In the following, the ADC testing performance using the DDEM DAC will be evaluated by deriving $e(V_t)$, defined in (1.3), for any voltage, V_t, in ADC input range $[V_{\min}, V_{\max}]$.

We first expand the number of current sources virtually to $2N$ by letting $i_{N+r} = i_r \ \ (r = 1,...,N)$. Then, virtually, we have $2N$ current sources:

$$i_1, i_2, ..., i_N, i_{N+1}, ..., i_{2N}$$

Let $V_0[0]=0$ and $V_0[k]=R_C\cdot\sum_{r=1}^{k} i_r ...(k=1,...,2N)$, in which R_C is the output resistance of the DDEM DAC. Note that the first half of the sequence is the output voltage sequence of a regular n-bit DAC. Let $V_m = V_0[N]$, and V_m is the maximum output of the DDEM DAC.

Define $LSB = V_m / N$.

Now, define $INL[k]$ and $DNL[k]$ for the original DAC without DDEM. Let

$$INL[k]=\left(V_0[k]-k\cdot LSB\right)/LSB.....(k=0,...,2N)$$

and $$DNL[k]=\left(V_0[k]-V_0[k-1]-LSB\right)/LSB.....(k=1,...,2N)$$

From this definition, we have:

$$INL[k]=\sum_{r=1}^{k} DNL[r].....(k=1,...,2N) \tag{4.1}$$

$$\sum_{r=k}^{k+N-1} DNL[r]=0.....(k=1,...,N) \tag{4.2}$$

With the DDEM Cyclic Switching Sequence, the DAC outputs $p\cdot N$ output voltage samples, which can be indexed by two control words: one is the input code k ($0\le k\le N$-1), and the other is the iteration code d ($1\le d\le p$). These $p\cdot N$ output voltage samples can further be decomposed into p ramps with N samples in each ramp.

The 1st ramp is given by $R_C\cdot\sum_{r=1}^{k} i_r \ \ (k=1,...,N)$. We rewrite this sequence as $\left\{V^{(1)}[k]\right\}$. Actually, $\left\{V^{(1)}[k]\right\}=\left\{V_0[k]:1\le k\le N\right\}$

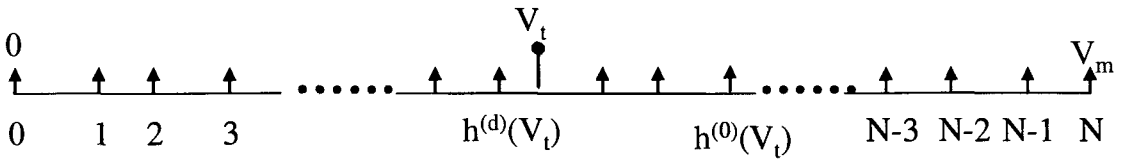
The d^{th} ($1 \le d \le p$) ramp is given by $R_C \cdot \sum_{r=1+q(d-1)}^{k+q(d-1)} i_r \quad (k = 1,...,N)$, or rewritten as

$$\left\{V^{(d)}[k]\right\} = \left\{V_0[k+q(d-1)] - V_0[q(d-1)] : 1 \le k \le N\right\}.$$

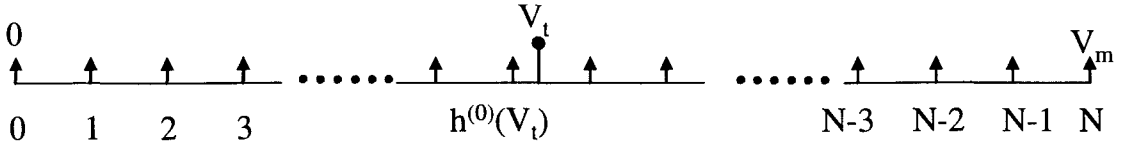
For an arbitrary voltage V_t less than V_m, there is a corresponding DAC input code that will generate a DAC output voltage in the d-th ramp that is equal to or just below V_t. Let $h^{(d)}(V_t)$ denote this DAC input code, then $h^{(d)}(V_t)$ is also the number of elements in $\left\{V^{(d)}[k]\right\}$ that are less than V_t.

$$h^{(d)}(V_t) = \left|\left\{V^{(d)}[k] : V^{(d)}[k] \le V_t, 1 \le k \le N\right\}\right|.$$

The definition of $h^{(d)}(V_t)$ can be explained using Figure 4.1.a. In this figure, we use an axis with N+1 mark arrows to represent the d^{th} ramp sequence. The relative position of the k^{th}($1 \le k \le N$) arrow on the axis represents the value of $V^{(d)}[k]$. By definition, V_t is located between the $[h^{(d)}(V_t)]$-th sample (arrow) and the $[h^{(d)}(V_t)+1]$-th sample.



a) Output sequence of the dth ramp of the DDEM DAC



b) Output sequence of an ideal DAC

Figure 4.1 DAC output sequences

Let $h^{(0)}(V_t) = floor(N \cdot V_t / V_m) = |\{k:\ k \cdot LSB \le V_t, 1 \le k \le N\}|$, in which $0 \le h^{(0)}(V_t) \le N$. $h^{(0)}(V_t)$ is marked on the axis in Figure 4.1.b with uniformly spaced arrows, corresponding to the output voltages of an ideal DAC.

If the DDEM DAC is ideal, we should have $h^{(d)}(V_t) = h^{(0)}(V_t)$ for any *d*. However, for a non-ideal DAC, $h^{(0)}(V_t)$ may be different from $h^{(d)}(V_t)$, as shown in Figure 4.1.a. The difference between V_t and the $[h^{(0)}(V_t)]$-th sample in the sequence $\{V^{(d)}[k]\}$ reflects the DAC INL at code $h^{(d)}(V_t)$ and may be quite large. This voltage difference divided by the DAC LSB can be used to approximate the code difference between $h^{(0)}(V_t)$ and $h^{(d)}(V_t)$. Here, the approximation is based on the assumption that the local DNL of the DAC sequence is not large, and this assumption is valid for thermometer-coded DAC design. This approximation error can be viewed as a quantization error and is distributed over the range of $(-0.5, 0.5]$. Denote this error as $\varepsilon^{(d)}$. Thus, we have the following expression for $h^{(d)}(V_t)$:

$$h^{(d)}(V_t) = h^{(0)}(V_t) + \frac{V_t - V^{(d)}\left[h^{(0)}(V_t)\right]}{LSB} + \varepsilon^{(d)} \qquad (1 \le d \le p) \qquad (4.3)$$

For every *d*, if we apply (4.1), we have:

$$V^{(d)}[h^{(0)}(V_t)] = V_0[h^{(0)}(V_t) + q(d-1)] - V_0[q(d-1)]$$

$$= LSB \cdot \left(INL[h^{(0)}(V_t) + q(d-1)] - INL[q(d-1)] + h^{(0)}(V_t)\right)$$

$$= LSB \cdot \left(\sum_{k=1}^{h^{(0)}(V_t)} DNL[k + q(d-1)] + h^{(0)}(V_t) \right)$$

Substitute this into (4.3), we have:

$$h^{(d)}(V_t) = \frac{V_t}{LSB} - \sum_{k=1}^{h^{(0)}(V_t)} DNL[k + q(d-1)] + \varepsilon^{(d)} \quad (1 \le d \le p) \tag{4.4}$$

Now calculate $h(V_t)$ and $e(V_t)$ as defined in Section 1.2. We consider the range between 0 and V_m, that is $V_{max}=V_m$ and $V_{min}=0$. By the definition given in Section 1.2, $h(V_t)$ denotes the total number of DDEM DAC voltage samples that are less than V_t.

$$h(V_t) = \left| \left\{ V^{(d)}[k] : V^{(d)}[k] \le V_t, 1 \le d \le p, 1 \le k \le N \right\} \right|$$

$h(V_t)$ can be obtained by summing up all the $h^{(d)}(V_t)$:

$$h(V_t) = \sum_{d=1}^{p} h^{(d)}(V_t) = p\frac{V_t}{LSB} - \sum_{d=1}^{p} \sum_{k=1}^{h^{(0)}(V_t)} DNL[k + q(d-1)] + \sum_{d=1}^{p} \varepsilon^{(d)}$$

If $h^{(0)}(V_t) = q \cdot t + m$ $(0 < m \le q, 0 \le t < p, t \,\&\, m \in Z)$, together with (4.2), we finally have:

$$h(V_t) = \sum_{d=1}^{p} h^{(d)}(V_t) = p\frac{V_t}{LSB} - \sum_{d=1}^{p} \sum_{k=1}^{m} DNL[k + q(d-1)] + \sum_{d=1}^{p} \varepsilon^{(d)} \tag{4.5}$$

Note that most part of $h^{(d)}(V_t)$ is cancelled by each other after summation when applying (4.2). This is the major reason that DDEM approach can reduce the total amount of error. With (4.5), we can derive the error expression $e(V_t)$. For the range [0, V_m], $e(V_t) = h(V_t) - C_h \cdot V_t$, the coefficient C_h is given by $C_h = \frac{pN}{V_m}$ for this DDEM DAC. For any V_t that is not close to 0 or V_m,

$$e(V_t) = p\frac{V_t}{LSB} - \sum_{d=1}^{p}\sum_{k=1}^{m} DNL[k + q(d-1)] - \frac{pN}{V_m}\cdot V_t + \sum_{d=1}^{p}\varepsilon^{(d)}$$

$\frac{pN}{V_m}\cdot V_t$ is exactly $p\frac{V_t}{LSB}$ by definition. If we change the order of DNL[k] summation, we have:

$$e(V_t) = -\sum_{k=1}^{m}\sum_{d=1}^{p} DNL[k + q(d-1)] + \sum_{d=1}^{p}\varepsilon^{(d)} \tag{4.6}$$

This tells us that $e(V_t)$ is made of two terms. We will first discuss the first term. A careful examination of the first tem reveals that it is the INL of a q-level DAC, but measured in LSB of the N-level original DAC. As depicted in Figure 4.2, the q-level DAC is obtained by combining the corresponding current sources from each of the p columns. Obviously, the unit current source of the q-level DAC is p times of that of the N-level DAC.

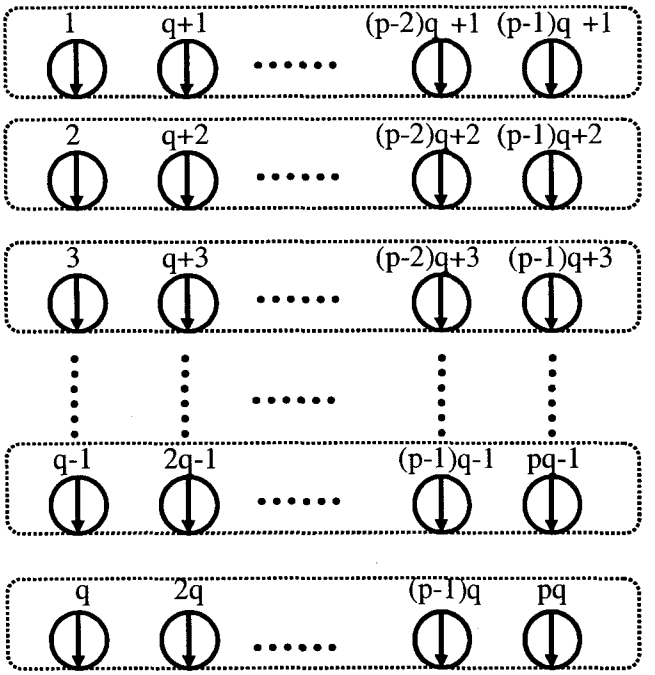


Figure 4.2 A q-level DAC by grouping N-level DAC elements

In Figure 4.2, we depict and index the $N=pq$ current elements of the original N-level DAC, and sort them into groups of p to make a q-level DAC. If the current source mismatching errors are mainly due to random process errors, then the INL variance of an N-level thermometer-coded DAC with normalized element variance σ^2 is given by $\alpha N\sigma^2$ in which α is a constant. Again, if the current source errors are random and linearly independent, then the normalized element variance of the q-level DAC is given by σ^2 / p and the q-level DAC's INL variance is given by $\alpha q\sigma^2 / p$. Note that this number is measured in the LSB of the q-level DAC. If we measure it in the LSB of the original DAC, it needs to be multiplied by p^2, since the q-level DAC's unit current source is p time lager. It follows that the q-level DAC has an INL given by $\left(\alpha q\sigma^2 / p\right)\cdot p^2 = \alpha qp\sigma^2 = \alpha N\sigma^2$, which is at the same level as the N-level DAC's INL when both are measured in LSB of the N-level DAC. Therefore, we can conclude that the first item in (4.6) is bounded by INL of the n-bit DAC.

The second term in (4.6) is caused by the quantization effect. Without knowing the actual quantization errors, we can approximately assume that the individual quantization errors share the same distribution with a certain standard deviation. If they are uncorrelated, the variance of the second term grows proportionally with p. With this we can conclude that the second term in (4.6) approximately grows with $\sqrt{p}$. However, if they are correlated, the second term will be quite large when p gets large. In the extreme case that $\varepsilon^{(d)}$'s are fully correlated, the second term in (4.6) increases linearly with p. A precise estimation of this quantization error term requires knowing the exact distribution of $\left\{\varepsilon^{(d)}\right\}$. However, we can approximately assume that $\left\{\varepsilon^{(d)}\right\}$ have a standard deviation of $\frac{1}{\sqrt{12}}$, since the quantization

error is normally treated to have a uniform distribution over the range of $(-0.5, 0.5]$ with the standard deviation equal to $\frac{1}{\sqrt{12}}$. Based on this assumption, the standard deviation of the second term in (4.6) is given by $\frac{\sqrt{p}}{\sqrt{12}}$ if $\varepsilon^{(d)}$'s are independent.

When p is small, the first term in (4.6) dominates due to large INL of the original DAC since DDEM is applied to a low-linearity DAC. We can ignore the quantization error term and approximate $e(V_t)$ to:

$$e(V_t) \approx -\sum_{k=1}^{m}\sum_{d=1}^{p} DNL[k + q(d-1)] \tag{4.7}$$

Since the full range of $h(V_t)$ is $p \cdot N$, the percentage error in the DDEM DAC output voltage sample's distribution is bounded by $INL/p \cdot N$. This means that the DDEM DAC gives an equivalent accuracy of n_{eq} bits, where n_{eq} is given by:

$$n_{eq} \approx \log_2\left(pN/\left(INL/0.5\right)\right) \approx n + 1 + \log_2 p - \log_2 INL$$

In the real situation, n_{eq} is less than this since the DAC nominal output range is expanded to ensure the test reliability, as discussed in Chapter 3. Assume that the DAC nominal output range is larger than the input range of the ADC under test by a factor of α. Then the percentage error in the DDEM DAC output voltage sample's distribution is bounded by $\frac{INL}{pN/(1+\alpha)}$. This gives:

$$n_{eq} \approx \log_2\left(\left(pN/(1+\alpha)\right)/\left(INL/0.5\right)\right) \approx n + 1 + \log_2(1-\alpha) + \log_2 p - \log_2 INL$$

The DAC effective number of bits (ENOB) is defined by $n - \log_2\left(INL/0.5\right)$, giving:

$$n_{eq} \approx ENOB_{DAC} + \log_2(1-\alpha) + \log_2 p$$

Here, the subscript "DAC" is added for clarity. This tells us that n_{eq} increases by 1 bit every time p doubles when p is small. However, as p becomes very large, the second term can become larger than the first term, depending on the statistical behavior of $\{\varepsilon^{(d)}\}$. For each fabricated DDEM DAC, how the second term varies with p should be relatively fixed. When p is adequately large, the second term $\sum_{d=1}^{p} \varepsilon^{(d)}$ dominates. This means if p is very large, the improvement in n_{eq} will be less than 1 bit when doubling p. Under the situation that $\varepsilon^{(d)}$'s are independent, the standard deviation of $\sum_{d=1}^{p} \varepsilon^{(d)}$ increases linearly with $\sqrt{p}$. When it is normalized by $p \cdot N$, the magnitude of $\frac{\sum_{d=1}^{p} \varepsilon^{(d)}}{p \cdot N}$ is inversely proportional to $\sqrt{p}$. This means that, when p is large, n_{eq} increases by 0.5 bit every time p doubles. There exists a transition point for p that n_{eq} incremental speed changes from 1 bit to 0.5 bit when doubling p. Denote this transition point as p_T. p_T happens when $\frac{\sqrt{p}}{\sqrt{12}}$ is comparable to the INL of the original DAC. When p is equal to p_T, $\sum_{d=1}^{p} \varepsilon^{(d)}$ is comparable to the first term in (4.6) in magnitude, and the summation of these two terms gives $n_{eq} \approx ENOB_{DAC} + \log_2(1-\alpha) + \log_2 p_T - 1$ in the worst case that their magnitudes are added.

In summary, we have the following formula for the equivalent linearity of a DDEM DAC:

$$n_{eq} \approx \begin{cases} ENOB_{DAC} + \log_2(1-\alpha) + \log_2 p & p < p_T \\ ENOB_{DAC} + \log_2(1-\alpha) + \log_2 p_T - 1 & p = p_T \\ ENOB_{DAC} + \log_2(1-\alpha) + \log_2 p_T - 1 + 0.5\log_2(p/p_T) & p > p_T \end{cases}$$

in which p_T is given by solving $\frac{\sqrt{p_T}}{\sqrt{12}} \approx INL_{DAC}$. In the real situation, α can be set to be as small as 0.02, as discussed in Chapter 3. Therefore, $\log_2(1-\alpha)$ is very small. Ignore this term, and we have:

$$n_{eq} \approx \begin{cases} ENOB_{DAC} + \log_2 p & p < p_T \\ ENOB_{DAC} + \log_2 p_T - 1 & p = p_T \\ ENOB_{DAC} + \log_2 p_T - 1 + 0.5\log_2(p/p_T) & p > p_T \end{cases} \quad (4.8)$$

Equation (4.8) is obtained based on the assumption that $\varepsilon^{(d)}$'s are independent. The correlation among $\varepsilon^{(d)}$'s can affect in two aspects. First, p_T will be smaller than what is expected. Secondly, when p is larger than p_T, we have:

$$n_{eq} \approx ENOB_{DAC} + \log_2 p_T - 1 + \alpha\log_2(p/p_T)$$

in which α is less than 0.5. In the extreme case that $\varepsilon^{(d)}$'s are totally correlated, p_T is given by solving $\frac{p_T}{\sqrt{12}} \approx INL_{DAC}$ and $\alpha = 0$. This causes the DDEM DAC performance to saturate after p passes a quite small number p_T.

Here, we can see that a careful design should try to make $\varepsilon^{(d)}$'s independent to maximize the performance. It is important to have randomly distributed DAC source elements to make the DDEM approach effective. In circuit design, we should try to minimize the highly-correlated

errors among DAC current elements with proper layout and element routing strategy, as discussed later.

4.2 Cost-Effective Design of DDEM DAC

Equation (4.8) given above is a good estimation of the DDEM DAC test capability. It is shown that the equivalent linearity increases by 1 bit when the original DAC ENOB increases by 1 bit. It is also shown that the equivalent linearity increases by 1 bit if p doubles when p is small and then 0.5 bit when p gets large. This equation gives the hint on how to optimize a DDEM DAC design with a targeted performance under certain constraints. In this section, the DDEM optimal design will be presented based on (4.8).

