

2006

# Methodology for testing high-performance data converters using low-accuracy instruments

Le Jin

*Iowa State University*

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Electrical and Electronics Commons](#)

## Recommended Citation

Jin, Le, "Methodology for testing high-performance data converters using low-accuracy instruments " (2006). *Retrospective Theses and Dissertations*. 1269.

<https://lib.dr.iastate.edu/rtd/1269>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact [digirep@iastate.edu](mailto:digirep@iastate.edu).

**Methodology for testing high-performance data converters using  
low-accuracy instruments**

by

**Le Jin**

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

2006

Copyright © Le Jin, 2006. All rights reserved.

UMI Number: 3217283

### INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

**UMI**<sup>®</sup>

---

UMI Microform 3217283

Copyright 2006 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company  
300 North Zeeb Road  
P.O. Box 1346  
Ann Arbor, MI 48106-1346

Graduate College  
Iowa State University

This is to certify that the doctoral dissertation of

Le Jin

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

Chapter 1 Introduction .....	1
Analog-to-Digital Converter Test Using Low-Linearity Stimulus .....	2
Digital-to-Analog Converter Test Using Low-Resolution ADCs.....	5
Dissertation Organization.....	6
Chapter 2 Accurate Testing of Analog-to-Digital Converters Using Low Linearity Signals with Stimulus Error Identification and Removal.....	7
Abstract .....	7
I. Background.....	8
II. Introduction .....	10
III. ADC Linearity Testing Using Nonideal Stimuli.....	12
IV. Error Analysis .....	24
V. Simulation results .....	28
VI. Test Results from A 16-bit SAR ADC .....	32
VII. Conclusions.....	37
References .....	37
Chapter 3 SEIR Linearity Test of Precision A/D Converters in Non-Stationary Environments with Center-Symmetric Interleaving .....	40
Abstract .....	40
I. Introduction.....	41
II. ADC Linearity Testing with SEIR .....	42
III. Non-Stationary Test Environment .....	45
IV. Center-Symmetric Interleaving of Test Signals.....	48
V. Simulation and Experimental Results .....	58
VI. Conclusions .....	67
References .....	67
Chapter 4 Code-Density Test of Analog-to-Digital Converters Using Single Low- Linearity Stimulus Signal.....	69

Abstract .....	69
I. Introduction.....	69
II. Characteristics of Pipeline ADCs.....	73
III. ADC Testing with Single Nonlinear Stimulus Signal.....	77
IV. Simulation Results .....	81
V. Experimental Results.....	85
VI. Limitations and Further Discussions .....	89
VII. Conclusions.....	91
Acknowledgements .....	92
References .....	92
 Chapter 5 Linearity Test of Analog-to-Digital Converters Using Kalman Filtering .....	 95
Abstract .....	95
I. Introduction.....	95
II. Code-Density Test for ADC Linearity .....	97
III. ADC Linearity Test with Kalman Filtering .....	102
IV. Simulation Results .....	108
V. Experimental Results.....	115
VI. Conclusions .....	119
Acknowledgements .....	119
References .....	119
 Chapter 6 Testing of Precision DACs Using Low-Resolution ADCs with Dithering ....	 121
Abstract .....	121
I. Introduction.....	122
II. Existing Methods for DAC Testing.....	124
III. Linearity Specifications of DACs .....	126
IV. Test Precision DACs Using Low-Resolution ADCs with Dithering .....	127
V. Performance Analysis and Other Issues.....	134
VI. Simulation Results .....	140
VII. Experimental Results .....	144

VIII. Conclusions .....	148
Acknowledgements .....	149
References .....	149
Chapter 7 Conclusions .....	151
Acknowledgements .....	153

# Chapter 1

## Introduction

There has been explosive growth in the consumer electronics market during the last decade. As the IC industry is shifting from PC-centric to consumer electronics-centric, digital technologies are no longer solving all the problems and electronic devices integrating mixed-signal, RF and other non-purely digital functions are becoming new challenges to the industry. System-on-a-chip (SoC) design and built-in self-test are two important technologies behind the integration of these functions and are of great interest to the industry and the academia.

When digital testing has been studied for long time, testing of analog and mixed-signal circuits is still in its development stage. Existing solutions for testing analog and mixed-signal circuits have two major problems. First, high-performance mixed-signal test equipments are expensive and it is difficult to integrate their functions on chip. Second, it is more and more challenging to improve the test capability of existing methods to keep up with the fast-evolving performance of mixed-signal products demanded on the market. The *International Technology Roadmap for Semiconductors* identified mixed-signal testing as one of the most daunting system-on-a-chip challenges.

My works have been focused on developing new strategies for testing data converters using system identification and data processing algorithms. Different from the conventional methods that require test instruments to have better performance than the device under test, our algorithms allow the use of medium and low-accuracy instruments in testing. Therefore, we can provide practical and accurate test solutions for high-performance data converters.

Meanwhile, the test cost is dramatically reduced because of the low price of such test instruments. These algorithms have the potential for built-in self-test and can be generalized to other mixed-signal circuits. When incorporated with self-calibration, these algorithms can enable new design techniques for mixed-signal integrated circuits.

### **Analog-to-Digital Converter Test Using Low-Linearity Stimulus**

The analog-to-digital converter (ADC) is one of the world's largest volume analog and mixed-signal (AMS) integrated circuit products and is viewed as one of the system drivers for AMS chip design. Linearity test of high-performance ADCs is a well-known important and challenging problem. A precision linearity test can help validate the design of a high-performance ADC, reduce the number of wasted parts, and enable calibration, so it is necessary to have methods for accurately characterizing linearity of high-resolution high-speed ADCs. Also the testing cost has essential meaning to the manufacturers because of the high volume.

The ADC testing capability is mainly determined by three enabling technologies: fast data capture, precision clock timing and linear stimulus generation. The bottle neck in testing of next generation high-performance ADCs is the linear signal generation, as the present state-of-the-art technologies on timing and data capture can handle the testing need of up coming ADCs. The code-density test method is widely adopted for testing ADCs' static linearity in the industry, because its implementation is straightforward and its computational complexity is low. This method uses a ramp or sine wave as the stimulus signal, with linearity at least one decade better than the specification of the ADC under test. This linearity requirement makes the test of high-performance ADCs an increasingly challenging problem, since the ADC resolution is continuously going up along with the emerging demand of high-

performance applications in communications, imaging, and industrial controls. Furthermore, a full-code histogram test for a high-resolution ADC requires a large number of samples, which implies long and expensive test time. Due to long testing time, the nonstationarity of the test environment will cause errors in linearity testing as well. Because of the above facts, there is lack of widely-adopted cost-effective approaches for testing high-performance ADCs that are pushing the edge of current technologies.