A typical situation is to design a stimulus source with a certain performance requirement within a limited die area. Without loss of generalization, we target designing a DDEM DAC capable of testing 14-bit ADCs with test error bounded by 0.25 LSB. The DDEM DAC will be fabricated in 0.5-um CMOS process and the available die area is about $2mm^2$. We use this example to present a systematic approach to design a cost-effective DDEM DAC.

Although (4.8) gives a good theoretical prediction, some extra simulations is also needed to help design a cost-effective DDEM DAC, as some non-ideality might not be covered in the theoretical analysis. In the following part of this section, simulations are adopted as a supplement and aid the theoretical analysis.

From (4.8), only original DAC ENOB and the iteration number p affect the DDEM DAC performance. More specifically, if minimum-sized current elements are used in the DAC, then from the design viewpoint, DAC ENOB is determined only by its number of bits (NOB)

and the layout. In this section, we focus the discussion on how to select DAC NOB, DDEM iteration number p, and DAC layout to make the design optimal.

1) DDEM DAC NOB

It is obvious that the DDEM DAC test capability can be enhanced by increasing DAC NOB (denoted as n). Assume all the DAC current elements have the same normal distribution with normalized standard deviation σ. Then, if DAC NOB increases by 1 bit, its INL will increase by $\sqrt{2}$ times as the INL standard deviation is given by $\alpha\sqrt{N}\sigma$ in which α is a constant. By the definition of ENOB, DAC ENOB increases only by half bit. Note that the maximum p also doubles when n increases by 1. Hence, increasing DAC NOB by 1 bit can lead to 1.5-bit increment in test capability.

However, we cannot afford very large DAC NOB due to the die area limitation. Since we are attempting to limit the die area to about 2 mm^2 using 0.5-μm CMOS technology, in this DDEM DAC design example, the maximum affordable DAC NOB is 12 bits with previous design experience if the minimum-sized PMOS current source element is used. Calculations show that the minimum-sized PMOS current source element in AMI05 0.5-μm CMOS technology gives an element standard deviation of about 6%. This number will be used in the following discussions.

2) DDEM iteration number

By (4.8), if DAC ENOB is fixed, the DDEM DAC equivalent linearity increases by 1 bit every time p doubles when p is small, and then only 0.5 bit after p passes the transition value p_T given the DAC elements are totally independent. In the real situation, the

correlation in DAC elements makes p_T smaller than the expected value and the DDEM DAC performance saturate when p is large.

Simulation was done to compare the 12-bit DDEM DAC performance with different DDEM iteration numbers p, and the equivalent linearity was quantized to draw a curve. In the simulation, the DAC has 4096 randomly generalized current elements with the same normal distribution, and the normalized standard deviation is set to be 6%. The DDEM DAC is used to test a 14-bit ADC. Although the simulation result of only one DAC-ADC pair is shown here, the result is repeatable for all the random generalized DAC-ADC pairs in simulation.

The original DAC's INL[k] plot is shown in Figure 4.3. The DAC's INL is about 4 LSB. By definition, this DAC's ENOB is about 9 bits. When p is 8, the equivalent linearity should be about 12 bits theoretically by applying (4.8). On the other hand, we have $p_T \approx 192$ by solving $\frac{\sqrt{p_T}}{\sqrt{12}} \approx INL_{DAC}$. Since $log_2 p$ can only be an integer, we can take $p_T = 128$.

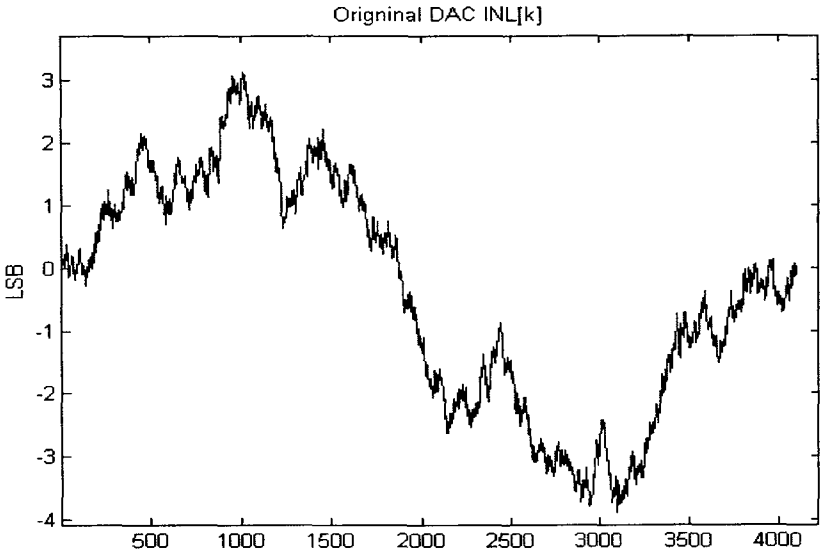


Figure 4.3 Original DAC INL[k] (simulated)

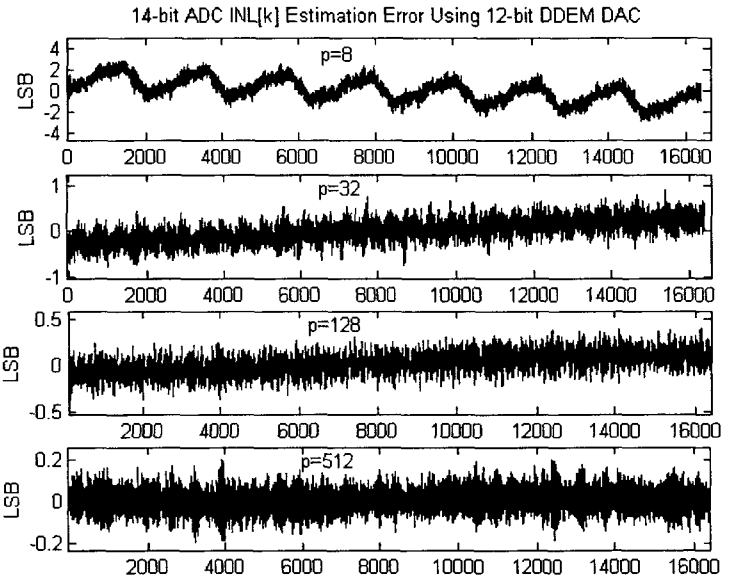


Figure 4.4 ADC test errors under different p

The DDEM DAC's output samples are used as the stimulus for the 14-bit ADC linearity test. The ADC INL[k] test errors (difference between estimated INL[k] and true INL[k]) are compared under different DDEM iteration number p ranging from 8 to 1024. Figure 4.5 shows ADC INL[k] test error curves when p is 8, 32, 128, and 512, respectively. It is clear that test errors decrease when p increases.

Under each p, we can calculate the DDEM DAC equivalent linearity based on the test errors. Basically, if we use a 14-bit ideal DAC to test a 14-bit ADC, the maximum INL[k] test error (absolute value) will be about 1 LSB. Based on this, if the test error is ε LSB, we can claim that the test stimulus source has an equivalent linearity of $14-\log_2 \varepsilon$ bits. With this definition, the 12-bit DDEM DAC equivalent linearity under different p is calculated and shown in Figure 4.5. The expected equivalent linearity when p is small calculated from (4.8) is also depicted in Figure 4.5. This curve is expanded to the full comparison range as a reference.

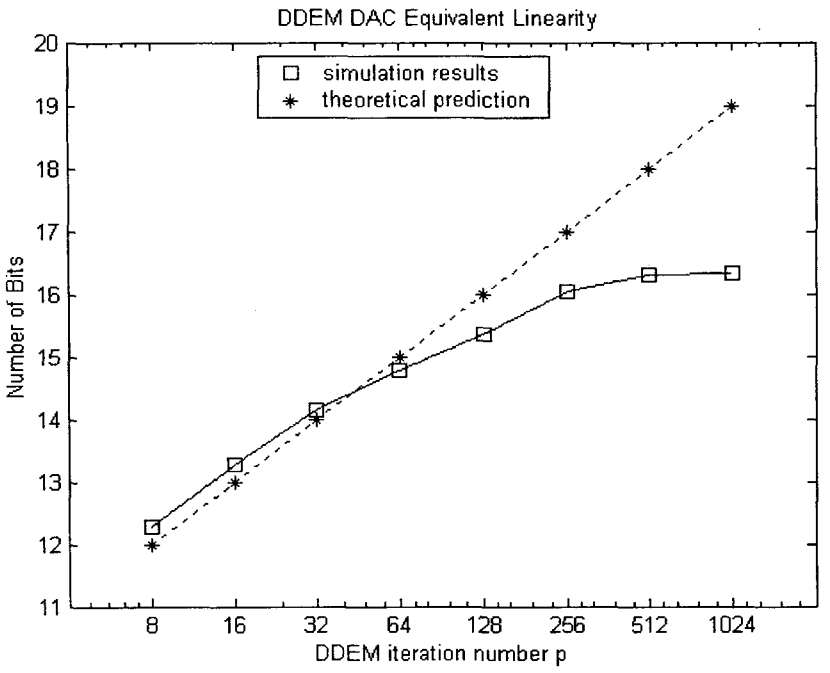


Figure 4.5 DDEM DAC equivalent linearity

In Figure 4.5, the simulated equivalent linearity is 12 bits for p=8. n_{eq} increases by almost exact 1 bit when p changes from 8 to 16 and 16 to 32. The incremental speed for n_{eq} decrease starting (less than 1 bit) from p=64. And starting from p=128, n_{eq} increases by about half bit when p doubles. p=512 gives an equivalent linearity of ~16.3 bits. We can see that the simulation results agree precisely with the theoretical prediction by (4.8).

Both the theoretical prediction by (4.8) and simulation results shown in Figure 4.5 tell us that it is sufficient to take p=128 to have 16-bit equivalent linearity and make the test error less than ¼ LSB when using this DDEM DAC to test 14-bit ADCs. It will be safe to limit the DDEM iteration number p to be 512.

On the other hand, increasing p also means increasing the complexity of the DDEM control logic. It has been found that a DDEM control unit occupies roughly the same die area as a

DAC current source element. If we can limit the maximum available iteration number p, the number of DDEM control units can be reduced by letting multiple DAC elements share the same control unit. In a 12-bit DDEM DAC design, if the maximum available iteration number is not limited by hardware, then a total 2^{12}=4096 control units are required, which will occupy roughly the same die area as the current source elements. However, if the maximum available DDEM iteration number is set to be 512 in the hardware, then only 512 control units are needed, and the required die area is only 1/8 of the area for the current source elements (4096 elements in total). This can save about 44% die area compared to the situation having 12-bit DDEM control.

It follows that it is necessary and sufficient to limit the maximum available DDEM iteration number p to 512 for this 12-bit DDEM DAC design to achieve the targeted test performance while greatly save the chip area. This conclusion is predicted by (4.8), and validated via simulation results.

3) Systematic error reduction

From the previous discussion, the DDEM DAC to be implemented will have 12-bit resolution, and the DDEM iteration number will be limited to 512. With this setup, the DAC will have 4096 current elements. All these elements will be divided into groups of 8, with each group controlled by 1 control unit. Then, in total, we have 512 element groups and 512 control units, and each group has a control unit by side.

The DDEM control will take care of the random mismatching errors in current source elements. However, non-random systematic errors can not be gotten rid of by DDEM. There are two major types of systematic error. One is the error caused by improper layout

arrangement, and the other is the error related to intrinsic device nonlinearities. The first type of systematic error can be reduced by proper layout strategy, which will be shown in the following.

As described in Section 2.2, all the control units (or element groups) will be numbered to form a virtual circle. The formation of the q-level DAC, shown in Figure 4.2, is controlled by this numbering. It has been pointed out by (4.7) that to reduce the error in DDEM DAC output distribution, this q-level DAC should be optimized. Hence, a good layout scheme should try to number the element groups in such a way that the q-level DAC INL error is minimized. In addition, the layout routing complexity should also be taken into consideration. In the real design, due to the large number of control units, element (group) numbering should be adequately simple to keep the routing complexity well under control.

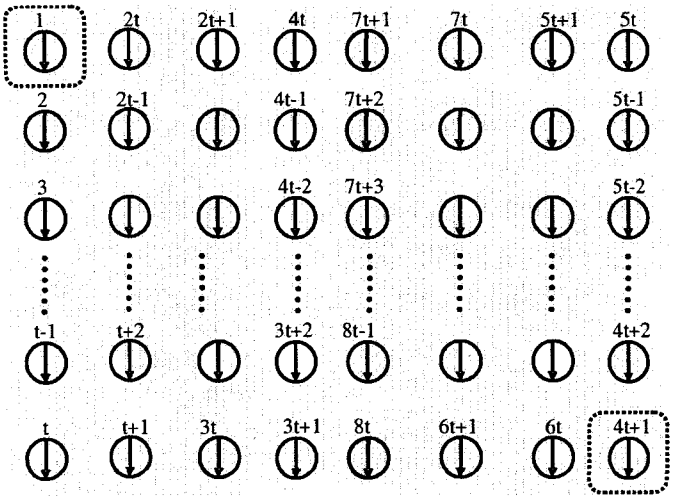
Equation (4.7) is the most essential equation to discover the insight property related to current element numbering/layout. We rewrite (4.7) here and disregard the negative sign:

$$e(V_t) \approx \sum_{k=1}^{m} \sum_{d=1}^{p} DNL[k + q(d-1)] \tag{4.9}$$

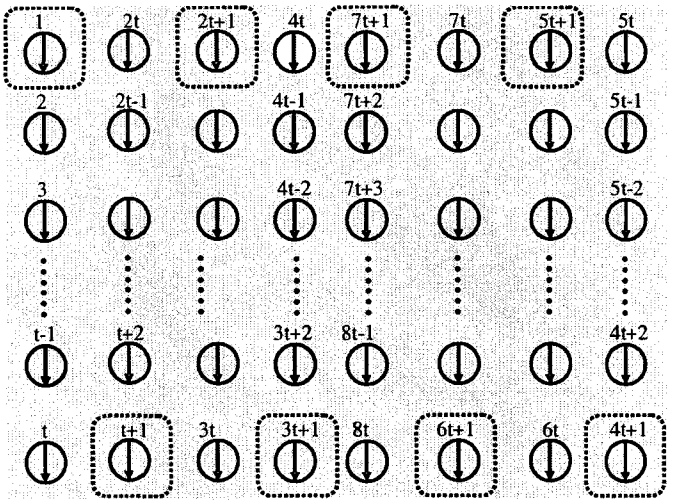
Equation (4.9) tells us that the q-level DAC's INL should be minimized to minimize $e(V_t)$. Totally random errors among DAC elements will not bring a large INL for the q-level DAC, as the random errors can be averaged out when summing every p elements to form a q-level DAC. However, non-random errors, such gradients errors, in current elements might be accumulated when forming the q-level DAC if improper layout is used. In [10], this type of systematic error is estimated by measuring only one part out of the chips from the same run and then compensated for the remaining parts. Such systematic error compensation lowers

the test efficiency. It is significant to minimize the systematic error by proper layout strategy. If the physical placement of the DDEM DAC elements is already fixed, then the layout issue becomes how to number the DAC elements such that the q-level DAC's INL is minimized.

Here, we propose a layout scheme to handle the linear gradient error, as in many cases, the linear gradient error is the major source of error. However, it will be shown that the resulted layout strategy also works for other, nonlinear gradient errors.



(a) 1st element of the q-level DAC when p=2



(b) 1st element of the q-level DAC when p=8

Figure 4.6 DDEM DAC layout scheme

The direct way to eliminate the influence of linear gradient error is to make the q-level DAC's elements share the same "centroid", or, specifically, any of the q-level DAC's elements has the layout center as its "centroid". [30] Assume the physical layout of the current element array is composed of 8 columns, as depicted in Figure 4.6, and take this as an example. Figure 4.6 gives a way to number the current elements. In Figure 4.6, each column contains t elements. All the DAC elements are numbered in a sequence, as shown in Figure 4.6, and this determines the sequence by which the control units or current elements are connected to form a circle. If p=2, the first element of the q-level (q=8t/p) DAC is composed of element 1 and 4t+1, and its "centroid" is exactly the center of the whole layout, as shown in Figure 4.6(a). If p=8, the first element of the q-level DAC is composed of element 1, t+1, 2t+1, ..., 6t+1, and 7t+1. The "centroid" of these 8 elements is still that center point, as shown in Figure 4.6(b). It can be seen that no matter how p changes, any element of the q-level DAC also has the same center point as its "centroid", which makes the q-level DAC insensitive to any linear gradient error.

The layout scheme shown in Figure 4.6 has been verified by simulation. In the simulation, a 12-bit DDEM DAC using this layout scheme is used to test a 14-bit ADC with p equal to 512. The results are shown in Figure 4.7. The first error curve (a) is obtained by using a DDEM DAC with only random mismatching errors. The second error curve (b) is obtained by using the same DDEM DAC with 2% linear gradient error, in both the vertical and horizontal directions, added to the random mismatching errors. The third error curve (c) is obtained by using this DDEM DAC with 2% second-order gradient error added to the random mismatching errors. The maximum ADC INL[k] test errors are quite the same for these three situations (0.2277, 0.2374, and 0.2575 LSB, respectively). This shown that by

using the layout scheme in Figure 4.7, the DDEM DAC is not sensitive to gradient errors (linear or nonlinear).

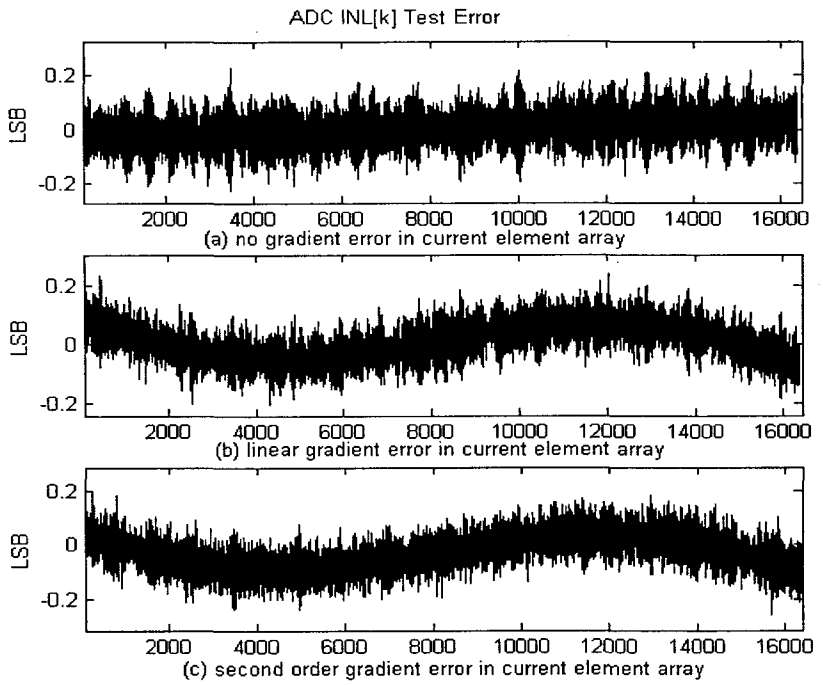


Figure 4.7 ADC INL[k] test errors using DDEM DACs with different gradient errors

The layout scheme shown in Figure 4.7 is robust to gradient errors. Another merit is that this layout scheme is quite easy to implement, as the element/control unit routing will be adequately simple.