If the requirement on highly linear signals can be removed, the testing problem for high-speed high-resolution ADCs becomes much more tractable. It can be shown that one nonlinear signal is generally insufficient for ADC testing. My colleagues and I developed a stimulus error identification and removal (SEIR) algorithm for testing ADCs that uses two unknown nonlinear signals. The two signals will generate two sets of histogram data when used to test an ADC. It is required that the two signals have identical nonlinearity and the offset between them is a constant voltage. Therefore, the two histograms are strongly correlated to each other with small difference introduced by the offset. By exploiting the relationship between the two test signals, the input nonlinearity can be identified and its effect on the histogram data can be removed. The ADC's linearity performance will then be accurately characterized with the identified input nonlinearity information. Since the SEIR algorithm allows the test signal to have lower linearity than that of the ADC under test, high-resolution ADCs can be tested with low-accuracy instruments.

The SEIR algorithm requires a constant offset between the two nonlinear test signals, but a physical test environment is always time varying and will introduce errors in the offset. A strategy that minimizes the effect of environment nonstationarity on test accuracy is proposed. It can be shown that if the two signals are generated with the proposed pattern, called center-

symmetric interleaving (CSI), the dominant time-varying errors can be cancelled and a constant offset can be obtained. The combination of the SEIR algorithm and the CSI strategy provides a solution to the challenging problem of testing high-performance ADCs, utilizing low-accuracy instruments in a realistic time-varying environment. This approach dramatically reduces the test time and cost, as the stimulus signal can be easily generated at very high speeds, and enables the built-in self-test of high-resolution ADCs, as a simple stimulus generator can be integrated on chip.

The above SEIR-CSI methodology is a general I/O based testing method. It takes the ADC under test as a black box and does not make use of any information on the ADC architecture. More investigation shows that exploiting extra knowledge on the ADC architecture can help further simplify the SEIR testing algorithm. We developed an approach using a single nonlinear stimulus signal for testing ADCs with some widely adopted architectures, including cyclic and pipeline. It first identifies and removes the input errors based on the ADC's structure characteristics, and then accurately measures the ADC's linearity. The single signal test algorithm further simplifies the ADC testing problem, as compared to the SEIR algorithm, by making use of some readily available information. Since pipelined and cyclic ADCs are very popular in nowadays IC systems, this algorithm can have a significant impact on the ADC design practice.

For current and upcoming high-resolution ADCs, time for full-code INL and DNL test, which is directly related to the cost, is prohibitively long because of the large number of variables to be accurately measured. We proposed a modified code-density algorithm that uses the Kalman filter to reduce the effect of errors on the histogram data. This algorithm can achieve the same level of accuracy as that of the conventional code-density algorithm but

using a significantly smaller number of samples, which means shorter test time and lower test cost. The additional computations introduced by the modified algorithm is very few and negligible with current days' computing power. The new method is very efficient and can be used to enable testing of high-resolution ADCs with better coverage and reduce the time and cost of testing medium-resolution ADCs.

### **Digital-to-Analog Converter Test Using Low-Resolution ADCs**

The digital-to-analog converter (DAC) serves as the interface between the digital processing functions and analog signals. As the SoC design style getting more and more popular and requirements for high-quality AMS circuitries continuously going up, the demand for high-performance DACs is growing rapidly. World's leading AMS integrated circuits companies, such as Analog Devices, Texas Instruments and National Semiconductors, are all manufacturing high-speed high-resolution DACs for applications such as wireless communications and digital signal processing. The best commercial parts, such as AD9779 from ADI and DAC5687 from TI, have 16-bit resolutions and more than 500 MSPS update rates. The next generation products with better performance are currently under working and will come to the market very soon. Along with the advancement in DAC performance, there are consequently new needs in DAC design and testing.

Bench test plays important roles in design development, parameter tuning, debugging, and product validation stages of a DAC, while production test measures the specifications, sifts good, bad and marginal parts, and enables calibration for improving the performance of a DAC. An efficient testing method with high accuracy, short test time and low cost is very necessary for both of the two cases. It is well known that DAC testing is more challenging than ADC testing, as DACs usually have higher resolutions and speeds than ADCs.

Measurement devices used in conventional DAC testing methods should have better performance and run faster than the device under test (DUT) to provide accurate characterization results. It is a nontrivial task to manufacture sufficiently fast and accurate instruments for testing the current and future highest-performance DACs.

Our work is targeting at providing cost-effective solutions to the high-performance DAC testing problem. We have come up with and investigated a novel method of using low-accuracy instruments to test high-performance DACs. It is shown that high-resolution DACs can be accurately tested by using low-resolution ADCs with appropriate voltage dithering. Because of the availability of very high-speed low-resolution ADCs, this approach provides a potential solution to the testing problem for high-speed high-resolution DACs.

### **Dissertation Organization**

This dissertation comprises a collection of five papers introducing new methods for testing data converters using low-accuracy instruments. Chapter 2 is a paper published in the *IEEE Transactions on Instrumentation and Measurement*, presenting the stimulus error identification and removal algorithm. Chapter 3 is a manuscript submitted to the above transactions based on a paper published in the *Proceedings of 2005 International Test Conference*. It discusses the implementation of the SEIR algorithm with center-symmetric interleaving. Chapter 4, 5 and 6 are three manuscripts submitted to 2006 International Test Conference, respectively. Chapter 4 presents the algorithm for testing ADCs of some widely-adopted structures using a single nonlinear signal. Chapter 5 talks about how to use Kalman Filter to improve the efficiency of high-resolution ADC testing. Chapter 6 discusses a digital-to-analog converter testing algorithm using low-resolution ADCs with dithering.

## Chapter 2

# Accurate Testing of Analog-to-Digital Converters Using Low Linearity Signals with Stimulus Error Identification and Removal

A paper published in the IEEE Transactions on Instrumentation and Measurement

Le Jin, Kumar Parthasarathy, Turker Kuyel, Degang Chen, and Randall Geiger

### Abstract

Linearity testing of analog-to-digital converters (ADCs) can be very challenging because it requires a signal generator substantially more linear than the ADC under test. This paper introduces the Stimulus Error Identification and Removal (SEIR) method for accurately testing ADC linearity using signal generators that may be significantly less linear than the device under test. In the SEIR approach, two imprecise nonlinear but functionally related excitations are applied to the ADC input to obtain two sets of ADC output data. The SEIR algorithm then uses the redundant information from the two sets of data to accurately identify the nonlinearity errors in the stimuli. The algorithm then removes the stimulus error from the ADC output data, allowing the ADC nonlinearity to be accurately measured. For a high resolution ADC, the total computation time of the SEIR algorithm is significantly less than the data acquisition time and therefore does not contribute to testing time. The new approach was experimentally validated on production test hardware with a commercial 16-bit successive approximation ADC. Integral nonlinearity test results that are well within the device specification of  $\pm 2$  LSB were obtained by using 7-bit linear input signals. This

approach provides an enabling technology for cost-effective full-code testing of high precision ADCs in production test and for potential cost-effective chip-level implementation of a built-in self-test capability.