4) Systematic error compensation

It has been pointed out that the systematic error caused by device nonlinearity cannot be handled by any layout scheme, since this type of nonlinearity is inevitable. For example, we can use a differential pair to reduce the even-order, nonlinear errors in current source outputs due to the nonlinear characteristic of CMOS transistors. However, the odd-order output nonlinearity still exists. Fortunately, the nonlinear error sequence (output sequence

subtracted by the straight end-point fitline) for a given current steering DAC usually has a highly-predictable shape. In normal situations, the error sequence has an almost symmetric bow shape, as shown in Figure 4.8. It is obvious that the major part of this nonlinear error curve is even-symmetrical. (Note that any form of error curve can be decomposed into an even-symmetric curve and an odd-symmetric curve.) In practice, we use an inversed output voltage sequence to cancel the even-symmetrical nonlinear errors. Consider the output sequence *Seq1* on the left side of Figure 4.8. This output sequence has an even-symmetric, nonlinear error sequence compared to the fitline. When *Seq1* is inversed in value and then re-ordered, we have *Seq2*. *Seq2* also has an even-symmetric error sequence, which exactly cancels the nonlinear error in *Seq1*. In DDEM DAC, both the positive and negative output sequences ($(V_+ - V_-)$ and $(V_- - V_+)$) are used for the ADC test to take advantage of this error cancellation.

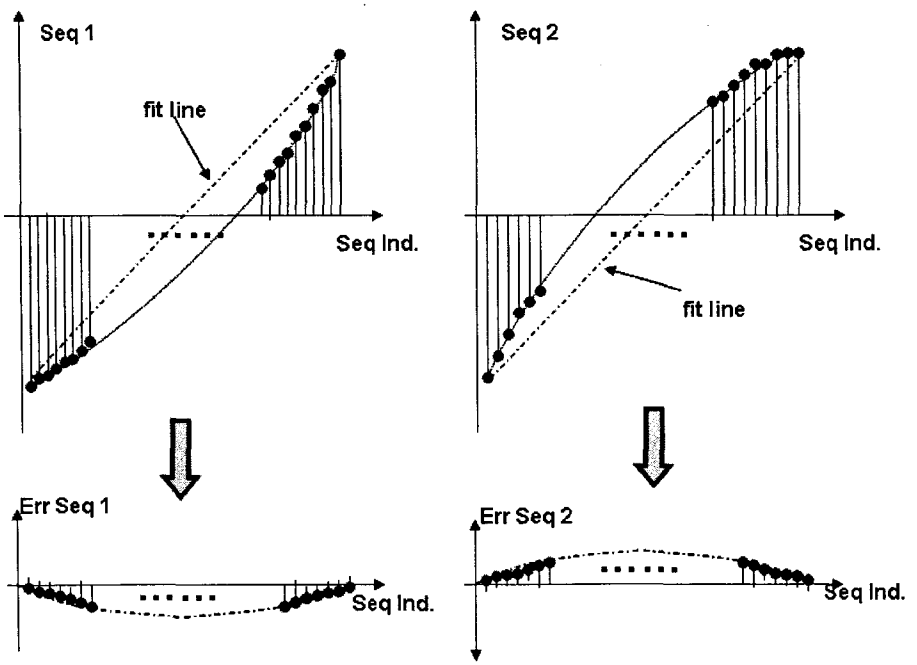


Figure 4.8 Inversed output sequence to cancel even-symmetrical nonlinear error

The above compensation method requires nothing other than the standard histogram-based test. To make it effective, it is critical to maintain the output error sequence as even-symmetrical as possible. In practical circuit design, centroid layout can be used to reduce the odd-symmetrical nonlinear error.

So far, based on the analytical analysis, a systematic approach to design a cost-effective DDEM DAC with certain performance and cost constraints has been given, which covers the key aspects of a DDEM DAC. In the next chapter, two DDEM DAC designs will be presented that follow this approach.

Chapter 5 DDEM DAC Design

Two DDEM DACs have been designed, fabricated, and measured. Both give satisfactory performance, as expected and as shown in the following sections.

5.1 First DDEM DAC Design

The first DDEM DAC was designed only for the purpose of concept proof. It has 8-bit apparent resolution with full-p DDEM control availability (maximum p=256).

In implementing the DDEM DAC, two critical circuit parts are the current source element array and the DDEM control logic circuit. Other design issues, such as gradient effect and output nonlinearity, have also been considered. As a benefit of the DDEM approach, the design is a quick and simple design, and it occupies very small die area.

1) Current source element

There are quite a few current steering DAC structures. Since the element matching issue is not critical with the DDEM approach, we can use a simple structure that uses fewer devices and, hence, less die area. Also, DDEM makes it possible to use small devices. By using a simple structure with small devices, the DAC speed can be very high, since it has small parasitics and, therefore, a small capacitance load.

The current element structure used is the simple single-supply positive-output structure with three PMOS transistors [31], as depicted in Figure 5.1. To balance the output, both M1 and M2, respectively, have their drains connected to an external output resistor. The three references are supplied as $Vu>Vb>Vd$, and one shift register unit (SR) is used to control M2's

gate. If M2's gate is connected to *Vu*, the current in this unit flows through M1 and *Rp* to the ground; if M2's gate is connected to *Vd*, the current flows through M2 and *Rn* to the ground. For the single-ended output mode, the voltage crossing either *Rp* or *Rn* serves as the output. For the differential output mode, the voltage difference between *Rp* and *Rn* serves as the output.

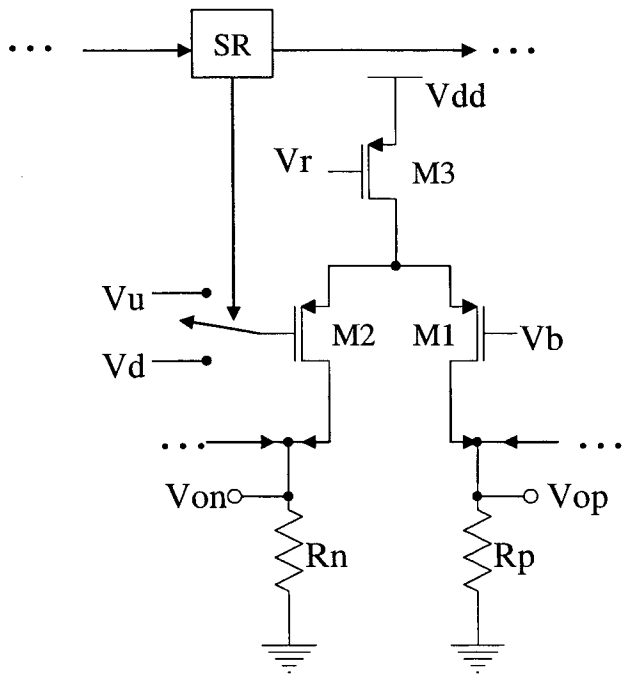


Figure 5.1 Current steering element structure

To maximize the speed, the reference voltages should be chosen properly such that during switching all three transistors stay in the saturation region.

2) DDEM control logic

We described the Cyclic DDEM Switching Sequence in Section 2.2. For each input code k, the DDEM DAC outputs p samples. For all the output samples corresponding to all the N input codes, it is equivalent to make the DDEM DAC output p N-step ramps, with the p

index current sources as the first element. We built the control logic to generate these *p* ramps.

The control logic circuit is just a 256-bit shift register ring, with each unit controlling one current source. Starting from the all-zero state, one of the register units is selected as the index point, and a logic '1' is continuously pumped into this unit. Then, each time the clock signal advances, one more register unit is set to '1'. Thus, the DAC outputs a monotonic ramp voltage by clustering current on *Rp*. In the meantime, the voltage crossing *Rn* is a declining signal. When all register units are set to '1', one RESET signal clears all the units, a different index element is chosen, and the same operation is applied. To achieve high speed with a small die area, the simple shift register unit was adopted, as shown in Figure 5.2. It contains only 6 transistors, with 2 CMOS inverters and 2 NMOS transistor switches in series. Two-phase non-overlapping clock signals are required to drive this shift register. [32]

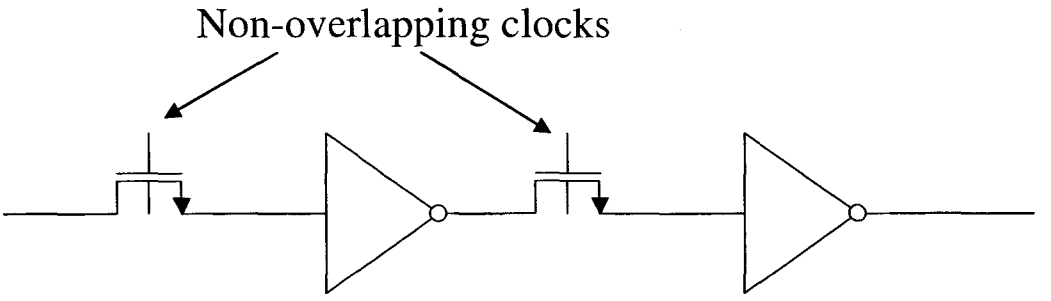


Figure 5.2 6-transistor shift register unit

3) Systematic error minimization and compensation

As pointed out in the previous chapter, the non-periodic mismatching error among current sources can be averaged out by DDEM. It has also been shown by both theoretical analysis and simulation results that two types of systematic errors cannot be totally averaged out by

DDEM. One is the output node nonlinearity error, and the other is the periodic error in the DAC elements. The sources of the first type of error may be the nonlinearity from the MOS transistors or the load resistance nonlinearity. The periodic error may be from the layout gradient effect. We should reduce these two types of errors during the design. For example, in this design, we used the cascoded current source structure to alleviate the output node nonlinearity.

Although the systematic errors cannot be averaged out by DDEM, fortunately, these errors are highly predictable. With a set of chips from the same fabrication run, we may test a few and predict the systematic errors for all the chips from that run (DDEM will take care of the remaining non-systematic errors). We can then compensate for the systematic errors during ADC testing.

5.2 First DDEM DAC Experimental Results

The 8-bit DDEM DAC was fabricated in a 0.50-μm standard CMOS process provided by AMI through MOSIS. The core die size is about 0.9mm×0.9mm for double 8-bit DACs (0.4 mm^2 for each single DAC). The die photo is shown in Figure 5.3. The power supply voltages are 5V for both digital and analog parts. When driving two 22 ohm resistance loads, the power consumption is 260mW for the analog part and 60mW for the digital part, with 0~1V output range at the single ended output nodes (-1~+1V for differential mode). The power dissipation can be dramatically reduced by using larger resistors, as the loads to make output current low, while maintaining the same output voltage range.

To test the DDEM DAC performance, DDEM control signals generated by a pattern generator were applied to the DAC and the DAC output (single-ended mode) is sampled

using a data acquisition board with high-resolution ADCs. Though the DAC can operate at a speed of 10M sample/sec, it was tested with a clock speed of 1 kHz due to the speed limitation caused by the data acquisition board. The DAC output with different iteration number p was collected and stored in computer for performance evaluation.

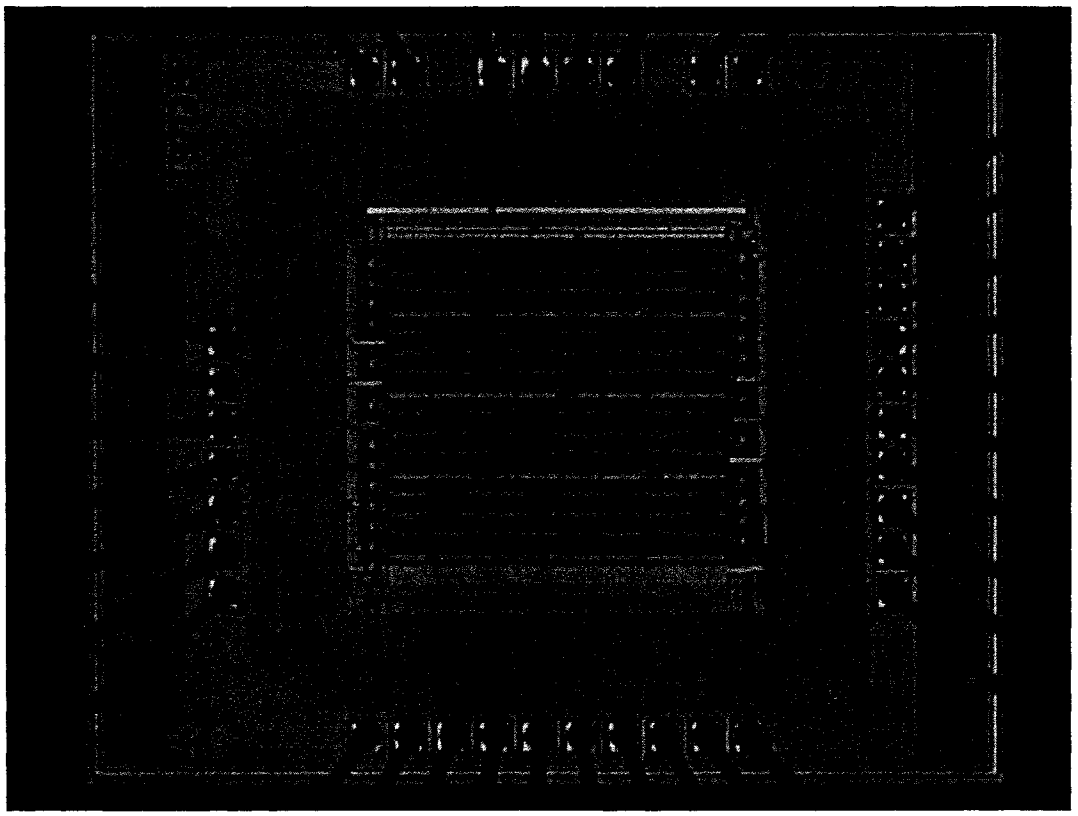


Figure 5.3 Die photo of first DDEM DAC

Figure 5.4 shows that without DDEM, the original 8-bit DAC has an INL error of 10.3 LSB, which means that the original DAC has a linearity of less than 4 bits. The main error source for this DAC is the systematic nonlinearity. We can estimate the systematic error by measuring one or more samples from the fabricated chips. We can then apply the same systematic error compensation to all the chips from the same run, which brings the DAC linearity to about 5 bits linear without DDEM.

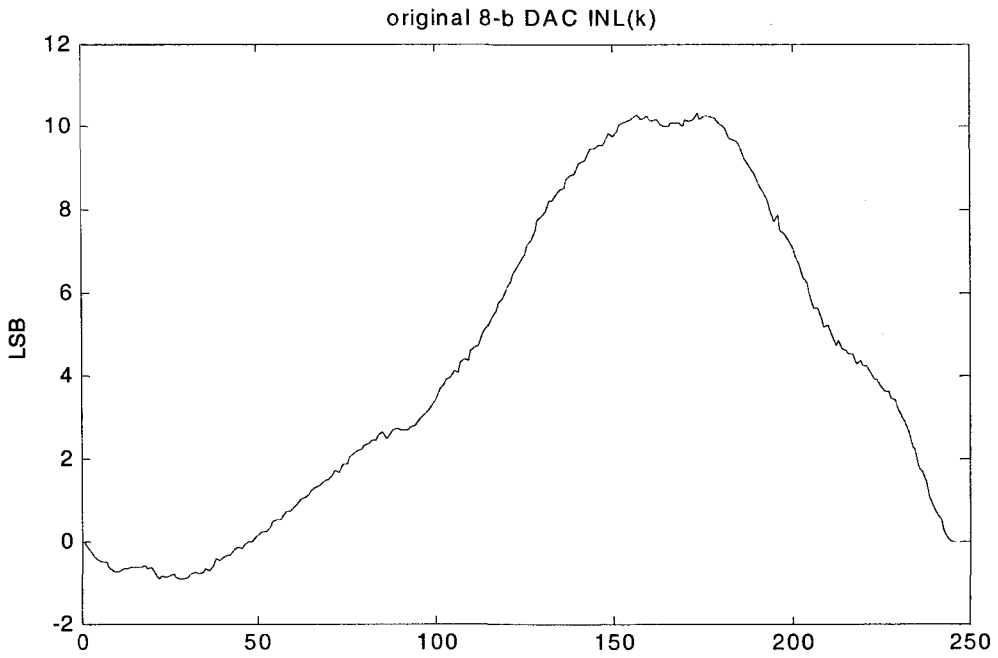
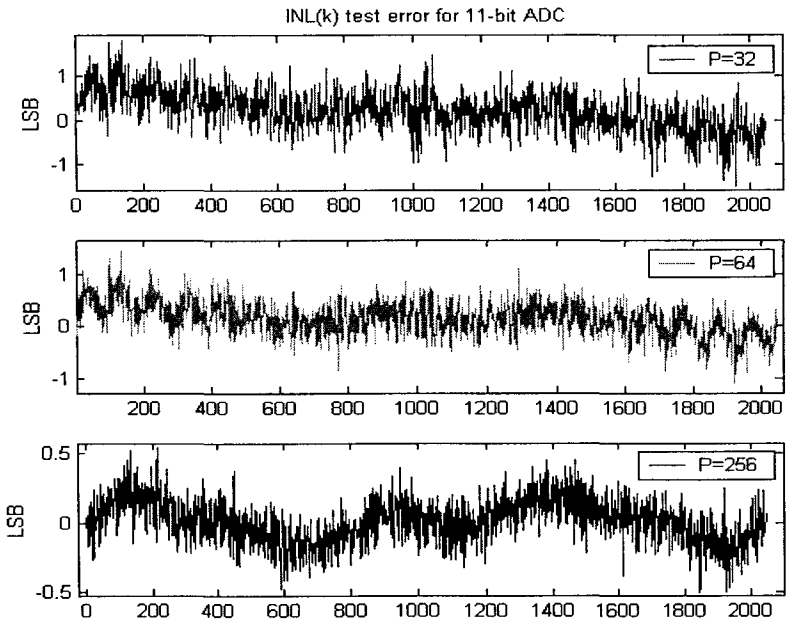


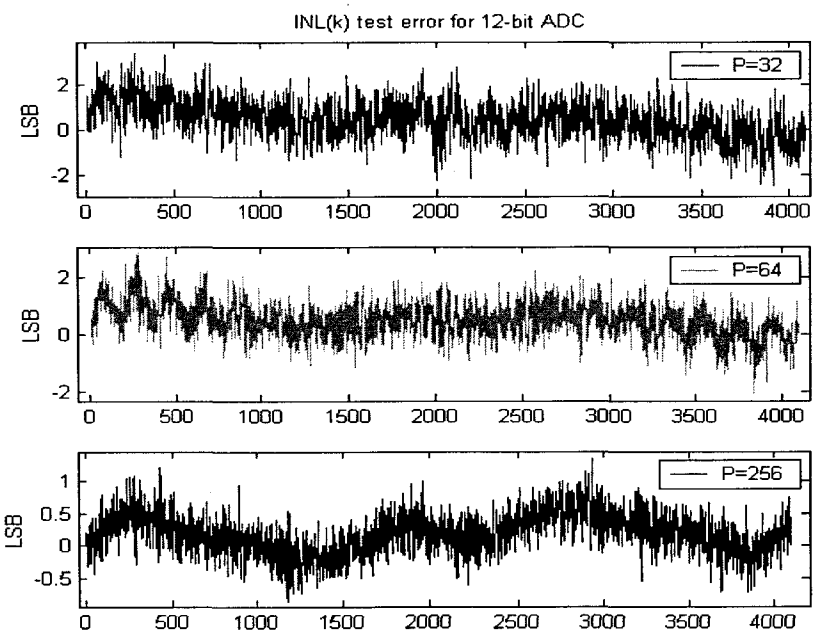
Figure 5.4 INL[k] of the Original 8-bit DAC (Experimental)

The measured and stored DAC output was used as the stimulus to simulated ADCs, and the ADCs' INL[k] are estimated based on its output histogram. The simulated ADC's true INL[k] are known by calculation. We calculated the difference between the estimated INL[k] and the true INL[k]. The difference is the test error with DDEM DAC output as the stimulus to the ADC under test.