## I. Background

The “histogram method” is a standard approach for quasi-static linearity testing of analog-to-digital converters (ADCs) [1-3]. However, during the past decade, linearity testing of ADCs has not received much research attention due to several reasons. As long as best practices are followed, modern mixed-signal Automated Test Equipments (ATEs) can be used to make quasi-static linearity testing of ADCs a fairly straightforward production task for low to medium resolution ADCs [4]. High-precision delta-sigma ADCs are inherently sufficiently linear and do not require linearity testing. In the communications circuit area, high-speed pipelined ADCs are widely used and are usually production tested with high-frequency input signals [2], whereas quasi-static linearity testing is primarily used for debugging [5] or calibration [6]. Probably the biggest reason, however, can be attributed to the challenges associated with generating highly linear or spectrally pure test signals with no major technological breakthroughs occurring in this area in the past decade.

Nevertheless, quasi-static linearity testing remains a test challenge for the production of certain classes of high performance ADCs and the increasing downward production cost pressures are making the convenient use of expensive mixed-signal ATEs for testing low and medium resolution ADCs more difficult to justify. In this work, emphasis will be placed on the more challenging task of quasi-static linearity testing of high performance ADCs with little mention of the low and medium resolution devices, but application of the concepts introduced here for the production testing of low and medium resolution devices is

straightforward and in some applications may provide a more cost-effective test flow for low and medium resolution devices that does not require time on expensive mixed signal ATEs.

In a high performance ADC, specifications like 16-bit or higher resolution, 1 MSPS or higher conversion rate, little or no output latency, and an input signal frequency exceeding the ADC's Nyquist rate are common. Recent examples include 16-bit 1.25 MSPS and 18-bit 500 KSPS Successive Approximation Register (SAR) ADCs and a 16-bit 5 MSPS multi-bit delta sigma ADC [7]. These ADCs employ techniques such as precision laser trimming or dynamic element matching to achieve high linearity at relatively high sampling speeds. These high performance ADCs are typically used in medical applications including ultrasound and computer aided tomography as well as precision industrial process control and ATEs that serve the testing industry.

To better appreciate the challenges in quasi-static linearity testing of high precision ADCs, the performance requirements on the signal sources used to generate the input to the ADC under test will be reviewed. Conventional wisdom dictates that signal sources must be significantly more linear than the ADC under test. Usually, an acceptable test procedure would provide test accuracy to within 10% of the device specification. An ADC with a  $\pm 2$  LSB maximum linearity error specification would require test accuracy to be well within  $\pm 0.2$  LSB. Considering that one LSB is around  $76 \mu\text{V}$  for a 16-bit ADC with a 5 V supply, source linearity of better than  $15.2 \mu\text{V}$  (0.2 LSB) is required and providing this degree of source linearity is an extremely challenging task. The task is even more challenging when testing an 18-bit ADC. Some applications dictate that all codes of the ADC be tested in production, resulting in long test times that often run several minutes on an expensive mixed signal ATE. The long test time is usually required to average out the effects of input noise in

the test environment. However, requiring the source to remain stationary (low drift) during such long tests presents another challenge for linearity testing. Source architectures used in ATE equipment that provide good linearity, like the delta-sigma structures, are not known for good drift performance, and vice versa. In addition, linear sources often have slow settling characteristics and the source settling often dominates acquisition and test time. Practical test solutions for future high precision ADCs require a relaxation of the performance requirements on the signal source, an apparent paradoxical expectation.

## **II. Introduction**

In this work, a new approach for ADC testing has been developed that relaxes the requirement on source linearity. If the source is allowed to be nonlinear with no stringent requirements on the specific linearity characteristics and no need for prior knowledge on the characteristics of the nonlinearity, the design requirements for the source will be dramatically reduced. Such sources can be designed to have better drift characteristics, and to work faster, properties which are key to improving accuracy and reducing ADC test time. Furthermore, such nonlinear sources can be placed on the Device Interface Board (DIB) to reduce requirements and cost of the ATE or even incorporated on chip with a small die area to facilitate use in a Design for Test (DFT) or a Built-In Self-Test (BIST) environment. Recent research using the concept of using nonlinear excitations for ADC testing can be found in [8, 9, 11-13, 15-17]. As a proof of concept, two different approaches were discussed in the authors' previous work [8]. One of these approaches was sensitive to device noise making applications to precision ADCs difficult. The second included an algorithm that requires a matrix inversion with computational complexity proportional to the cube of the total number of ADC output codes which is not computationally-effective for high-resolution ADCs. With

both algorithms, low spatial-frequency nonlinearities in the signal source, a property that can be readily attained, were assumed. An application of one of these algorithms to testing 10-bit ADCs is reported in [9]. 10-bit resolution appeared to be a practical performance limit on the specific algorithm used in this previous work.

This paper presents the Stimulus Error Identification and Removal (SEIR) method for accurate and robust testing of ADC quasi-static linearity performance. This approach uses two imprecise nonlinear signals, one shifted with respect to the other by a constant voltage offset, to excite the ADC and obtain two sets of ADC outputs. By matching the signal levels corresponding to the same ADC transition level, a set of equations involving only the stimulus error components is established. This separates the signal source nonlinearity from the nonlinearity inherent in the ADC under test. By parameterizing the signal source nonlinearity using a set of basis functions, a set of equations linear in the parameterization coefficients are obtained. Standard Least Square (LS) methods can then be used to accurately identify the coefficients and thus identify the nonlinearity errors in the stimuli. The SEIR algorithm subsequently removes the stimulus error from the ADC output data, allowing the ADC nonlinearity to be accurately measured.

Both simulation results and experimental test results obtained from commercially available ADCs are presented. Experimental results confirm the fact that the identification and removal of the effects of input nonlinearity can be performed as a digital signal processing task during production by a tester computer with reasonably short time. Production test hardware typically used for high performance 16-bit SAR ADCs, which is a real challenge in the analog and mixed-signal linearity testing area, has been used to verify the SEIR method. Full code testing results from this new method are successfully correlated with those obtained in

an industry laboratory using state of the art production test equipment for mixed-signal circuits. The test time required for implementation of the proposed testing approach is short, making it viable for use in a production test environment.

### III. ADC Linearity Testing Using Nonideal Stimuli

In this section, a mathematical formulation of the proposed SEIR approach will be presented. First, the modeling of an ADC based on transition levels [18] and the nonlinear input signals along with the definition of transition times are presented. This is followed by a discussion on integral nonlinearity testing. Finally, a mathematical formulation that is used to estimate and remove the effects of input nonlinearity is given and a new ADC characterization approach using nonlinear input signals is described.

#### *ADC and input nonlinearity modeling*

Let us consider an  $n$ -bit ADC with  $N = 2^n$  output codes. The static input-output characteristic of such a device can be modeled as

$$D(x) = \begin{cases} 0, & x \leq T_0; \\ k, & T_{k-1} < x \leq T_k, k = 1, 2, \dots, N-2; \\ N-1, & T_{N-2} < x; \end{cases} \quad (2.1)$$

where  $D$  is the digital output code,  $x$  is the analog input voltage, and  $T_k, k = 0, 1, \dots, N-2$ , are transition levels of the ADC. If the input signal voltage is less than  $T_k$ , the output digital code will be less than or equal to  $k$ . If the input signal voltage is larger than  $T_k$ , the output digital code will be greater than  $k$ . Although the transition levels are generally indexed from 1 to  $N-1$ , we use indexes 0 to  $N-2$  to number them in this paper. This will make it a bit easier for us to give definitions of some other terms later. The choice of indexes does not change the meaning of the transition levels. Equation (2.1) is valid under the assumption that the ADC is

monotonic and has no missing codes. This assumption is justifiable for most of the commercial ADCs.

Linearity testing of an ADC corresponds to investigating how linearly transition levels of an ADC are distributed. An ideally linear ADC with the same terminal transition levels,  $T_0$  and  $T_{N-2}$ , has transition levels uniformly spaced between  $T_0$  and  $T_{N-2}$  with a constant voltage increment of  $Q=(T_{N-2}-T_0)/(N-2)$ . This increment is called a Least Significant Bit (LSB). Transition levels of the ideally linear ADC are called terminal-based ideal transition levels and denoted as  $I_k$ . They can be expressed as

$$I_k = T_0 + \frac{T_{N-2} - T_0}{N - 2}k, \quad k = 0, 1, \dots, N - 2. \quad (2.2)$$

Equation (2.2) is called a terminal-based fit line. It represents a straight line connecting the terminal transition levels of the ADC. For linearity testing, true transition levels of an ADC will be compared to the corresponding terminal-based ideal transition levels. The terminal-based integral nonlinearity for code  $k$  ( $INL_k$ ) is defined to be the difference between the true and terminal-based ideal transition levels. Expressing  $INL_k$  in LSB, we get

$$INL_k = \frac{T_k - I_k}{Q} = \frac{T_k - T_0}{T_{N-2} - T_0}(N - 2) - k \text{ (LSB)}, \quad k = 1, 2, \dots, N - 3. \quad (2.3)$$

The overall terminal-based integral nonlinearity ( $INL$ ) is then defined by the expression

$$INL = \max_k \{ | INL_k | \}. \quad (2.4)$$

A larger value of terminal-based  $INL$  indicates that an ADC has higher nonlinearity. For simplicity the word “terminal-based” will not be carried in the following part of the paper, but all the nonlinear parameters,  $INL$  and  $INL_k$ 's, used in this work are based on the terminal transition levels and the corresponding terminal-based ideal transition levels.

An ideal ramp signal, as assumed in traditional linearity testing, can be visualized as a signal that increases linearly with time  $t$ , whereas a more realistic ramp signal always has some nonlinearity that makes it deviate from a straight line. A real ramp signal can be modeled as

$$x(t) = x_{os} + \eta t + F(t), \quad (2.5)$$

where  $x_{os}$  is a DC offset voltage,  $\eta t$  is the linear part of the signal and  $F(t)$  is the nonlinear component. Defining the transition time  $t_k$  to be the time at which the value of the analog ramp signal is equal to the  $k^{\text{th}}$  transition level of the ADC, we get

$$T_k = x(t_k), \quad k = 0, 1, \dots, N-2. \quad (2.6)$$

Monotonicity of the signal source is assumed in this work and hence the output codes sampled before time  $t_k$  will be always less than or equal to  $k$ . Effects of the noise in the input signal will be discussed in Section IV. To simplify the derivation, we perform some linear operations on equation (2.5) which will not affect the final test results. First, we denote the first transition time to be the origin of time scale, i.e.  $t_0 = 0$ . Second, we normalize the time intervals so that the last transition time corresponds to the unit time, i.e.  $t_{N-2} = 1$ . By doing so, we have scaled and translated the time axis to be unit free so that our algorithm will look the same for all clock frequencies and will be independent of the actual time when tests are conducted. The linear component  $\eta t$  of the signal is defined such that the nonlinearity in the input signal is 0 at both  $t = 0$  and  $t = 1$ , that is

$$F(0) = F(1) = 0. \quad (2.7)$$

These operations are equivalent to choosing

$$x_{os} = T_0 \text{ and } \eta = T_{N-2} - T_0. \quad (2.8)$$

Substituting equation (2.8) into (2.5), we get

$$x(t) = T_o + (T_{N-2} - T_o)t + F(t), \quad 0 \leq t \leq 1. \quad (2.9)$$

Equation (2.9) represents a signal whose magnitude is equivalent to the first and last transition levels of the ADC at the normalized times 0 and 1, respectively. With this notation, the nonlinearity of the input signal is completely characterized by  $F(t)$ .

Assuming no prior knowledge about the general form of  $F(t)$ , we try to identify this input nonlinearity independently during the test. As a first step, we expand  $F(t)$  over a complete set of basis functions denoted by  $\{F_j(t), j=1,2,3,\dots\}$ . By identifying a finite number of the major coefficients of the basis functions for this expansion, we can estimate the value of  $F(t)$  to an accuracy higher than the resolution of the ADC under test. To simplify the derivation, we choose familiar and widely-used trigonometric functions on  $[-1, 1]$  to be the set of basis functions. Applying odd extension on  $F(t)$  to cover the interval  $[-1, 1]$ , which includes a mathematically negative time, we get:

$$\tilde{F}(t) = \begin{cases} F(t), & 0 \leq t \leq 1; \\ -F(-t), & -1 \leq t < 0. \end{cases} \quad (2.10)$$