To compensate for the systematic error, we first estimated the error by testing one randomly selected DAC. Based on the estimated nonlinear error, we obtained the compensation values for testing ADC. The compensation values were then applied to other DACs. Experimental results show that the compensation values work for all other tested DDEM DACs.



(a) INL[*k*] test error for the 11-bit ADC



(b) INL[*k*] test error for the 12-bit ADC

Figure 5.5 ADC INL[*k*] test error with varying *p* (experimental)

Figure 5.5 (a) and (b) show the test errors after systematic error compensation for an 11-bit ADC and a 12-bit ADC using a DDEM DAC with p=32, 64, and 256, respectively. The simulated ADCs' INL ranges from 5 to 10 LSB. From Figure 5.5, we can see that the test error decreases dramatically when p increases, and that when p = 256 (full-p DDEM), the 8-bit DAC can test the 11- and 12-bit ADCs with errors bounded by about ±0.5 LSB and ±1 LSB, respectively, using DDEM. (For the 12-bit ADC, the output histogram has around 16 hits for each bin.) Note that such a test performance is achievable only by using a DAC at least 13~14-bit linear if DDEM is not used, but in this case, the original DAC's linearity is less than 5 bits without DDEM. It should be also noted that such performance is already better than what can be achieved with the best on-chip linear ramp generator for ADC self-test in literature. [4]

5.3 Second DDEM DAC Design

The DDEM DAC presented above is an 8-bit one. The equivalent linearity after systematic error compensation is about 12 bits, which meets the requirement to test ADCs with resolutions lower than 10 bits if the test error is expected to be lower than ¼ ADC LSB. This performance is comparable to any previously reported on-chip stimulus source for ADC test. However, the test capability is limited by the DAC resolution; also, the systematic error compensation brings an inconvenience to the real application of this DAC.

In the new design, a DDEM DAC was designed with the target of testing 14-bit ADCs with test error lower than ¼ ADC LSB, which means the equivalent linearity should be about 16 bits based on the design approach presented in Section 4.2. The chip area is limited to 2mm^2 in 0.5-um CMOS technology, and the systematic error compensation should be eliminated in

this design. From the discussion in Section 4.2, the implemented DDEM DAC should have 12-bit resolution with the maximum available iteration number limited to 512. The simple, but effective, layout strategy presented in Section 4.2 will be adopted.

As a benefit of the DDEM approach, we don't need to spend much time on the current element matching issue. The DDEM DAC can be viewed as an all "digital" DAC, and we can design the DDEM DAC in a "digital" way, which leads to usage of minimum-sized devices as well as a quick and simple design. The details are presented as follows.

1) DDEM DAC Structure

The target of this design is to implement a DDEM DAC capable of testing 14-bit ADCs with the test error bounded by ¼ LSB at 14-bit level. Guided by rigorous analysis of the DDEM DAC, the apparent resolution and DDEM iteration number of the DDEM DAC are chosen to be 12 bits and 512 (9-bit DDEM control), respectively. All the 4096 current elements are divided into groups of 8, with each group controlled by one DDEM control unit, as shown in Figure 5.6.

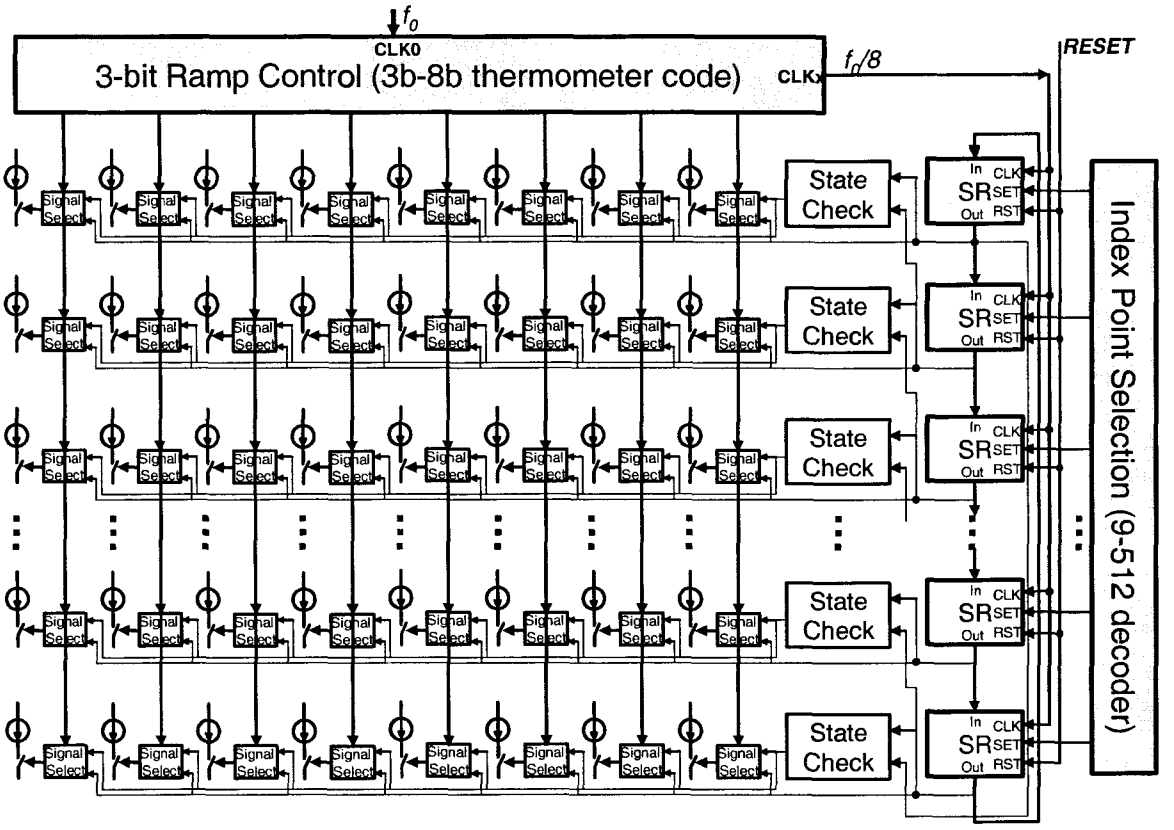


Figure 5.6 Second DDEM DAC structure

Figure 5.7 shows the detail of the DDEM control circuitry. The DDEM control contains a 512-unit shift register (SR) ring, with each unit controlling one current element group. Starting from the all-zero state, one of the SR units is selected as the index point by the decoders shown in Figure 5.6, and a logic '1' is continuously pumped into this unit. Then, each time the DDEM control clock *CLKx* advances, one more SR unit is set to '1'. Thus, the DAC outputs a monotonic 512-step ramp if each group is totally controlled by the associated SR unit. In reality, with the aid of control signal selection logic, finer voltage steps are obtained by using a 3-bit ramp control to increasingly turn on the 8 current elements inside each group during the first *CLKx* cycle with its SR unit set to high. 512 ramps are obtained

by changing the index point with the decoders. The operation here is equivalent to that described in Figure 2.5.

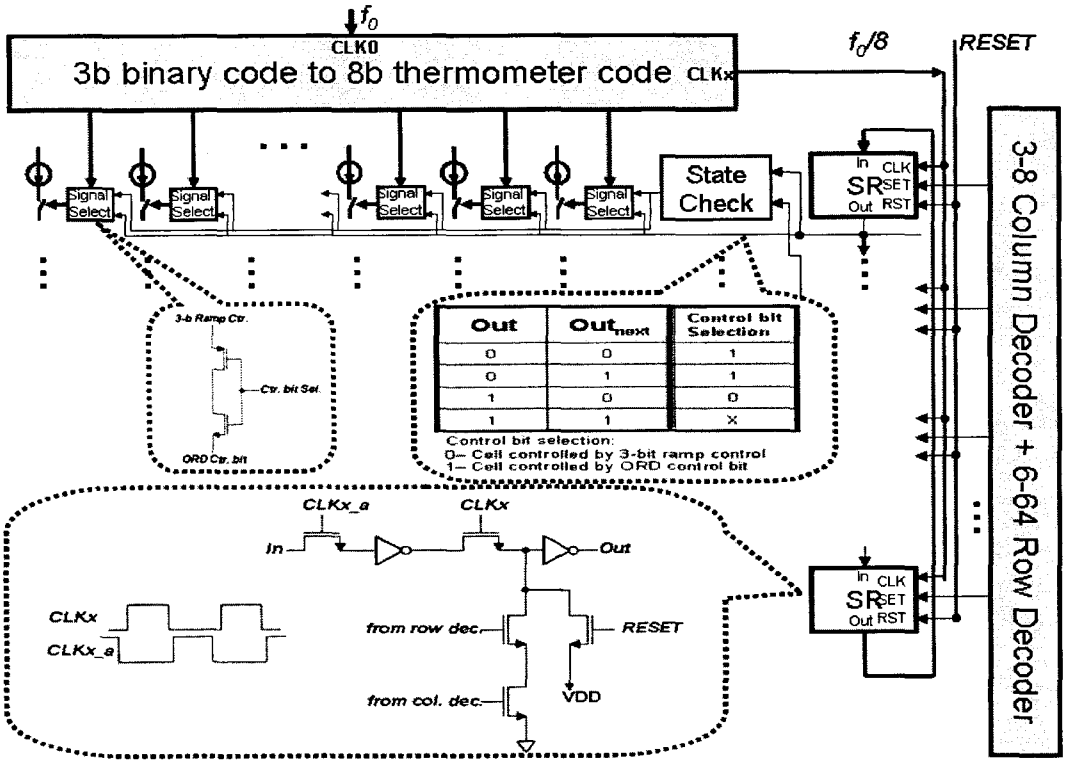


Figure 5.7 Detail of DDEM DAC structure

2) Current Source Element

As a benefit of the DDEM technique, careful matching of current source elements is not necessary. Minimum-sized current elements can be used such that the DAC can be designed in a digital manner. However, it is critical to minimize the systematic output errors, while the DDEM technique takes care of the random errors. Two methods help reduce the systematic errors: 1) differential output pair to cancel even-order errors, 2) inversed output sequence to cancel even-symmetrical, nonlinear errors. Following these guides, a totally symmetrical, current source cell structure is adopted in this design, as shown in Figure 5.8. Three PMOS

transistors, T_1-T_3, form the basic current source structure: T_1 is the current device with its gate connected to a biasing voltage *Vr*, and T_2 and T_3 serve as a switch pair. The 4 reference switching transistors, T_4-T_7, are also shown in Figure 5.8. When the control bit is high, T_2's gate is connected to *Vh*, and T_3's gate is connected to *Vl*. Since *Vh* is set to be higher than *Vl*, when the control bit is high, the current from the drain of T_1 will go through T_3 to the Iop node. Similarly, when the control bit is low, T_1's output current will go through T_2 to the Ion node. This way, the current source functions symmetrically when the control bit signal is set between high and low. Two external resistors, *Rp* and *Rn*, are connected to *Iop* and *Ion*, respectively. The current from Iop/Ion will be collected on *Rp*/*Rn*, and the voltage difference across *Rp* and *Rn* serves as the output voltage.

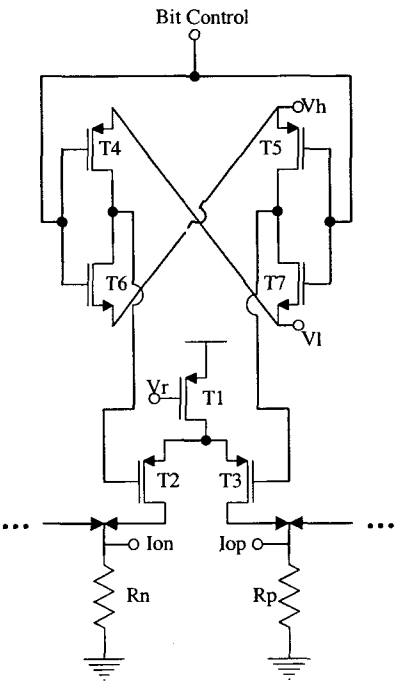


Figure 5.8 Improved current steering element structure

Simulation shows that the settling error of the current source in Figure 5.8 is less than 0.02%, within 5ns. Actually, the settling errors can also be handled by the DDEM algorithm. Hence,

the DDEM DAC can be operated at a very high speed (up to hundreds of MHz) as a result of the simple structure and small device size. Still, to maximize the speed, the two reference voltages *Vh* and *Vl* should be set properly. The difference between *Vh* and *Vl* should be chosen carefully such that when the control bit signal changes, the current from T_1 can be almost totally switched to either T_2 or T_3, while during switching all three transistors T_1-T_3 will stay in the saturation region. The appropriate values for *Vr*, *Vh*, and *Vl* can be found through transistor-level simulation.

In addition, to make the output nonlinear error sequence symmetrical (note that the inversed output sequence cannot cancel non-symmetrical nonlinear errors), a star-shaped power supply routing, as shown in Figure 5.9, is adopted such that the distances from all the current elements to the VDD pad are roughly the same.

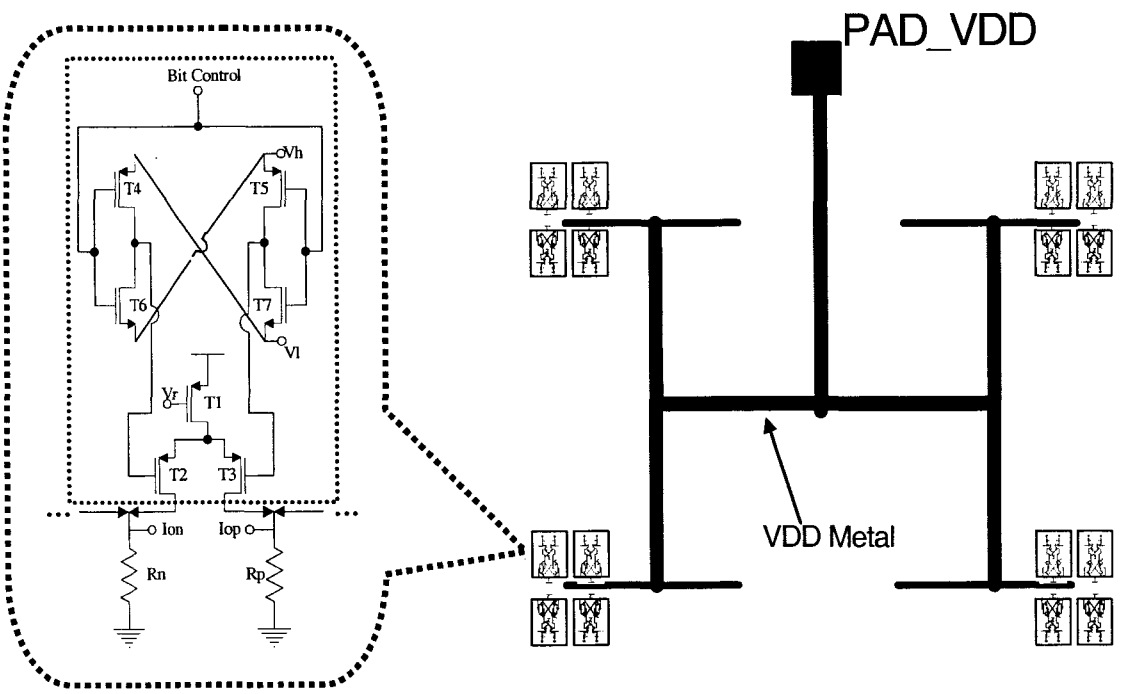


Figure 5.9 Star VDD of current source element array

3) Layout

In the first DDEM DAC design, a spiral-shaped layout was used to achieve a good "averaged" DAC. However, this layout strategy does not bring any benefit to the output distribution of the DDEM DAC. It has been shown that this layout scheme is vulnerable to gradient errors and that systematic error compensation is required after fabrication.

In the second design, the simple but effective layout strategy introduced in Section 4.2 is adopted. All the 4096 current source elements are divided into 512 groups that are placed in 8 columns with 64 groups in each column. All 512 groups are numbered following the strategy stated in Section 4.2, as shown in Figure 4.6.

5.4 Second DDEM DAC Experimental Results

The 12-bit DDEM DAC was fabricated in the same 0.50-μm standard CMOS process as the first one. The core die size is 1.5mmmm×1.4mm=2.1mm^2. The die photo is shown in Figure 5.10. The power supply voltages are 5V for both digital and analog parts. When driving the 22-ohm resistance loads, the differential output range is -1.1~1.1 volts. The actual output range can be tuned by changing the resistance loads or the biasing voltage.

The DDEM DAC was tested on a Credence Electra IMS (Integrated Measurement System) tester. The tester provides the power supply, biasing, reference voltages, RESET signal, two-phase non-overlapping clocks, and 9-bit DDEM iteration control signals. The output voltage across *Rp* and *Rn* is sampled using an 18-bit digitizer. When the DDEM DAC was clocked at 100MHz, neat ramps can be observed from the oscilloscope. However, the DDEM DAC output was only measured using 1MHz clocks due to the speed limitation of the high-

resolution digitizer. The DDEM DAC's output samples under the total 512 iteration control codes were measured.