$\tilde{F}(t)$  can be expanded in terms of trigonometric functions as

$$\tilde{F}(t) = \sum_{j=1,2,\dots} a_j \sin(j\pi t) + \sum_{j=0,1,\dots} b_j \cos(j\pi t), \quad -1 \leq t \leq 1, \quad (2.11)$$

where  $a_j, j=1, 2, \dots$  and  $b_j, j=0, 1, 2, \dots$  are the coefficients of the  $j^{\text{th}}$  harmonic. Since the extended function is odd, the coefficients of the cosine functions are all 0 and only sine functions are needed to express the nonlinearity. On  $[0, 1]$ ,  $F(t)$  can thus be parameterized as

$$F(t) = \sum_{j=1}^M a_j \sin(j\pi t) + e(t). \quad (2.12)$$

Since we are only interested in a finite accuracy expansion of  $F(t)$ , only the first  $M$  basis functions are included in (2.12) and thus  $e(t)$  is the residue of the nonlinearity that is not modeled by the  $M$  basis functions. By completeness of the basis function set,  $M$  can always be appropriately chosen so that the residue is small to any desired level.  $F(t)$  is said to be identified if we can determine the value of  $a_j, j=1, 2, \dots, M$ . For simplicity, we will not carry the term  $e(t)$  most of the time in the following derivations. The effect of neglecting the term  $e(t)$  will be analyzed in Section IV. Other choices for a set of basis functions  $\{F_j(t), j=1, 2, 3, \dots\}$  can also be used to approximate  $F(t)$  and each element of the set should vanish at  $t=0$  and  $t=1$ . Formally, they will satisfy the relationships

$$\begin{aligned} F(t) &\cong \sum_{j=1}^M a_j F_j(t), & 0 \leq t \leq 1; \\ F_j(0) &= F_j(1) = 0, & j = 1, 2, \dots, M. \end{aligned} \quad (2.13)$$

An example of an alternative set of basis functions is the set of polynomial functions

$$\begin{aligned} F_1(t) &= t(t-1); \\ F_2(t) &= t(t-1)(t-0.5); \\ F_3(t) &= t(t-1)(t-0.5)(t-0.25); \\ F_4(t) &= t(t-1)(t-0.5)(t-0.25)(t-0.75); \\ &\dots \end{aligned} \quad (2.14)$$

Using the expanded nonlinear component, the input signal can be written as

$$x(t) \cong T_o + (T_{N-2} - T_o)t + \sum_{j=1}^M a_j F_j(t), \quad 0 \leq t \leq 1. \quad (2.15)$$

Figure 2.1 illustrates the relationships between the true and terminal-based fit line transition levels, the input and output of an ADC, and the ideal and a realistic ramp signal. The horizontal axis corresponds to time with transition time points labeled. The vertical axis corresponds to the input voltage with transition levels labeled. The region corresponding to

different output codes are denoted as dotted areas. The output code of an ADC will be a digital code  $k$  when time is between transition times  $t_{k-1}$  and  $t_k$  and correspondingly the input signal value is between the transition levels  $T_{k-1} = x(t_{k-1})$  and  $T_k = x(t_k)$ .

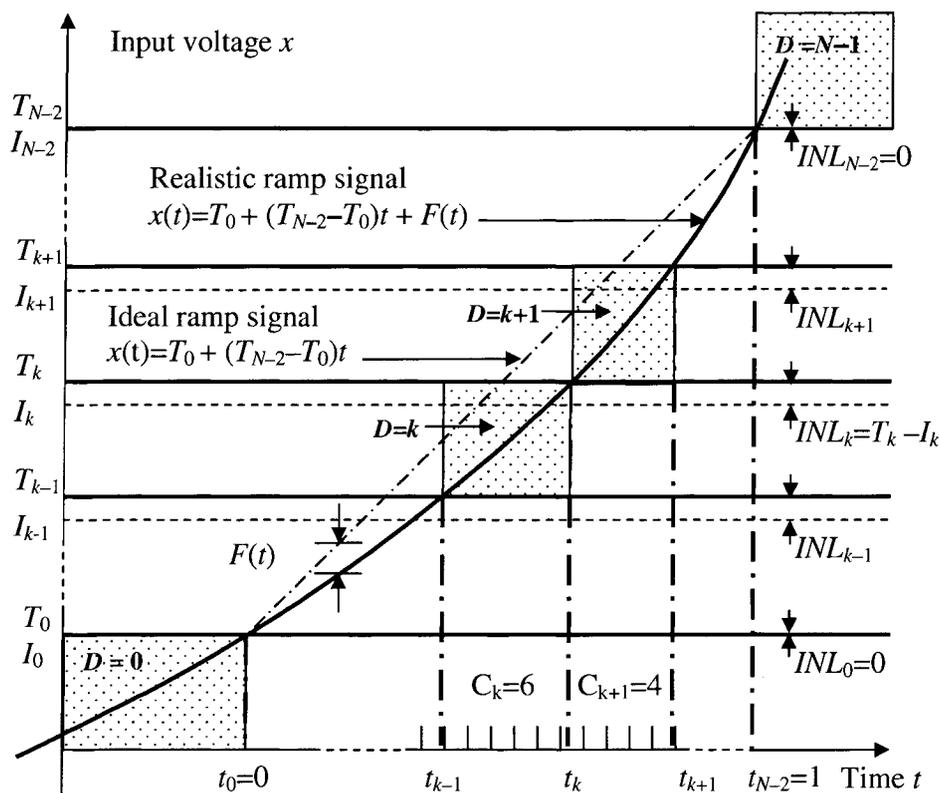


Figure 2.1 Basic terminology in ADC linearity testing.

### **Linearity testing for ADCs**

The goal of ADC linearity testing is to identify transition levels and determine the  $INL$  of an ADC. However, transition levels of an ADC cannot be measured directly from the ADC output. An alternative is to measure transition times and calculate the values of transition levels by using equation (2.6). Substituting equation (2.15) into (2.6), we have

$$T_k \cong T_o + (T_{N-2} - T_o)t_k + \sum_{j=1}^M a_j F_j(t_k), \quad 0 \leq t_k \leq 1. \quad (2.16)$$

Equation (2.3) can then be used to express  $INL_k$  as a function of the associated transition time and coefficients of the nonlinear component,  $a_j$ 's,

$$INL_k \cong (N-2)t_k + \sum_{j=1}^M a_j F_j(t_k) - k, \quad k = 1, 2, \dots, N-3. \quad (2.17)$$

In equation (2.17), coefficient  $a_j$ 's are in terms of LSBs.

Transition times of an ADC can be measured by using the traditional histogram test. Let  $C_k$ ,  $k = 0, 1, 2, \dots, N-1$ , represent the bin counts obtained in a histogram test for each code. If the sampling period of an ADC is a constant, the time when a sample is taken is linearly proportional to the number of samples that have been taken so far. So the number of samples can be viewed as a measurement of time. For instance,  $C_1$  samples of code 1 will have been taken since  $t = 0$  when the output code changes from 1 to 2, indicating that  $C_1$  sampling periods of time have elapsed. Similarly,  $C_1 + C_2$  samples will have been taken when the output code changes from 2 to 3. In general,  $C_1 + C_2 + \dots + C_k$  sampling periods of time will have elapsed since  $t = 0$  when the output code changes from  $k$  to  $k+1$ . This leads to the following time instances:

$$\hat{t}_k = T_c \sum_{i=1}^k C_i, \quad k = 1, 2, \dots, N-2, \quad (2.18)$$

where  $T_c$  is a scaling factor that scales the time period measured in terms of the number of samples to the normalized time defined earlier. Since the output code changes from  $N-2$  to  $N-1$  when  $t = 1$ , the total number of samples taken between  $t = 0$  and  $t = 1$  is given by

$$C_1 + C_2 + \dots + C_{N-2} = \sum_{i=1}^{N-2} C_i. \quad (2.19)$$

Therefore an appropriate scaling factor is given by

$$T_c = \left( \sum_{i=1}^{N-2} C_i \right)^{-1}. \quad (2.20)$$

Since by definition  $\hat{t}_k$  is the last sampling instance before the output code changes from  $k$  to  $k+1$ , we have

$$x(\hat{t}_k) \leq T_k \text{ and } T_k < x(\hat{t}_k + T_c). \quad (2.21)$$

Since  $x(t)$  is assumed to be a monotonically increasing function of time, we have

$$\hat{t}_k \leq t_k < \hat{t}_k + T_c. \quad (2.22)$$

Thus estimating the  $k^{\text{th}}$  transition time using the time instance defined in (2.18) involves an uncertainty of at most one clock period as shown in (2.22). The magnitude of this uncertainty can be reduced by increasing the number of samples taken so that  $T_c$  is sufficiently small. In such a case, it is safe to assume the approximation error is insignificant and

$$t_k \cong \hat{t}_k = T_c \sum_{i=1}^k C_i. \quad (2.23)$$

Substituting (2.20) into (2.23), we have

$$\hat{t}_k = \sum_{i=1}^k C_i / \sum_{i=1}^{N-2} C_i. \quad (2.24)$$

Using equation (2.17), we have an estimate for  $INL_k$

$$\hat{INL}_k = (N-2)\hat{t}_k + \sum_{j=1}^M a_j F_j(\hat{t}_k) - k, \quad k = 1, 2, \dots, N-3. \quad (2.25)$$

If input nonlinearity were known in a parameterized form, equation (2.24) and (2.25) would relate the bin counts  $C_k$ 's to the  $INL_k$ 's of an ADC. However, input nonlinearity is typically not known before hand and cannot be determined from equation (2.25). Equation (2.25) comprises a set of  $N-3$  linear equations in the  $N-3$   $INL_k$  values and the  $M$  nonlinearity

coefficients representing a total of  $N+M-3$  unknowns. Thus the set of equations given in (2.25) are, in general, insufficient to solve for all of the unknowns. The traditional histogram method for determining  $\hat{INL}_k$  is a special case of the situation above for which the input signal is assumed to be ideally linear so that all  $a_j = 0$  and equation (2.25) can be simplified to

$$\hat{INL}_k = (N-2)\hat{t}_k - k, \quad k = 1, 2, \dots, N-3. \quad (2.26)$$

The assumption is good only when the maximum input nonlinearity is much smaller than 1 LSB. However, if the input nonlinearity is comparable to or larger than 1 LSB, it will introduce significant errors in the  $INL_k$  estimation if equation (2.26) is used to estimate the  $INL_k$ . The estimation error can be obtained by subtracting equation (2.26) from (2.17) and is given as:

$$\hat{INL}_k - INL_k = -\sum_{j=1}^M a_j F_j(\hat{t}_k) + d(\hat{t}_k - t_k), \quad k = 1, 2, \dots, N-3. \quad (2.27)$$

The first term on the right-hand side of (2.27) is the result of input nonlinearity and the second term comes from the errors in transition time approximation. The input nonlinearity gets included in the estimated values of  $INL_k$ . This will result in misinterpretation of the true linearity performance of an ADC if neglected. For example, if we use an input source with 10-bit linearity to test a 16-bit ADC with a true 1-LSB  $INL$ , equation (2.26) will estimate the ADC to have about 64-LSB  $INL$ . The second term in equation (2.27),  $d(\hat{t}_k - t_k)$ , is the effect of quantization error in the transition time. However, with a reasonable number of samples per code, this error is usually a small fraction of 1 LSB and much smaller than the error caused by the input nonlinearity. The effects of the above terms will be discussed in details in Section IV.

***ADC linearity testing with multiple nonlinear stimuli***

In equation (2.25), nonlinearities from the ADC and from the input signal are coupled with each other and the task of identifying them simultaneously is not apparently doable. One possible approach towards solving this problem is to separate and identify the input nonlinearity first and then test the ADC linearity performance with that knowledge. The proposed SEIR approach involves testing the ADC with two input signals. The two signals are otherwise identical except an unknown but fixed offset  $\alpha$  between them. Following the form of equation (2.9), the two input signals can be expressed as

$$x_1(t) = T_0 + (T_{N-2} - T_0)t + F(t), \quad (2.28)$$

$$x_2(t) = T_0 + (T_{N-2} - T_0)t + F(t) - \alpha. \quad (2.29)$$

The linear and nonlinear components in the two signals are the same with the only difference being the DC offset voltage. As described earlier for a single input case, each of the two input signals can define a set of transition times as in (2.6),

$$T_k = x_1(t_k^{(1)}), \quad (2.30)$$

$$T_k = x_2(t_k^{(2)}). \quad (2.31)$$

Since the two signals are used to test the same ADC, the transition levels referred to in equation (2.30) are the same as those referred to in (2.31). However, the two sets of transition times are different because of the offset change and have been denoted differently with superscripts 1 and 2, respectively. For example, if  $\alpha$  is positive, it will take a longer time for the 2<sup>nd</sup> signal to reach a transition voltage than it will take for the 1<sup>st</sup> signal to reach the same transition voltage, i.e.  $t_k^{(1)} < t_k^{(2)}$ . If we equate the right hand sides of equation (2.30) and (2.31) with respect to the same transition level, the ADC nonlinearity, which is embodied by

the transition levels  $T_k$ , will disappear and we will get equations involving only the input nonlinearity. These equations can be used to identify the input nonlinearity.