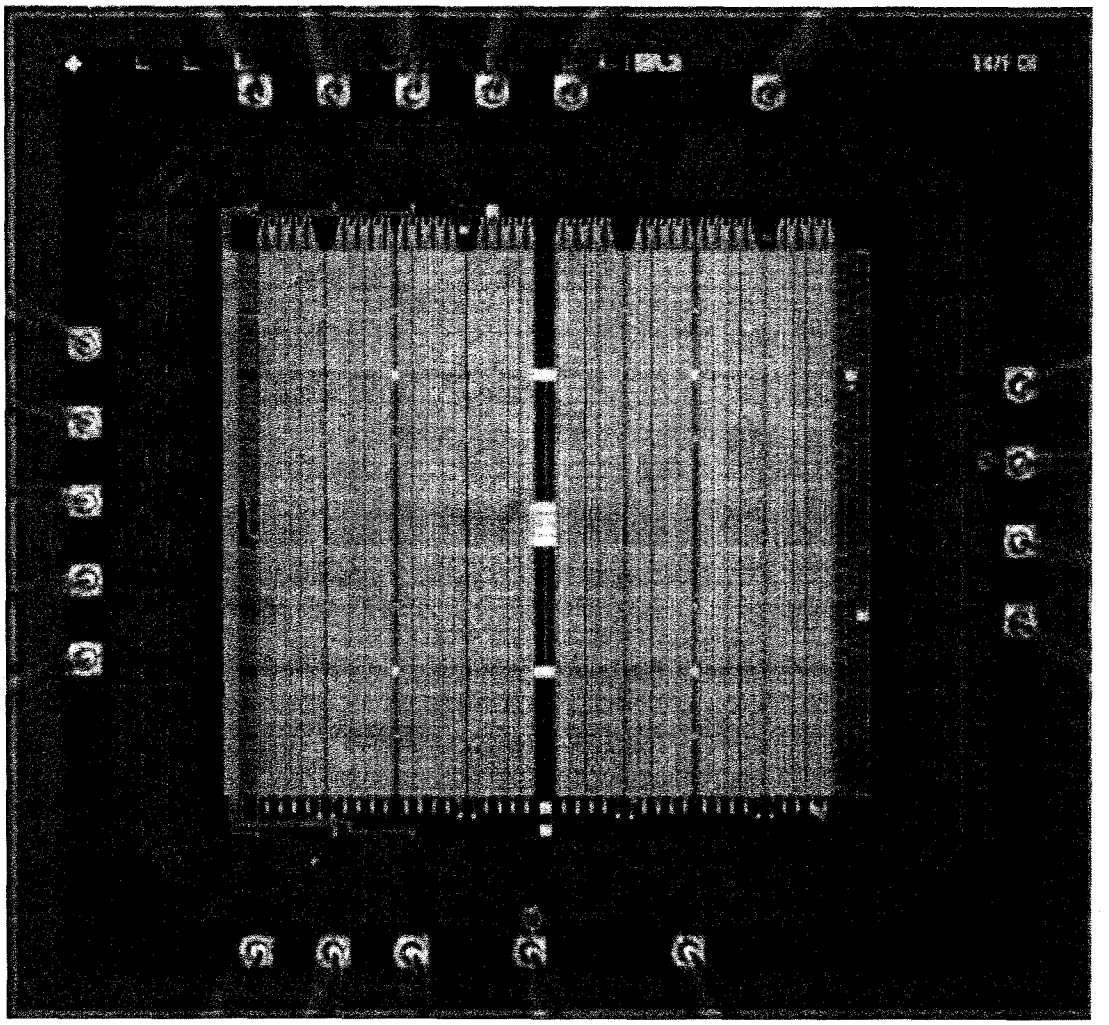


Figure 5.10 Die photo of the second DDEM DAC

Figure 5.11 shows that without DDEM, the original 12-bit DAC has an INL error of 2.3 LSBs, which means that the original DAC is about 10-bit linear. The transition point p_T in (4.8) is about 64. By (4.8), the DDEM DAC is expected to have an equivalent linearity of about 15.5 bits when p=512 by the following calculation

$$n_{eq} \approx ENOB_{DAC} + \log_2 p_T - 1 + 0.5\log_2(p/p_T) = 9 + 6 - 1 + 1.5 = 15.5$$

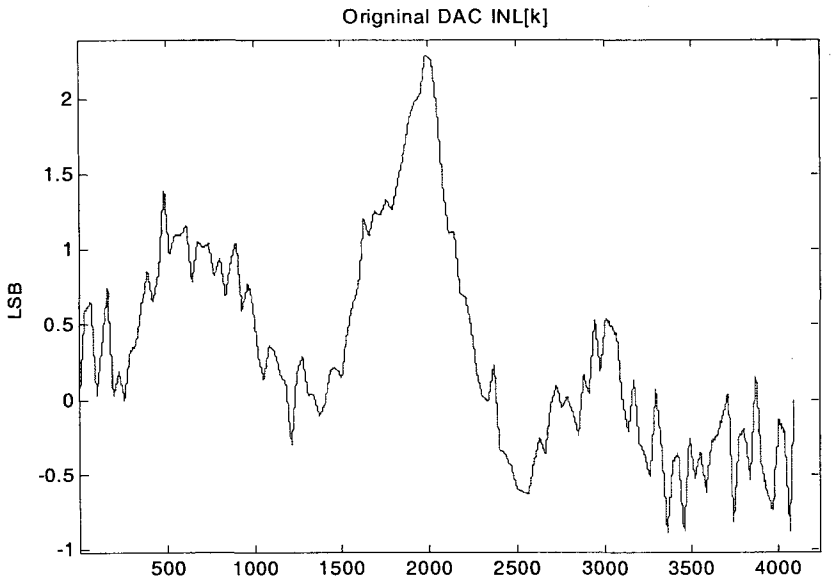


Figure 5.11 INL[*k*] of the original 12-bit DAC (experimental)

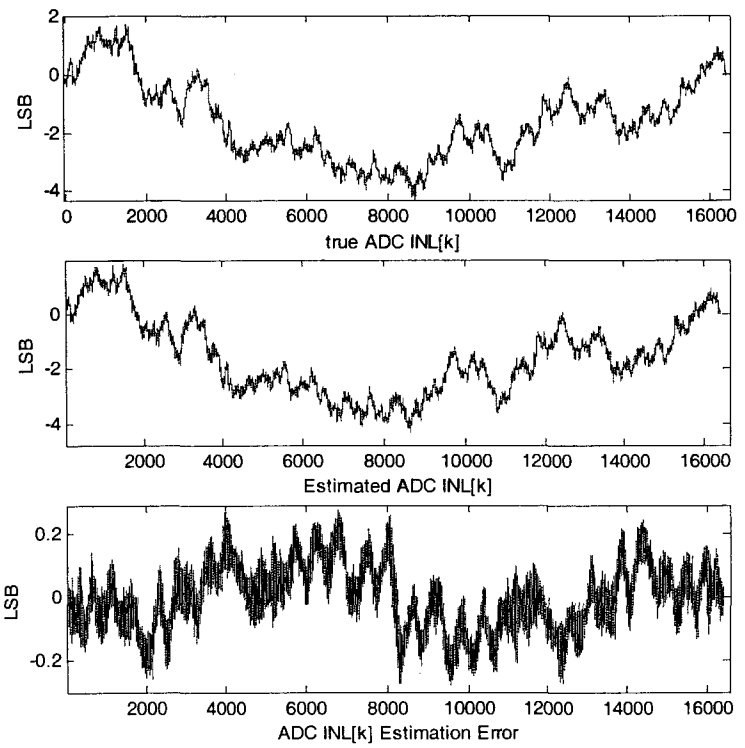


Figure 5.12 ADC INL[*k*] test curves using 12-bit DDEM DAC (experimental)

To evaluate the DDEM DAC performance, the measured DDEM DAC output samples were used as the stimulus source to test simulated 14-bit ADCs. As done previously, the estimated INL[k] curve using the DDEM DAC was compared to the ADC true INL[k] curve, and the difference is the ADC test error. The result for one ADC test is shown in Figure 5.12. The INL[k] estimation error is bounded by ±0.3 ADC LSB, which means that the DDEM DAC has an equivalent linearity of about 16 bits ($14-\log_2 0.3=15.7$). Note the previous calculation gives $n_{eq}=15.5$. The actual test performance matches the theoretical prediction by (4.8).

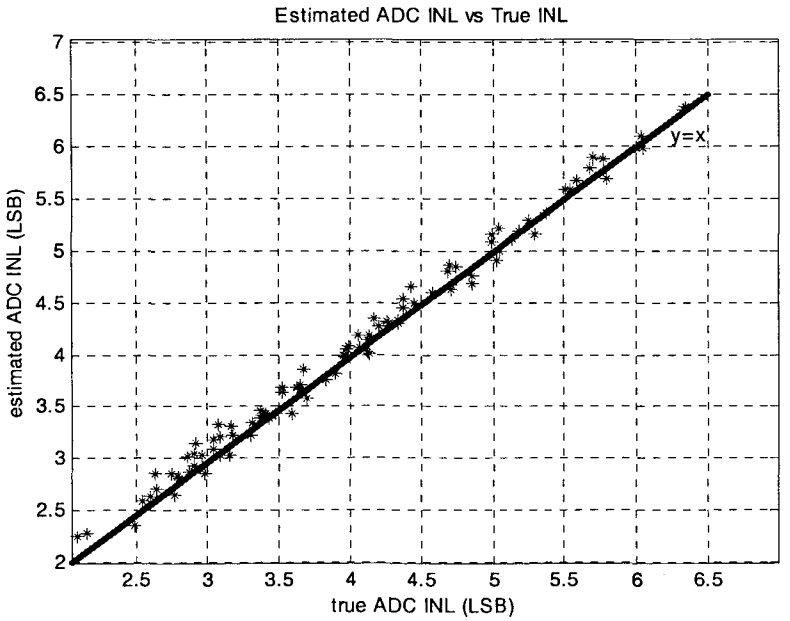


Figure 5.13 ADCs' true INL vs. estimated INL (experimental)

To verify the robustness of the DDEM DAC performance, the measured DDEM DAC was used to test 100 simulated 14-bit ADCs with different amounts of INL errors. The estimated ADC INLs, using the DDEM DAC as test stimulus source versus true ADC INLs, are shown in Figure 5.13. The ADC INL estimation errors (defined as the estimated INL minus the true

INL) range from –0.16 to 0.24 ADC LSB, all within ¼ LSB at the 14 bit level for 100 parts. The error amount indicates that the DDEM DAC has achieved 16-bit performance in ADC linearity testing and the performance is robust.

It should be emphasized that the 16-bit linearity performance was achieved from an original 10-bit linear DAC. Table 1 provides a comparison between this work and other on-chip stimulus sources for ADC test in the literature. Only those having experimental results are listed in this table. As can been seen, this new DDEM DAC outperforms any previously reported on-chip stimulus source by 5 bits. Since this DDEM DAC design is a "digital" DAC, it can be scaled down easily for newer technologies. Compared to other source generators, this new DDEM DAC is very die area efficient.

Table 5.1 Performance comparison

Source Generator	Year	Source Type	Die Area & Technology	Performance
2nd DDEM DAC	2005	DDEM DAC	2.1mm^2@ 0.5μm CMOS	16 bits
1st DDEM DAC [10]	2004	DDEM DAC	0.4mm^2 @ 0.5μm CMOS	12 bits
B. Provost et. al. [4]	2003	Linear Ramp	0.18mm^2@ 0.18μm CMOS	11 bits
C. Jansson et. al. [14]	1994	Linear Ramp	N/A @ 2μm CMOS	8 bits

Chapter 6 Segmented DDEM

The basic DDEM technique is applied on thermometer-coded (TC) current steering DACs. However, it will be difficult to implement TC DACs when the resolution goes high. To overcome this problem, two techniques based on the basic DDEM technique are proposed as solutions. With these techniques, the ADC BIST cost can be further reduced by using simpler circuit structures. In this chapter, the Segmented DDEM approach will be presented. [11]

6.1 Method Description

For an n bit current steering DAC, we can divide the n bits to two parts: $n= n_M+n_L$, in which n_M represents the more significant bits and n_L the less significant bits. If we let $N_M = 2^{n_M}$ and $N_L = 2^{n_L}$, we have $N = 2^n = N_M \cdot N_L$. For a DAC input code k, we can break it up as follows:

$$k = k_M N_L + k_L \quad \begin{pmatrix} 0 \le k \le N-1, \\ 0 \le k_M \le N_M - 1, \\ 0 \le k_L \le N_L - 1 \end{pmatrix} \tag{6.1}$$

To get the analog signal corresponding to k, we can obtain the analog signals corresponding to k_M and k_L with different respective weights first and then combine them together. To implement this, we can use a MSB current source array to generate k_M and a LSB current source array to generate k_L. Here, the MSB and LSB array have N_M-1 and N_L-1 current source elements, respectively, and the weight of each MSB array element is N_L times that of a LSB array element. This is termed the segmented-thermometer-coded (STC) DAC structure. A 4-

bit STC current steering DAC is shown in Figure 6.1 as an example. In this example, n=4, n_M=n_L=2 and N_M=N_L=4.

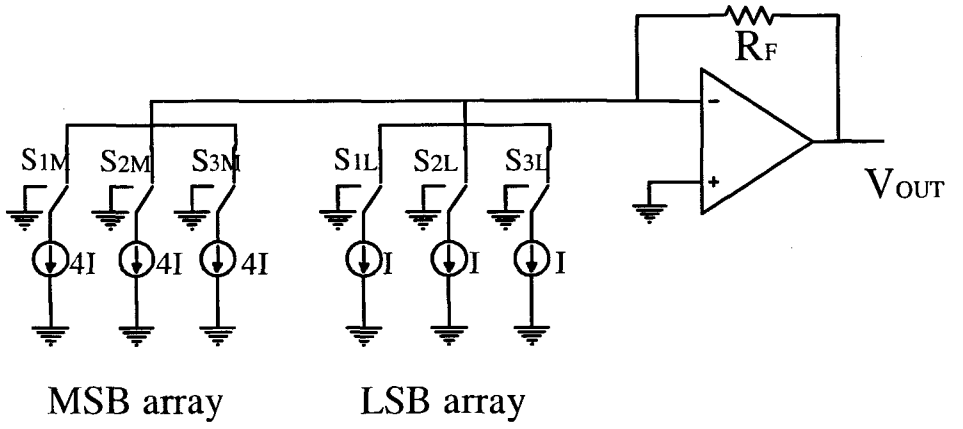
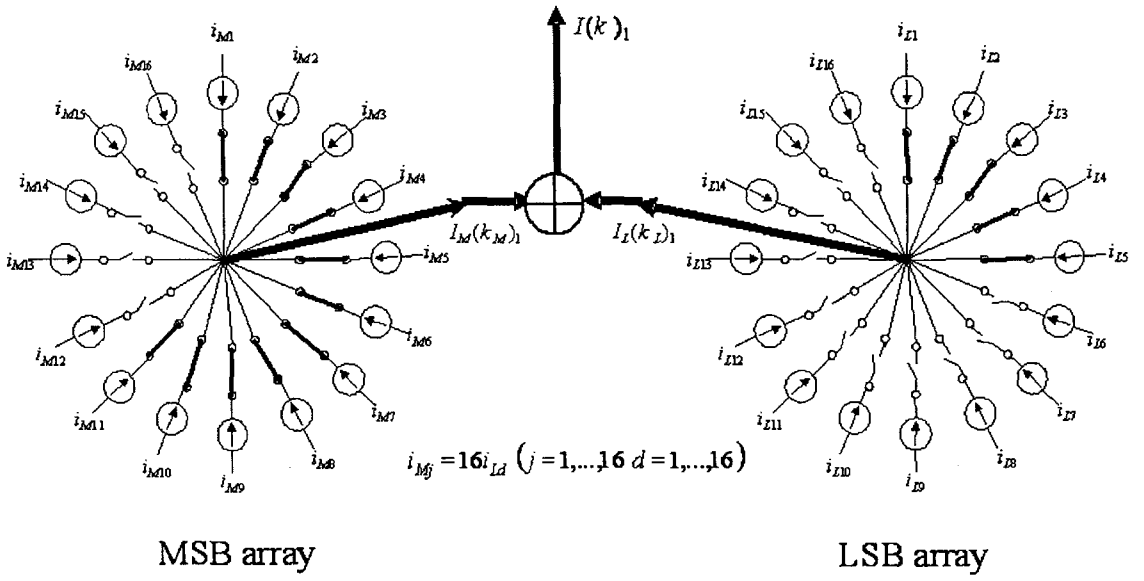
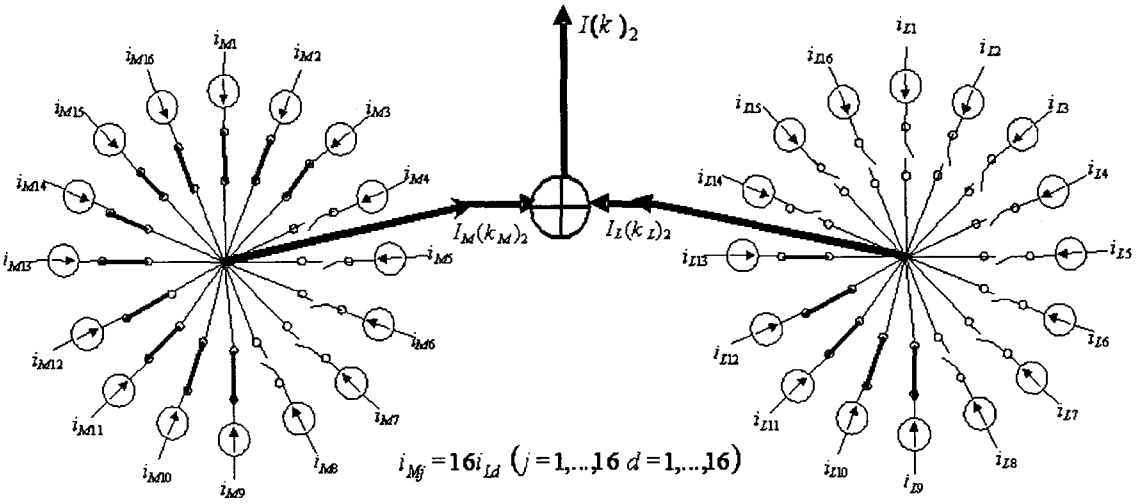


Figure 6.1 4-bit segment coded current steering DAC structure

To implement the DDEM for a STC DAC, we only need to apply DDEM to both the MSB array and the LSB array simultaneously. We add one extra current source element for both the MSB and LSB array; then, the MSB array has N_M current source elements, and the LSB array has N_L currents source elements. Suppose now that the DAC input code is $k = k_M N_L + k_L$ and that each code needs to have p output samples. To generate each output sample for a code k, the DDEM method picks k_M current sources from the MSB array and k_L current sources from the LSB array by applying DDEM switching scheme to the MSB and LSB arrays, respectively. Figure 6.2 illustrates the current source switching scheme for 8-bit STC DAC. In this example, we have n_L=n_M=4 and N_L=N_M=16. For each input code k, 2 samples are output. In Figure 6.2, k=191=11×N_L+5; hence, k_M=11 and k_L=5. For the first output sample, i_{M1}~i_{M11} are selected from the MSB array, and i_{L1}~i_{L5} are selected from the LSB array; for the second output sample, i_{M9}~i_{M16} and i_{M1}~i_{M3} are selected from the MSB array, and i_{L9}~i_{L13} are selected from the LSB array.



(a) 1st output when k=191



(b) 2nd output when k=191

Figure 6.2 DDEM switching of an 8-bit STC DAC

We use $i_{M,j}\ \ (j=1,...,N_M)$ to represent the j^{th} current source element out of the total N_M elements of the MSB array and $i_{L,j}\ \ (j=1,...,N_L)$ to represent the j^{th} current source element out of the total N_L elements of the LSB array. Each DAC input code k has p output samples.

Let's define $q_M = N_M / p$ and $q_L = N_L / p$. Each output for a given k is the summation of the selected k_M MSB current elements and k_L LSB current elements. The d^{th} current summation is denoted by $I_d[k]$. We have:

$$I_d[k] = \sum_{j=1}^{k_M} i_{M,(d-1)q_M+j} + \sum_{j=1}^{k_L} i_{L,(d-1)q_L+j} \quad d = 1,\ldots,p \tag{6.2}$$

The average of p samples is denoted by $\bar{I}[k]$.

$$\bar{I}[k] = \frac{1}{p}\sum_{d=1}^{p}\left(\sum_{j=1}^{k_M} i_{M,(d-1)q_M+j} + \sum_{j=1}^{k_L} i_{L,(d-1)q_L+j}\right) \tag{6.3}$$

The static performance of the segment-coded DAC with DDEM will be evaluated based on $I_d[k]$ and $\bar{I}[k]$, as we did to the basic DDEM DAC.

6.2 Performance Evaluation

In this section, we show that the "averaged DAC" of a STC DAC using DDEM approach has a DC transfer curve that approaches a straight line, or, equivalently, we show that the INL of the averaged DAC is very small. We also show that the performance degradation due to the segmented structure is limited to an acceptable range.

For the STC DAC, we suppose the desired value for all the MSB array elements is i_{M0}, while the desired value for all the LSB array elements is i_{L0}. Due to process and other variations, the actual value of each current source is given by:

$$\begin{aligned} i_{M,j} &= i_{M0}(1+\varepsilon_{M,j}) \quad (j = 1,\ldots,N_M) \\ i_{L,j} &= i_{L0}(1+\varepsilon_{L,j}) \quad (j = 1,\ldots,N_L) \end{aligned} \tag{6.4}$$

We assume that each MSB array element is just the combination of N_L current source elements that are used to build the TC DAC in Chapter 2 and that the LSB array elements are

identical to the current source elements used in the TC DAC. Then, ideally, $i_{M0}=N_L * i_{L0}$, and then each $\varepsilon_{M,j}\,(j=1,...,N_M)$ is independent and has an identical Gaussian distribution.

$$\varepsilon_{M,j}\ \ i.i.d.\ \sim N\left(0,\frac{1}{N_L}\sigma^2\right) \tag{6.5}$$

The standard deviation of each LSB array element is still σ^2. However, by careful layout design and good matching technique, we can make the relative error between the LSB and MSB array elements quite small. We can then approximate to the following equations:

$$\sum_{j=1}^{N_L} i_{L,j} = \frac{1}{N_M}\sum_{j=1}^{N_M} i_{M,j} \tag{6.6}$$

$$\sum_{j=1}^{N_L} \varepsilon_{L,j} = \frac{N_L}{N_M}\sum_{j=1}^{N_M} \varepsilon_{M,j} \tag{6.7}$$

If we can make better matching, we can even have:

$$\sum_{d=1}^{p} \varepsilon_{L,(d-1)q_L+j} = \frac{N_L}{q_L N_M}\sum_{j=1}^{N_M} \varepsilon_{M,j} \tag{6.8}$$
$$where\ N_L = p\cdot q_L\ \&\ q_L is\ small$$

Equation (6.8) is only valid when the MSB and LSB arrays are well matched. Good matching can be achieved by the post-fabrication calibration affordably. For instance, we can tune the reference voltages for the MSB and LSB arrays accordingly. Although not included in this analysis, a small amount of mismatching error can be tolerated, as shown in the simulation results presented in the next section.

For the DDEM STC DAC, we can use the output current instead of voltage in all our computations, since R_F is a constant factor and can be taken out of the derivations. We define the fit line based on the two code end points: 0 and N. The corresponding output for

the input code 0 for the STC DAC is 0. We assume the output current for input code N is obtained when all the N_M MSB current sources are switched. This output is given by:

$$I[N]=\sum_{j=1}^{N_M} i_{M,j} = N_M i_{M0} + i_{M0}\sum_{j=1}^{N_M}\varepsilon_{M,j} \tag{6.9}$$

Then, the LSB of the DEM segment coded DAC is given by:

$$\overline{LSB}=\frac{I[N]}{N}=\frac{N_M i_{M,0}}{N}+i_{M,0}\frac{1}{N}\sum_{j=1}^{N_M}\varepsilon_{M,j}=i_{L,0}+i_{L,0}\frac{1}{N_M}\sum_{j=1}^{N_M}\varepsilon_{M,j} \tag{6.10}$$

It can be verified that $\frac{1}{N_M}\sum_{j=1}^{N_M}\varepsilon_{M,j}$ is Gaussian and that

$$\frac{1}{N_M}\sum_{j=1}^{N_M}\varepsilon_{M,j} \sim N(0,\frac{1}{N}\sigma^2) \tag{6.11}$$

The fit-line of the DEM DAC is then given by:

$$\bar{I}_{fit}[k]=k\cdot\overline{LSB}=ki_{L0}+i_{L0}\frac{k}{N_M}\sum_{j=1}^{N_M}\varepsilon_{M,j} \tag{6.12}$$

Based on the fit line, we now compute the INL of the DDEM STC DAC. When the input code is $k=k_M N_L+k_L$, in which $\mathrm{k_M}=\mathrm{t_M q_M}+\mathrm{s_M}$ and $\mathrm{k_L}=\mathrm{t_L q_L}+\mathrm{s_L}$, the averaged DAC output current for code k is given by:

$$\begin{aligned}\bar{I}[k]=k\cdot i_{L0}+i_{M0}\cdot\frac{1}{p}\cdot\left(t_M\cdot\sum_{j=1}^{N_M}\varepsilon_{M,j}+\cdot\sum_{d=1}^{P}\sum_{j=1}^{s_M}\varepsilon_{M,(d-1)q_M+j}\right)+\\ +i_{L0}\cdot\frac{1}{p}\cdot\left(t_L\cdot\sum_{j=1}^{N_L}\varepsilon_{L,j}+\cdot\sum_{d=1}^{P}\sum_{j=1}^{s_L}\varepsilon_{L,(d-1)q_L+j}\right)\end{aligned} \tag{6.13}$$

Subtracting it by the corresponding fit line point yields:

$$\begin{aligned}\overline{INL}[k] &= \bar{I}[k] - \bar{I}_{fit}[k] \\ &= i_{M0}\cdot\frac{1}{P}\cdot\left(t_M\cdot\sum_{j=1}^{N_M}\varepsilon_{M,j}+\cdot\sum_{d=1}^{P}\sum_{j=1}^{s_M}\varepsilon_{M,(d-1)q_M+j}\right)+ \\ &\quad + i_{L0}\cdot\frac{1}{P}\cdot\left(t_L\cdot\sum_{j=1}^{N_L}\varepsilon_{L,j}+\cdot\sum_{d=1}^{P}\sum_{j=1}^{s_L}\varepsilon_{L,(d-1)q_L+j}\right)-i_{L0}\frac{k}{N_M}\sum_{j=1}^{N_M}\varepsilon_{M,j} \\ &= i_{L0}N_L\left[\left(\frac{1}{P}-\frac{s_M}{N_M}\right)\sum_{d=1}^{P}\sum_{j=1}^{s_M}\varepsilon_{M,(d-1)q_M+j}-\frac{s_M}{N_M}\sum_{d=1}^{P}\sum_{j=s_M+1}^{q_M}\varepsilon_{M,(d-1)q_M+j}\right] \\ &\quad + i_{L0}\left[\frac{1}{P}\cdot\left(t_L\cdot\sum_{j=1}^{N_L}\varepsilon_{L,j}+\cdot\sum_{d=1}^{P}\sum_{j=1}^{s_L}\varepsilon_{L,(d-1)q_L+j}\right)-\frac{k_L}{N_M}\sum_{j=1}^{N_M}\varepsilon_{M,j}\right]\end{aligned} \tag{6.14}$$

If we apply (6.7) and (6.8) to (6.14), the second part of (6.14) is equal to 0.

The distribution of the normalized $\overline{INL}[k]$ is then given by:

$$\frac{\overline{INL}\ [k]}{i_{L0}} \sim N\left(0, A\sigma^2\right)$$

$$\textit{where} \qquad A = N_L^2\left[\left(\frac{1}{P}-\frac{S_M}{N_M}\right)^2 Ps_M\frac{1}{N_L}+\left(\frac{S_M}{N_M}\right)^2 P(q_M - s_M)\frac{1}{N_L}\right] = \tag{6.15}$$

$$= N_L\frac{s_M(q_M - s_M)^2}{Pq_M}$$

Note that (6.15) is valid only when the MSB and LSB arrays are well matched. The variance of $\overline{INL}[k]$ may be larger than expressed in (6.14) due to the mismatch.

From (6.15), we can also estimate where and how much we expect the maximum deviation from the fit line to be. The variance of $\overline{INL}[k]$ reaches its maximum value at $s_M = \frac{1}{2}q_M$. Using this value for s_M and i_{L0} as 1 LSB, the largest standard deviation of $\overline{INL}(k)$ is:

$$\sqrt{N_L\frac{q_M}{4P}}\sigma = \sqrt{\frac{N}{4P^2}}\sigma \tag{6.16}$$

Comparing with the result obtained for a TC DAC in Chapter 2, we find that the result obtained there agrees with (6.16). Thus, if the MSB and LSB arrays match, the STC structure

achieves the same performance as the TC structure. The benefit is that the circuit complexity is greatly reduced, since fewer current sources and switches are required to implement DDEM. For example, an 18-bit TC DAC requires more than 250,000 current sources, switches, and switch control units, while an 18-bit STC DAC with $N_M = N_L$ needs only 1024. The switching complexity is also reduced considerably by having only to control 512 switches on each array rather than 250,000. To ensure LSB and MSB matching, a calibration scheme can be employed to adjust the LSB array values by tuning the biasing voltages before the DAC is used.

6.3 Simulation Results

For the following simulation results, the ADCs have 16-bit resolution while the STC DACs have 18-bit resolution. The simulated ADCs have various INL errors. The original DACs have an ENOB much less than 18 bits, since there are large mismatching errors for the current elements inside the MSB and LSB arrays. A 1% mismatching error between the LSB and the MSB arrays was included. For a real segmented DAC, this matching error can be easily tuned below the 1% level with the post-fabrication calibration. To simulate the actual test environment, noise was also added to the DAC output; this noise could be as big as ± 3 LSB_{DAC}.

Figure 6.3 shows the INL distribution of the 1000 simulated ADCs. A STC DAC with an original INL equal to 38 LSB is simulated, which means that the actual DAC linearity is less than 12 bits. The DDEM approach is then applied to this DAC and the outputs are sent to the ADCs. We estimate the INL for each ADC based on the histogram with these inputs and calculate how much it deviates from the true ADC INL. The results are shown in Figure 6.4.

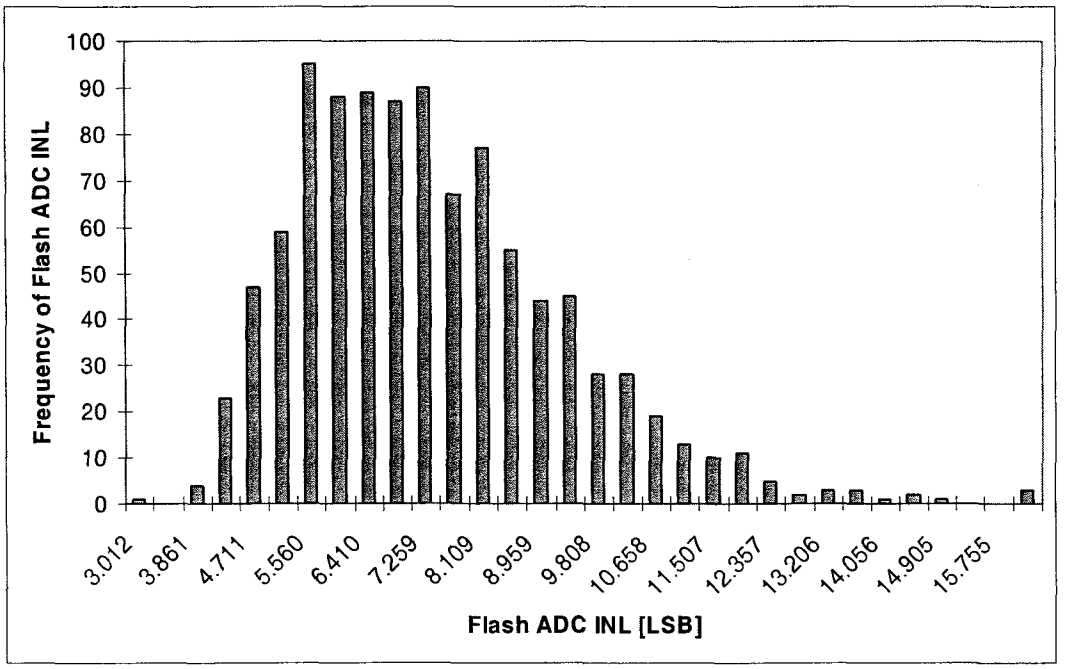


Figure 6.3 INL distribution for 1000 flash ADCs

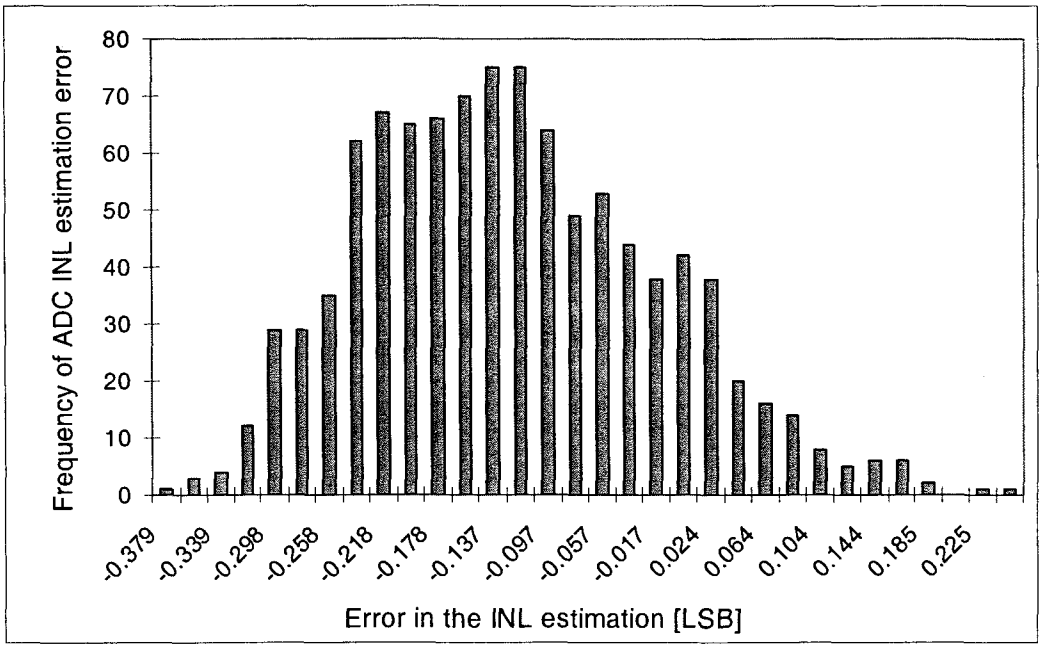


Figure 6.4 INL estimation error distribution using a DDEM STC DAC

Using the DDEM STC DAC, the error in the INL estimation is between -0.39 and 0.2 LSB. The performance degradation compared to the TC DDEM DAC may be attributable to LSB

and MSB matching errors. The estimation degradation is only a factor of 2, while the area and complexity was significantly reduced.

The resultant structure is suitable for BIST applications. In that case, each ADC has a particular DAC to test it. Then, 1000 pairs DAC-ADC are simulated. The DACs have the same errors as before, while the ADCs used have an INL distribution, as shown in Figure 6.5. As can be seen, the ADCs to be tested are actually 16-bit linear, since their INL is not bigger than ½ LSB in most of the cases. The DACs used for the testing have linearity of 12 to 13 bits without DDEM, which is actually 3 bits less than the linearity of the DUT.

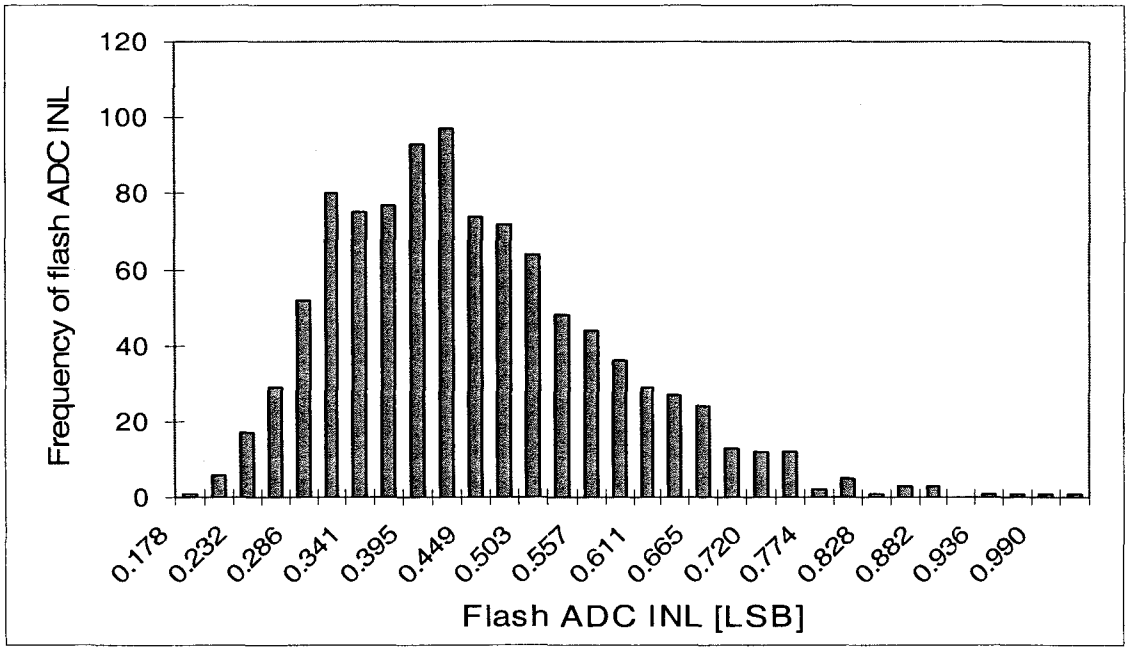


Figure 6.5 INL distribution for 1000 accurate ADCs

Figure 6.6 shows the simulation results for the 1000 DAC-ADC pairs. We can see that the estimated INL has an error of less than ½ LSB for a majority of the cases. This test verifies that the technique is suitable for BIST applications.