Let  $C_k^{(1)}$  and  $C_k^{(2)}$ ,  $k = 0, 1 \dots N-2$ , be the histogram data collected by using  $x_1$  and  $x_2$  as inputs to the ADC, respectively. Transition times can be estimated by using the bin counts as in equation (2.24),

$$\hat{t}_k^{(1)} = \sum_{i=1}^k C_i^{(1)} / \sum_{i=1}^{N-2} C_i^{(1)}, k = 1, 2 \dots N-2, \quad (2.32)$$

$$\hat{t}_k^{(2)} = \left[ \sum_{i=1}^k C_i^{(2)} - (C_0^{(1)} - C_0^{(2)}) \right] / \sum_{i=1}^{N-2} C_i^{(1)}, \quad k = 0, 1 \dots N-2. \quad (2.33)$$

In order for the two sets of transition times to have the same unit, both of the equations (2.32) and (2.33) are scaled and shifted by the same scaling factor and offset amount that are defined for the first signal, with origin at  $\hat{t}_0^{(1)} = 0$  and unit time at  $\hat{t}_{N-2}^{(1)} = 1$ . The 2<sup>nd</sup> signal having an offset in time is compensated in the numerator to resolve this issue as in equation (2.33). Similar to equation (2.25), we can then estimate the  $INL_k$  values using the estimated transition times,

$$\hat{INL}_k^{(1)} = (N-2)\hat{t}_k^{(1)} + \sum_{j=1}^M a_j F_j(\hat{t}_k^{(1)}) - k, \quad k = 1, 2 \dots N-3, \quad 0 \leq \hat{t}_k^{(1)} \leq 1; \quad (2.34)$$

$$\hat{INL}_k^{(2)} = (N-2)\hat{t}_k^{(2)} + \sum_{j=1}^M a_j F_j(\hat{t}_k^{(2)}) - \alpha - k, \quad k = 1, 2 \dots N-3, \quad 0 \leq \hat{t}_k^{(2)} \leq 1. \quad (2.35)$$

Notice that in equations (2.34) and (2.35), we only included those  $INL_k$  estimates for which the corresponding transition times are within the domain of definition for the basis functions. Because of this, the total number of equations available is  $N-3$  in (2.34) and  $N-3-\alpha$  in (2.35) if  $\alpha > 0$  (this is assumed below). It will be shown that for reasonable offset values, this reduction in the number of equations will not affect the performance of the proposed method.

Equations (2.34) and (2.35) constitute the body of the SEIR algorithm. As discussed earlier, we have a total of  $N+M-2$  unknowns. They are  $N-3$   $INL_k$ 's,  $M$   $a_j$ 's and the unknown offset  $\alpha$ . However, since two related input signals are used to test the ADC, we have  $2(N-3) - \alpha$  linear equations, nearly doubling of the number of unknowns. We are thus left with many more equations in (2.34) and (2.35) than the number of unknowns. Since  $INL_k$ 's in both equations are for the same ADC and must be equal at the same index  $k$ , by equating the individual  $INL_k$ 's in (2.34) and (2.35), we obtain a set of equations involving only input nonlinearity in a linear parameterization form,

$$(N-2)\hat{t}_k^{(1)} + \sum_{j=1}^M a_j F_j(\hat{t}_k^{(1)}) = (N-2)\hat{t}_k^{(2)} + \sum_{j=1}^M a_j F_j(\hat{t}_k^{(2)}) - \alpha. \quad (2.36)$$

Moving all the known terms to the left hand side and all terms with unknowns to the right hand side, we get

$$(N-2)(\hat{t}_k^{(2)} - \hat{t}_k^{(1)}) = \sum_{j=1}^M a_j (F_j(\hat{t}_k^{(1)}) - F_j(\hat{t}_k^{(2)})) + \alpha, \quad k = 1, 2, \dots, N-3, \hat{t}_k^{(2)} \leq 1. \quad (2.37)$$

For testing high resolution ADCs, typical values of  $M$  and  $\alpha$  (in LSB) are much smaller than  $N$ . Thus the number of equations in (2.37),  $N-3-\alpha$ , is much larger than  $M+1$ , the number of unknowns,  $a_j, j=1, 2, \dots, M$ , and  $\alpha$ . When the number of equations is larger than the number of unknown parameters, the system of equations comprising (2.37) is over constrained and the unknowns can be estimated by using the least squares (LS) method. For example, an ADC with more than 10-bit resolution will always have thousands of output codes while the number of parameters needed to model the nonlinearity in the excitation is usually less than 20 as we will discuss later. The LS method has an attractive property of effectively averaging

out the noise or errors in equation (2.37). The LS solution for estimating the unknowns can be expressed as

$$\{\hat{a}_1, \hat{a}_2, \dots, \hat{a}_M, \hat{\alpha}\} = \arg \min \left\{ \sum_{k=1, \hat{t}_k^{(2)} \leq 1}^{N-3} \left( (N-2)(\hat{t}_k^{(2)} - \hat{t}_k^{(1)}) - \left[ \sum_{j=1}^M a_j (F_j(\hat{t}_k^{(1)}) - F_j(\hat{t}_k^{(2)})) + \alpha \right] \right)^2 \right\}. \quad (2.38)$$

Once the input nonlinearity is identified from (2.38), ADC linearity testing becomes a straightforward job. Substituting solutions from (2.38) into either equation (2.34) or (2.35) or their combination, we can estimate the  $INL_k$  of an ADC. Using (2.34) for example, we have

$$\hat{INL}_k = (N-2)\hat{t}_k^{(1)} + \sum_{j=1}^M \hat{a}_j F_j(\hat{t}_k^{(1)}) - k, \quad k = 1, 2, \dots, N-3. \quad (2.39)$$

Equation (2.39) shows that the linearity performance of an ADC can be tested without being affected by input nonlinearity.

#### IV. Error Analysis

There are several sources of errors that will affect the performance of the SEIR approach. Among them, additive noise at the input to an ADC, the un-modeled error of the input signal nonlinearity as in equation (2.12), and the quantization error of transition times as in equation (2.27) have the most significant effects on the linearity testing results. Using the first signal as an example and considering the effects of noise and errors, the relationship between transition levels and the estimated transition time can be written as

$$T_k = T_o + (T_{N-2} - T_0)\hat{t}_k^{(1)} + \sum_{j=1}^M a_j F_j(\hat{t}_k^{(1)}) + e(\hat{t}_k^{(1)}) + n(\hat{t}_k^{(1)}) + d(\hat{t}_k^{(1)} - t_k^{(1)}), \quad (2.40)$$

where  $e(\hat{t}_k^{(1)})$  is the un-modeled error,  $n(\hat{t}_k^{(1)})$  is from the additive noise, and  $d(\hat{t}_k^{(1)} - t_k^{(1)})$  is from the quantization error in transition times. Without specific mention, we assume in the following part of this section that these noise and errors will not affect the LS estimation for