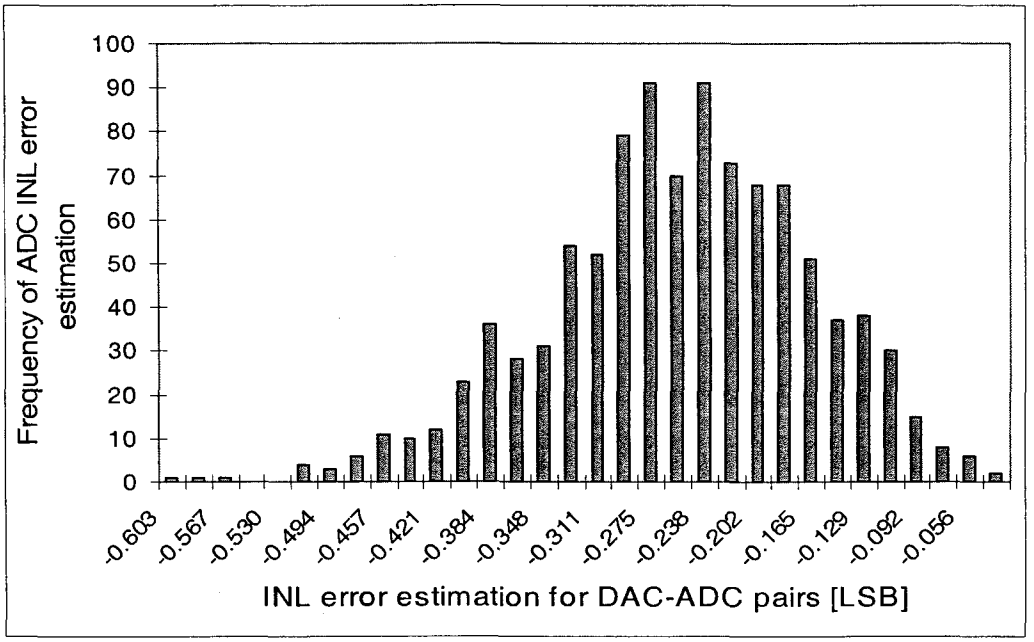


Figure 6.6 INL error estimation for DDEM STC DAC and ADCs 1000 pairs

To verify the robustness of the proposed Segmented DDEM technique as a test tool, we simulated 1000 ADCs with INL around 0.5 LSB. The DAC used to estimate the ADCs' INLs has an INL equal to 38 LSB and p= 128. Assume that the ADCs need to have less than 0.8 LSB INL to comply with their specifications, so the testing boundary to say that a part is a good one is below 0.8 LSB. The parts that have INL between 0.8 and 2 LSB are classified as "not so good" (NSG) parts and can be still marketed as less accurate parts. We can see in Figure 6.7 that although some good parts are tested as NSG ones, there are no NSG parts classified as good ones, which means that the customer will not receive a deficient part.

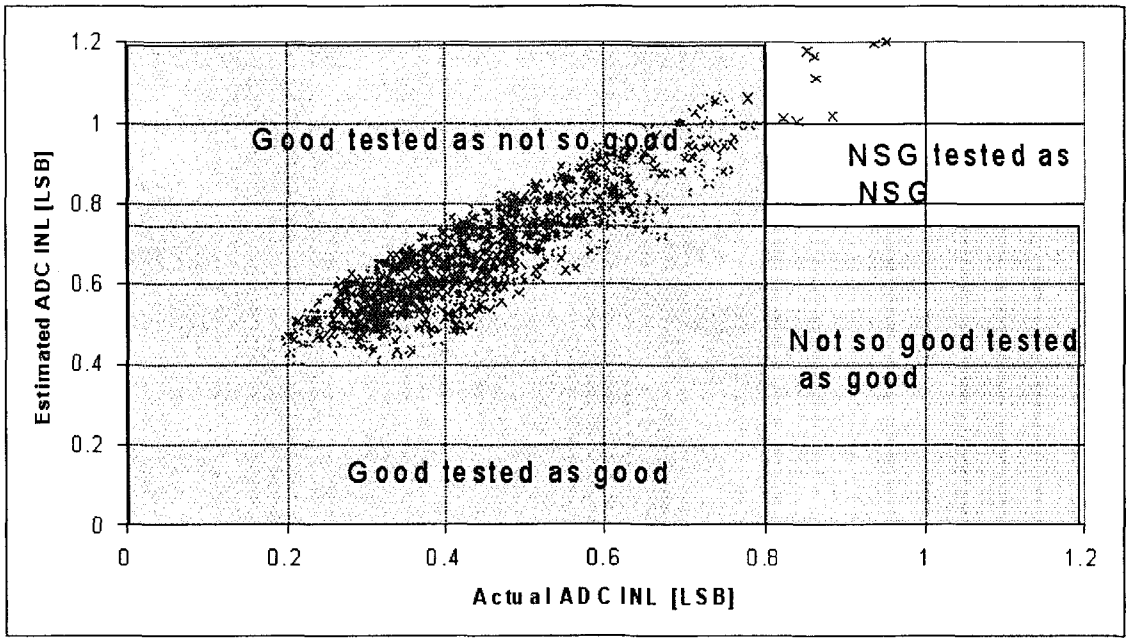


Figure 6.7 DDEM STC DAC used for 1000 accurate flash ADCs

Now let's summarize Chapter 6. In this chapter, the Segmented DDEM approach is introduced. Both theoretical analysis and simulations were used to validate the new DDEM architecture for testing ADCs. The architecture is suitable for BIST applications because it requires less area and uses a simpler switching scheme. The performance of the new DDEM STC DAC is similar to the simple DDEM DAC presented in previous chapters, while the hardware requirement is greatly reduced.

Chapter 7 Dither Incorporated DDEM

The Dither Incorporated DDEM (DiDDEM) is another approach based on the basic DDEM technique by which simpler circuit structure than that used in the basic DDEM approach can be adopted as ADC BIST stimulus sources.

7.1 Method Description

Similar to the DEM technique, dither is also a technique widely used in ADC design and application to improve the output performance statistically. [33] In this work, we adopt the dither technique in the test application as the DDEM case.

Before introducing the new approach, the DDEM DAC will be reviewed briefly in this section. An n-bit DDEM DAC has a total of 2^n current sources. Let $N=2^n$. Use $i_j \quad (j=1,...,N)$ to represent the j^{th} current source element out of the total N elements. Let p denote the DDEM iteration number. p represents the number of samples to be generated for each DAC input word k. Define $q=N/p$. The Cyclic DDEM switching scheme is applied to the current elements, as presented in Chapter 2.

According to theoretical analysis, if a DAC has an effective number of bits (ENOB) of n_{ENOB}, then with the DDEM approach, the DAC can achieve test performance comparable to a linear DAC, with $n_{ENOB}+\log_2 p$ bits resolution, when p is small, as shown in (4.8). Equation (4.8) also shows that when p is larger than a transition value p_T, the test performance increases only by a half bit if p doubles.

The DDEM approach greatly relaxes the stimulus requirement for ADC test. With the new technique based on the DDEM approach proposed in this chapter, the stimulus requirement

can be further reduced. In this technique, one low-resolution DDEM DAC and one low-resolution dither DAC are combined to provide the stimulus signals to the ADC under test as shown in Figure 7.1. We term the new approach as the Dither Incorporated DDEM (DiDDEM).

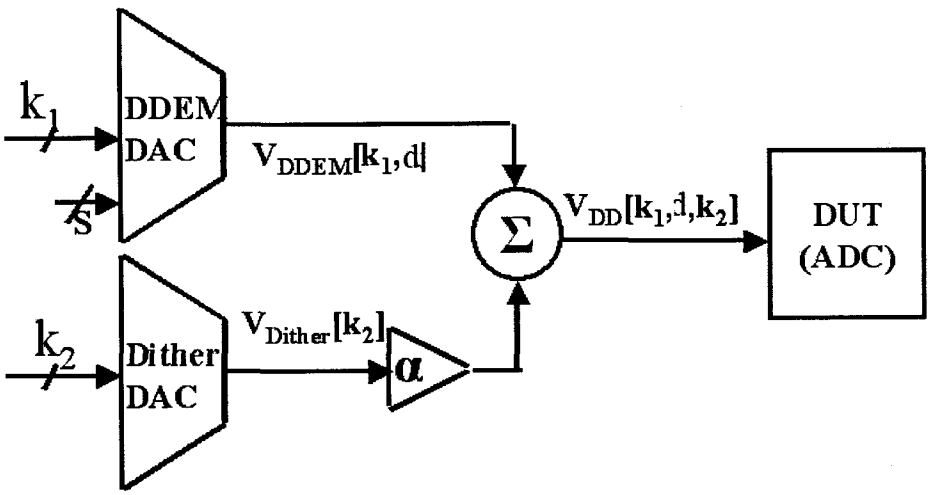


Figure 7.1 DiDDEM DAC for ADC test

Assume we have one DDEM DAC with n_1-bit resolution and one dither DAC with n_2-bit resolution. The DDEM iteration number is p. As we know, the DDEM DAC has two control words, one is the input code (denoted as k_1), and the other is the DDEM iteration code (denoted as d). We denote the output from the DDEM DAC as $V_1[k_1,d]\,(1 \le k_1 \le N_1 \,\&\, 1 \le d \le p, N_1 = 2^{n1})$. Denote the dither input code as k_2, and the output from the dither DAC as $V_2[k_2]$ $(1 \le k_2 \le N_2, N_2 = 2^{n2})$. The dither DAC's output is attenuated by a factor α and then added to the DDEM DAC's output. If the maximum outputs of the two DACs are V_{1m} and V_{2m}, respectively; nominally, α is given by:

$$\alpha = \frac{1}{p}\frac{V_{1m}}{V_{2m}} \tag{7.1}$$

The summation output serves as the stimulus to the ADC under test. It is clear that the final DAC output has 3 control words: k_1, d, and k_2. We denote the combined output as $V[k_1,d,k_2]$. Without repeating the control words, the DiDDEM DAC can have N_1*p*N_2 output samples.

7.2 Output Distribution

We use the output of the DiDDEM DAC as the stimulus to the ADC. The test performance is determined by the distribution of:

$$V[k_1,d,k_2] \quad (1 \le k_1 \le N_1, 1 \le d \le p \;\&\; 1 \le k_2 \le N_2)$$

In the following, the test error will be evaluated by estimating $e(V_t)$ defined in (1.3) for any voltage V_t in $[V_{min}, V_{max}]$.

First, let's briefly review the DDEM DAC output distribution. For the purpose of clarity, some equations in Chapter 4 are rewritten here with changes on the subscripts. Another important difference we should point out is that the quantization error effect will be ignored in the following analysis. This is partially justified by the fact that the incorporation of the dithering technique allows for further reduction in the DAC apparent resolution as well as the DDEM iteration number p. Since p will stay relatively small, the total effect of quantization errors will be dominated by the DAC nonlinearity. In the unlikely case when a relatively high-resolution DAC is designed and a large iteration number p is used in DDEM together with dither incorporated, the following derivation should be modified in a similar way as we did in Chapter 4 to accommodate for the effect of quantization errors.

The DDEM DAC has N_1 current elements $i_j \; (j = 1,...,N_1)$. Let $i_{N+j} = i_j \;\; (j = 1,..,N_1)$, then virtually we have $2N_1$ current elements $i_1, i_2, ..., i_{N_1}, i_{N_1+1}, ..., i_{2N_1}$.

Let $V_1[0]=0$ and $V_1[k_1]=R_C\cdot\sum_{j=1}^{k_1} i_j\ (k_1=1,...,2N_1)$, in which R_C is the output resistance of the DDEM DAC. Note that the first half of the sequence is the output voltage sequence of a regular n_1-bit DAC. Let $V_{1m}=V_1[N_1]$. V_{1m} is the maximum output of the DDEM DAC. Define

$$LSB_1=V_{1m}/N_1$$

Now define *INL* and *DNL* for the DDEM DAC. Let

$$INL_1[k_1]=\left(V_1[k_1]-k_1\cdot LSB_1\right)/LSB_1\ (k_1=0,...,2N_1)$$

and $$DNL_1[k_1]=\left(V_1[k_1]-V_1[k_1-1]-LSB_1\right)/LSB_1\ (k_1=1,...,2N_1)$$

With DDEM cyclic switching sequence, the DAC outputs p ramps. Let $q=N/p$. The d^{th} $(1\le d\le p)$ ramp is given by $R_C\cdot\sum_{j=1+q(d-1)}^{k+q(d-1)} i_j\ (k=1,...,N_1)$, which is equivalent to:

$$\left\{V_1^{(d)}[k_1]\right\}=\left\{V_1[k_1+q(d-1)]-V_1[q(d-1)]:1\le k_1\le N_1\right\}$$

For any V_t less than V_{1m}, let $g^{(d)}(V_t)$ denote the number of elements in $\left\{V_1^{(d)}[k_1]\right\}$ that are not larger than V_t.

$$g^{(d)}(V_t)=\left|\left\{V_1^{(d)}[k_1]:V_1^{(d)}[k_1]\le V_t,1\le k_1\le N_1\right\}\right|.$$

Let $$g^{(0)}(V_t)=floor\left(N_1\cdot V_t/V_{1m}\right)$$

We have the following approximation for $g^{(d)}(V_t)$:

$$g^{(d)}(V_t)\approx g^{(0)}(V_t)+\frac{V_t-V_1^{(d)}\left[g^{(0)}(V_t)\right]}{LSB_1}\ (1\le \mathrm{d}\le \mathrm{p})$$

$$\approx\frac{V_t}{LSB_1}-\sum_{k_1=1}^{g^{(0)}(V_t)}DNL_1[k_1+q(d-1)]\ \ (1\le \mathrm{d}\le \mathrm{p})$$

It is clear that the number of DDEM DAC output samples that are less than V_t is given by:

$$g(V_t)=\sum_{d=1}^{p} g^{(d)}(V_t) \approx p\frac{V_t}{LSB_1}-\sum_{d=1}^{p}\sum_{k_1=1}^{g^{(0)}(V_t)} DNL_1[k_1+q(d-1)]$$

If $g^{(0)}(V_t)=q\cdot t+m$ $(0<m\le q, 0\le t<p, t \,\&\, m\in Z)$, finally, we have:

$$g(V_t)\approx p\frac{V_t}{LSB_1}-\sum_{d=1}^{p}\sum_{k_1=1}^{m} DNL_1[k_1+q(d-1)] \tag{7.2}$$

Now look at the output of the DiDDEM DAC. We denote the n_2-bit dither DAC output sequence as $\{V_2[k_2]: 1\le k_2\le N_2\}$. Let $V_{2m}=V_2[N_2]$ .We can also define INL[k] and DNL[k] for the dither DAC.

The DiDDEM DAC output is expressed as $V[k_1,d,k_2]=V_1[k_1,d]+\alpha V_2[k_2]$.

The maximum output of the combined DAC is:

$$V_m=V_{1m}+\alpha V_{2m}$$

For any V_t less than V_m, let $h(V_t)$ denote the number of output samples that are not larger than V_t.

$$h(V_t)=\left|\{V[k_1,d,k_2]: V[k_1,d,k_2]\le V_t\}\right|$$

By definition, $h(V_t)=\sum_{k_2=1}^{N_2} g\left(V_t-\alpha V_2[k_2]\right)$.

Substitute (7.2) into it, and we have:

$$\begin{aligned} h(V_t) &= \sum_{k_2=1}^{N_2}\left\{p\frac{V_t-\alpha V_2[k_2]}{LSB_1}-\sum_{d=1}^{p}\sum_{k_1=1}^{m_{k_2}} DNL_1[k_1+q(d-1)]\right\} \\ &= N_2 p\frac{V_t}{LSB_1}-p\alpha\sum_{k_2=1}^{N_2}\frac{\alpha V_2[k_2]}{LSB_1}-\sum_{k_2=1}^{N_2}\sum_{d=1}^{p}\sum_{k_1=1}^{m_{k_2}} DNL_1[k_1+q(d-1)] \end{aligned} \tag{7.3}$$

in which m_{k_2} is given by $g^{(0)}\left(V_t-\alpha V_2[k_2]\right)=q\cdot t_{k_2}+m_{k_2}$.

There are 3 items in (7.3). The first item is the linear part and the second is a fixed value for any V_t. The third item contains the nonlinearity with respect to V_t. Let $h_e(V_t)$ denote this nonlinearity and change the summation order.

$$h_e(V_t) = -\sum_{d=1}^{p}\sum_{k_2=1}^{N_2}\sum_{k_1=1}^{m_{k_2}} DNL_1[k_1 + q(d-1)] \tag{7.4}$$

Let

$$h_{ed}(V_t) = \sum_{k_2=1}^{N_2}\sum_{k_1=1}^{m_{k_2}} DNL_1[k_1 + q(d-1)] \tag{7.5}$$

If the dither DAC is ideal, then m_{k_2} will be values taken from 1 to q when k_2 varies. Each number from 1 to q will be taken by $r_0 = \frac{N_2}{q}$ times. Then

$$h_{ed}(V_t) = \sum_{m=1}^{q}\left(r_0 \sum_{k_1=1}^{m} DNL_1[k_1 + q(d-1)] \right) \tag{7.6}$$

With the non-ideal dither DAC, the r_0 in (7.6) should be replaced by $r_0 + r_e[m]$, in which

$$r_e[m] \approx \sum_{k_2=(m-1)N_2/q+1}^{mN_2/q} DNL_2[k_2]$$

Then we have:

$$h_{ed}(V_t) \approx \sum_{m=1}^{q}\left(\left(r_0 + \sum_{k_2=(m-1)N_2/q+1}^{mN_2/q} DNL_2[k_2] \right)\sum_{k_1=1}^{m} DNL_1[k_1 + q(d-1)]\right) \tag{7.7}$$

If the standard deviations of the DDEM DAC and dither DAC DNL[k]'s are σ_1 and σ_2, respectively, the variance of $h_{ed}(V_t)$ is given by:

$$Var_{ed}(V_t) \approx \left(\sum_{k=1}^{q} k \frac{N_2}{q}\sigma_2^2 \right)\cdot \sigma_1^2 = \frac{(q+1)\sigma_1^2 N_2 \sigma_2^2}{2}$$

The variance for $h_e(V_t)$ is:

$$Var_e(V_t) = p \cdot Var_{ed}(V_t) \approx \frac{N_1\sigma_1^2 N_2\sigma_2^2}{2} \tag{7.8}$$

The normalized $h_e(V_t)$ standard deviation is given by:

$$\overline{\sigma}_e = \sqrt{Var_e((V_t))}/(N_1 p N_2) \approx \sqrt{2}\sqrt{N_1/2}\sigma_1 \cdot \sqrt{N_2/2}\sigma_2/(N_1 p N_2)$$

Also, we usually have:

$$\sigma_{INL_{k1}} = \sqrt{N_1/2}\sigma_1 \text{ and } \sigma_{INL_{k2}} = \sqrt{N_2/2}\sigma_2 .$$

If $n_{ENOB1} = \log_2 \frac{N_1}{\sigma_{INL_{k1}}}$ and $n_{ENOB2} = \log_2 \frac{N_2}{\sigma_{INL_{k2}}}$, then

$$\log_2 \frac{1}{\overline{\sigma}_e} = \log_2 p + n_{ENOB1} + n_{ENOB2} - 0.5 .$$

We may ignore this -0.5, and the test performance of the DiDDEM DAC will be equivalent to that of a DAC with ENOB equal to:

$$\tilde{n}_{ENOB} \approx \log_2 p + n_{ENOB1} + n_{ENOB2} \tag{7.9}$$

In real situations, the test performance is expected to be slightly lower than that given in (7.9) due to other non-major non-idealities not included in this analysis.

7.3 Simulation Results

The test performance using the proposed DiDDEM DAC is verified by simulation. In the simulation, the DDEM DAC is a 9-bit DAC, and the dither DAC is a 6-bit DAC; both have the same current element distribution with a normalized standard deviation equal to 0.1, which means that the mismatch can be up to about 30% if we count the 3σ range. The DDEM iteration number p is 64. The simulated ADCs under test are 14-bit ADCs. The

estimated INL[k] and INL with the DiDDEM DAC as the test stimulus are compared to true INL[k] and INL. The differences are the test errors using the DiDDEM DAC.

Figure 7.2 depicts the ADC true INL[k], estimated INL[k] curves and estimation error from a single simulation. The simulated DDEM DAC has an INL of 1.89 LSB at the 9-bit level, and the dither DAC has an INL of 1.16 LSB at the 6-bit level, so $n_{ENOB1}+n_{ENOB2} \approx 12$ bits. According to (7.9), the expected performance is $n_{ENOB1}+n_{ENOB2}+\log_2 p \approx 18$ bits. In simulation, the ADC true INL is 3.15 LSB, and the estimated INL is 3.25 LSB. INL estimation error is only 0.1 LSB. The maximum INL[k] estimation error is 0.3 LSB at the 14-bit level. So, the test performance is at about the 16-bit level. The actual achieved test performance is comparable to that predicated by (7.9).

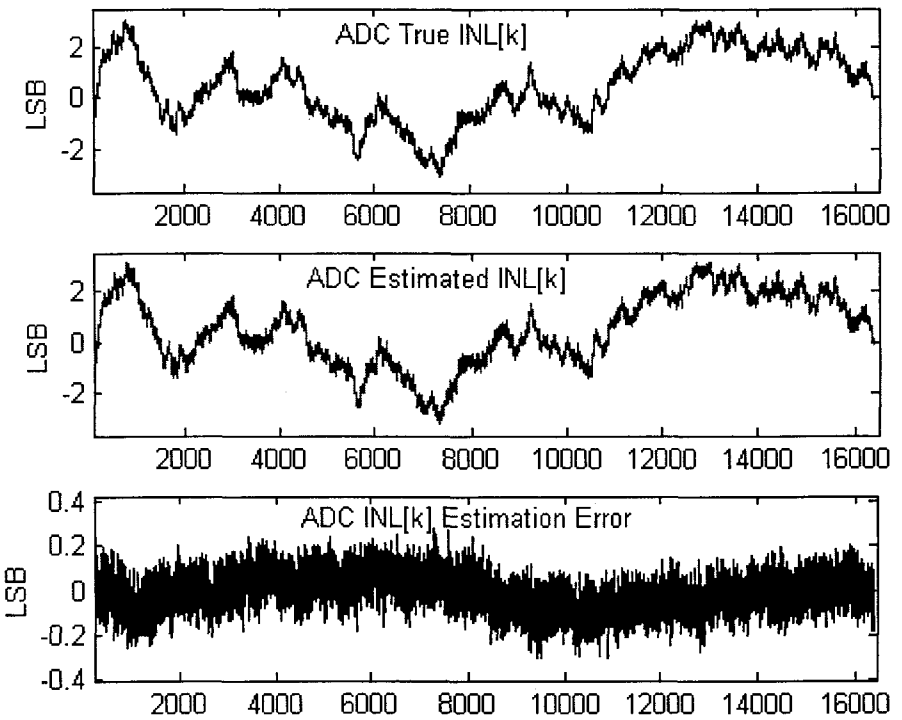


Figure 7.2 ADC true & est. INL[k] and estimation error using DiDDEM

To verify the robustness of the DiDDEM approach, 100 DiDDEM DACs are simulated to test 100 ADCs. The simulation setting is the same as previous. Figure 7.3 shows the estimated INL versus the true INL curve for these 100 DAC-ADC pairs. The maximum INL estimation error (estimated INL – true INL) is 0.42 LSB, and the minimum error is –0.15 LSB. First, the errors are quite small, which means the DiDDEM approach is robust. Secondly, the lower bound (-0.15) is adequately small, which means there is almost no risk of underestimating ADC's INL with the DiDDEM DAC. This merit is critical for real test, since it can guarantee no "bad" parts are delivered as "good" parts.

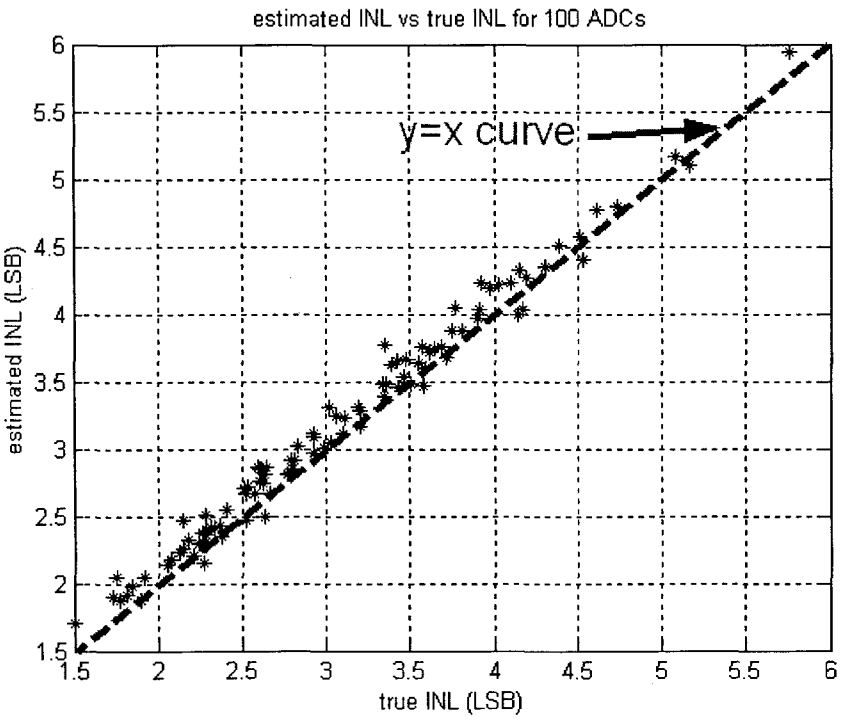


Figure 7.3 Estimated INL vs. true INL for 100 ADCs

In this chapter, a new dither-incorporated deterministic dynamic element matching (DiDDEM) approach has been proposed for high-resolution ADC test using extremely low-resolution DACs. With this approach, the outputs of a DDEM DAC and a dither DAC are

combined and serve as the stimulus to the ADC under test. Theoretical analysis shows that the test performance of a DiDDEM DAC is equivalent to that of a linear DAC, with ENOB equal to the summation of the DDEM DAC ENOB and dither DAC ENOB plus $\log_2 p$, where p is the DDEM iteration number. Simulation results show that the actual test performance is comparable to the theoretical prediction. The robustness and reliability of this approach have also been verified by simulation.

Conclusions

Cost-effective stimulus sources have been viewed as a bottleneck for BIST of ADCs, especially those deeply imbedded in SoCs. In this work, the Deterministic Dynamic Element Matching (DDEM) technology is presented as a solution to this problem. It has been shown through rigorous theoretical analysis, simulation results, and experimental verification that this DDEM approach can be applied on low-resolution/low-linearity current steering DACs to provide stimulus sources for the high-resolution ADC test. The robust performance and low implementation cost make the DDEM DAC an efficient on-chip solution to testing high-resolution ADCs and, consequently, an enabling technology for BIST of SoCs.

The major contributions of this work include:

1) Theoretical analysis of DDEM performance as an ADC test stimulus source

It is shown from the statistics viewpoint that the DDEM DAC has a flat output PDF. Then, a direct inspection on DDEM DAC's output voltage samples rigorously shows that the DDEM DAC equivalent linearity is given by

$$n_{eq} \approx \begin{cases} ENOB_{DAC} + \log_2 p & p < p_T \\ ENOB_{DAC} + \log_2 p_T - 1 & p = p_T \\ ENOB_{DAC} + \log_2 p_T - 1 + 0.5\log_2(p/p_T) & p > p_T \end{cases}$$

in which $ENOB_{DAC}$ is the original DAC raw linearity, p is the DDEM iteration number, and p_T is a transition value determined by the DAC element distribution property.

2) DDEM parameter optimization

Directed by the theoretical analysis, DDEM performance optimization has been made with the aid of numeric simulations.

3) Systematic design approach of DDEM DAC

Guided by the insight obtained from the analytical analysis, a systematic approach for cost-effective DDEM DAC design is developed.

4) Design and testing of 2 DDEM DACs

- First generation: an 8-bit DDEM DAC. With DDEM, this 8-bit DDEM DAC (less than 5-bit linear without DDEM) can achieve 12-bit equivalent linearity after systematic error compensation. The performance is already better than any on-chip linear ramp generator in terms of ADC test performance in the literature.
- Second generation: a 12-bit DDEM DAC. This design is an improved version of the 8-bit one. 16-bit equivalent linearity has been achieved, although the original linearity is only less than 10 bits. This outperforms any linear ramp generator by 5 bits.

5) New ADC test techniques based on DDEM

- DDEM on segmented thermometer-coded DAC is proposed by collaborating with Beatriz Olleta. The STC DDEM DAC gives a performance comparable to the basic DDEM DAC, while the die size requirement is largely reduced.
- Dither incorporated DDEM ADC is proposed to use extremely low-resolution DACs to test high-resolution ADCs with a robust performance.

References

[1] SIA, "International Technology Roadmap for Semiconductors," 2003 edition, http://public.itrs.net/Files/2003ITRS/Home2003.htm

[2] M. Burns and G.W. Roberts, "An Introduction to Mixed-Signal IC Test and Measurement," Oxford University Press, New York, 2000

[3] IEEE Std 1241-2000, "IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters", the Institute of Electrical and Electronics Engineers, Inc., New York, 2001

[4] B. Provost and E. Sanchez-Sinencio, "On-chip ramp generators for mixed-signal BIST and ADC self-test", IEEE Journal of Solid-State Circuits, vol. 38, no. 2, pp.263 -273, Feb. 2003

[5] K. L. Parthasarathy, Le Jin, D. Chen and R. L. Geiger, "A Modified Histogram Approach for Accurate Self-Characterization of Analog-to-Digital Converters", in Proceedings of 2002 IEEE International Symposium on Circuits and Systems, vol. 2, pp. II-376-379, Scottsdale, AZ, May 2002

[6] K. Parthasarathy, T. Kuyel, D. Price, Le Jin, D. Chen and R. L. Geiger, "BIST and Production Testing of ADCs Using Imprecise Stimulus", ACM Transactions on Design Automation of Electronic Systems, vol.8, no. 4, pp. 522-545, Oct. 2003

[7] B. Olleta, L. Juffer, D. Chen, and R. L. Geiger, "A Deterministic Dynamic Element Approach to ADC Testing". in Proceedings of 2003 IEEE International Symposium on Circuits and Systems, vol. 5, pp. V-533-535, Bangkok, Thailand, May 2003

[8] H. Jiang, B. Olleta, D. Chen, R. Geiger, "Parameter Optimization of Deterministic Dynamic Element Matching DACs for Accurate and Cost-Effective ADC Testing", in Proceedings of 2004 IEEE International Symposium on Circuits and Systems, vol. 1, pp. I-924-927, Vancouver, Canada, May 2004

[9] B. Olleta, H. Jiang, D. Chen, and R. L. Geiger, "Testing High Resolution ADCs using Deterministic Dynamic Element Matching", in Proceedings of 2004 IEEE International Symposium on Circuits and Systems, vol. 1, pp. I-920-923, Vancouver, Canada, May 2004

[10] H. Jiang, B. Olleta, D. Chen and R. L. Geiger, "Testing High Resolution ADCs with Low Resolution/ Accuracy Deterministic Dynamic Element Matched DACs", in Proceedings of 2004 International Test Conference, pp. 1379-1388, Charlotte, NC, Oct. 2004

[11] H. Jiang, B. Olleta, D. Chen and R. L. Geiger, "A Segmented-Thermometer-Coded DDEM DAC for high resolution ADC Test", in Proceedings of 2005 IEEE International Symposium on Circuits and Systems, vol. 1, pp.784-787, Kobe, Japan, May 2005

[12] H. Jiang, D. Chen and R. L. Geiger, "Dither Incorporated Deterministic Dynamic Element Matching for High Resolution ADC Test Using Extremely Low Resolution DACs", in Proceedings of 2005 IEEE International Symposium on Circuits and Systems, vol. 5, pp.4285-4288, Kobe, Japan, May 2005

[13] W. Schofield, D. Mercer, L. S. Onge, "A 16b 400MS/s DAC with <-80dBc IMD to 300MHz and <-160dBm/Hz noise power spectral density", in 2003 IEEE International Solid-State Circuits Conference Digest of Technical Papers, vol.1, pp. 126-482, San Francisco, CA, Feb. 2003

[14] C. Jansson, K. Chen and C. Svenssson, "Linear, polynomial and exponential ramp generators with automatic slope adjustment", IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 41 , no. 2, pp. 181-185, Feb. 1994

[15] F. Azaïz, S. Bernard, Y. Bertrand and M. Renovell, "Towards an ADC BIST scheme using the histogram test technique", in Proceedings of 2000 IEEE European Test Workshop, pp. 53-58, Cascais, Portugal, May 2000

[16] J. Wang; E. Sanchez-Sinencio and F. Maloberti, "Very linear ramp-generators for high resolution ADC BIST and calibration", in Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems, pp. 908 -911, East Lansing MI, Aug. 2000

[17] S. Bernard, F. Azais, Y. Bertrand and M. Renovell, "A high accuracy triangle-wave signal generator for on-chip ADC testing", Proceedings of 2002 IEEE European Test Workshop, pp. 89-94, Corfu, Greece, May 2002

[18] R. J. Van De Plassche, "Dynamic element matching for high-accuracy monolithic D/A converters", IEEE Journal of Solid-State Circuits, vol. 11, no. 6, pp. 795-800, Dec. 1976

[19] L. R. Carley, "A noise-shaping coder topology for 15+ bit converters", IEEE Journal of Solid-State Circuits, vol. 24, no. 2, pp. 267-273, Apr. 1989

[20] H. T. Jensen and I. Galton, "A Low-Complexity Dynamic Element Matching DAC for Direct Digital Synthesis", IEEE Transactions on Circuits and Systems, vol. 45, pp. 13-27, Jan. 1998.

[21] H. T. Jensen and I. Galton, "An analysis of the partial randomization dynamic element matching technique", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 45, pp. 1538-1549, Dec. 1998

[22] B. H. Leung and S. Sutarja, "Multibit Σ-Δ A/D converter incorporating a novel class of dynamic element matching techniques", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 39, pp 35-51, Jan. 1992

[23] R. T. Baird and T. S. Fiez, "Linearity Enhancement of Multibit ΔΣ A/D and D/A Converters Using Data Weighted Averaging", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 42, pp. 753-762, Dec. 1995

[24] D. Cini, C. Samori and A. L. Lacaita, "Double-index averaging: a novel technique for dynamic element matching in Σ-Δ A/D converters", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 46, pp. 353-358, Apr. 1999

[25] R. E. Radke, A. Eshraghi and T. S. Fiez, "A 14-Bit Current-Mode ΣΔ DAC Based Upon Rotated Data Weighted Averaging", IEEE Journal of Solid-State Circuits, Vol. 35, pp. 1074-1084, Aug. 2000

[26] I. Fujimori, L. Longo, A. Hairapetian, K. Seiyama, S. Kosic, Jun Cao and Shu-Lap Chan; "A 90-dB SNR 2.5-MHz output-rate ADC using cascaded multibit delta-sigma modulation at 8× oversampling ratio", IEEE Journal of Solid-State Circuits, vol. 35, pp. 1820-1828, Dec. 2000

[27] R. W. Adams and T. W. Kwan, "Data-Directed Scrambler for Multi-bit Noise Shaping D/A Converters", US Patent 5,404,142, Apr. 1995

[28] B. Olleta, D. Chen, and R. L. Geiger, "A Dynamic Element Matching Approach to ADC Testing", in Proceedings of the 45th IEEE Midwest Symposium on Circuits and Systems, vol. 2, pp II-549-552, Tulsa, OK, Aug. 2002

[29] Y. Cong and R. L. Geiger, "Formulation of INL and DNL yield estimation in current-steering D/A converters", in Proceedings of 2002 IEEE International Symposium on Circuits and Systems, vol. 3, pp. 149-152, Scottsdale, AZ, May 2002

[30] P. E. Allen and D. R. Holberg, "CMOS Analog Circuit Design", Oxford University Press, 2000

[31] D. A. Johns and K Martin, "Analog Integrated Circuit Design". John Wiley & Sons, Inc., 1997

[32] J. M. Rabaey, "Digital Integrated Circuits: A Design Perspective". Prentice-Hall International, Inc., 1996

[33] M. F. Wagdy, "Effect of various dither forms on quantization errors of ideal A/D converters", IEEE Transactions on Instrumentation and Measurement, vol. 38, no. 4, pp. 850-855, Aug. 1989

Acknowledgements

There have been many people helping me throughout the long journey to completion of this work. I would like to sincerely thank all my advisors, colleagues, friends, and sponsors for their instruction, support, assistance, and encouragement, in particular to the following people and agencies:

— Dr. Degang Chen and Dr. Randall Geiger for their instruction on my research work. Throughout my doctorate work, they have been continuously encouraging me to develop the ability of analyzing and solving problems, which I believe will benefit my whole life.

— Dr. Robert Weber, Dr. Chris Chu, and Dr. Fritz Keinert, my committee members, for their guidance and encouragement to finish this dissertation.

— Beatriz Olleta for her collaboration and precursive work on the DDEM approach.

— All my colleagues for their suggestions and assistance to complete this work, especially to Le Jin, Chengming He, Hanqing Xing, Mark Schlarmann, and Saqib Malik for their input on the theoretical analysis, circuit design, and experimental verification.

— Nancy Knight, Pam Myers, and all the other department staff for their help during my doctorate study.

— Semiconductor Research Corporation, National Science Foundation, the R. J. Carver Charitable Trust, and Iowa State University for their partial financial support to my doctorate work.

— MOSIS for their support on fabrication of the two DDEM DACs.

— Intel Corporation for the opportunity of practice training in the area of AMS circuit testing and debugging, which is directly related to my research work, especially to Mr. Paul Lind, Mr. James Zhou, and Mr. Nello Tanner for their supervision and instruction during my internship.

— Finally, my parents far away in China and my wife for their encouragement and support in completing this work.