$\hat{a}_j$ 's such that they can be assumed to be the same as  $a_j$ 's. This is a fair assumption based on the following justifications. First, by definition, the un-modeled error is orthogonal to the first  $M$  sinusoidal functions. Second, the additive noise and quantization error usually change very fast as a function of time and hence have little correlation to the first  $M$  low frequency basis functions. Third, the LS method has the ability to average out the effects of noise and errors. Therefore the error between the true  $INL_k$  and that calculated in equation (2.39) can be written as

$$e_{INL_k} = INL_k - \hat{INL}_k = e(\hat{t}_k^{(1)}) + n(\hat{t}_k^{(1)}) + d(\hat{t}_k^{(1)} - t_k^{(1)}). \quad (2.41)$$

If there is no systematic error in the measurement process, all the errors can be assumed to be from normal distributions with zero mean and the estimation of  $INL_k$  is unbiased,

$$E\{e_{INL_k}\} \approx 0. \quad (2.42)$$

The variance of the error in the  $INL_k$  estimation can then be determined from equation (2.41), which is the summation of the variance of the un-modeled error, the variance of noise effects, and that of the quantization effects,

$$\text{Var}\{e_{INL_k}\} = \sigma_e^2 + \sigma_n^2 + \sigma_d^2. \quad (2.43)$$

The three types of error sources will be further discussed in the following sections.

### ***Effects of the un-modeled error in input signals***

The magnitude of  $e(\hat{t}_k^{(1)})$  is dependent on the number of basis functions used in parameterization, i.e.  $M$ , and on the nonlinearity of the input signal itself. This error term can be reduced to an arbitrarily small level by increasing  $M$ . Since signal generators with low spatial frequency can be easily realized practically, the nonlinearity in the input, though it

may be large, can be parameterized with a reasonably small number of basis functions and still guarantee that the residue error is small.

***Effects of the additive noise in input signals***

Let us assume the additive noise at the input to an ADC is stationary with zero mean and variance  $\sigma^2$ . The noise may cause the output code to be different from its expected value thereby changing the bin counts. Larger variance of the noise makes the code more likely to be different from its expected value. However, with a reasonably large number of samples per code, a change of one or two samples' value will not have a significant effect on the total number of samples for a code. Intuitively, the variance of  $n(\hat{t}_k^{(1)})$  should increase with the variance of the additive noise but decrease with the average number of samples per code. By writing out the probability of a sampled voltage with noise larger than a specific transition level, which is a Bernoulli random variable, the variance of this random variable can be calculated. The variance of the error from additive noise of the transition level is the summation of variances of all different sampled voltages weighted by the probability. The probability of each sampled voltage is the same, but the variance is small if a sampled voltage is far away from the transition level of interest and large if it is close. With detailed statistic analysis, it can be shown that the following general relationship is true,

$$\sigma_n^2 = A \frac{\sigma}{N_s}, \quad (2.44)$$

where  $N_s$  is the average number of samples per code and  $A$  is a constant dependent on the distribution of the noise, which can be determined numerically. For Gaussian additive noise,  $A$  is 0.5642. This sensitivity to noise is a fundamental problem in conventional histogram

based ADC test methods and the effects here are comparable to those experiences when ideal linear ramps are used for testing.

***Effects of the quantization error in transition times***

The quantization error of transition times is bounded by equation (2.22). A smaller  $T_c$  means a larger average number of samples per code, i.e., a larger  $N_s$ , which in turn leads to smaller quantization errors. Assuming uniform distribution of the quantization noise, the variance of the quantization error can be expressed in terms of  $N_s$  as

$$\sigma_d^2 = \frac{1}{12N_s^2}. \quad (2.45)$$

The quantization error is also a problem in traditional histogram based testing and comparable to that associated with this approach.

Typically, in an all codes production test environment,  $N_s$  is between 20 and 100 samples per code. The magnitude of the additive noise determines which term of (2.44) and (2.45) is more important to the testing result. If the standard deviation of the additive noise is comparable to 1 LSB, the effect of the quantization error will be much smaller than the effect of the additive noise. For high resolution ADCs, up to 1 *LSB* RMS noise is typical. This was the rationale behind neglecting the effect of quantization in the earlier part of the discussion.

***Effects of the offset between two signals***

The value of the offset voltage  $\alpha$  between the two input signals also affects the final *INL* estimation results. If the offset is too small, the difference between the nonlinearity of the two input signals at the same code level will be very small and noise will have significant effects on the LS method. The assumption that estimated parameters  $\hat{a}_j, j = 1, 2, \dots, M$  are close to the true value doesn't hold any more and the numerical behavior of the LS method is

no longer reliable in that situation. On the other hand the offset can not be too large either. As mentioned earlier, the last  $\alpha$  equations in (2.37) will not be used in estimating the parameters, and hence the LS result is only optimal for part of the input nonlinearity and not necessarily optimal for the nonlinearity on the whole interval of  $[0, 1]$ . Analysis shows that offsets of 0.1 to 1 % of full range are appropriate for the proposed method. Both simulation and experimental results support this conclusion. Furthermore, the SEIR method estimates the amount of offset, so no prior knowledge on the amount of offset is assumed.

In the discussion till now, it has been assumed that the two input signals are identical except for a fixed offset. This may become difficult to guarantee in real testing environment. Various time varying effects such as drift in the reference voltage could result in slight variations in the offset value. The signal source may change from one run to another, which will introduce gain error and different nonlinearity between two signals. These nonstationarity effects can be eliminated to a certain extent by well designed measurement procedures, if not completely eliminated. In both simulation and experimental measurement results provided in this work, the effects of reference voltage drift are considered. Part of the problem has been solved by interleaving the two input signals in time to excite the ADC and collect histogram data. Simulation and experimental results show that by interleaving the two signals with a “common-centroid” sequence, most of the non-stationary effects can be cancelled.

## V. Simulation results

To verify the performance of the SEIR approach, simulation has been done on ADCs of different resolutions and structures. Simulation results show that the SEIR approach can accurately identify *INL* of an ADC by using nonlinear excitations under various situations.

Results for a 14-bit simulated ADC under different noise levels and average numbers of samples per code are summarized in this section. For the purpose of simulation, the nonlinear input signal is modeled as

$$x_1(t) = v_{os} + \eta[t + 0.04 * (t^2 - t)] + n(t). \quad (2.46)$$

The maximum nonlinearity specified in (2.46) is 1% of the total input range. This corresponds to 7-bit linearity of the input signal. The offset between the first and second signals is 128 LSBs. However as mentioned earlier, this data is unknown to the algorithm. In the simulation, 20 sinusoidal basis functions were used in the parameterization of the input signal. The reference voltage has a time dependent drift with a linear gradient of 100 ppm over the total test time. This is to duplicate the effect of voltage drift in the real test environment. The two input signals are time-interleaved during data capture. The true  $INL_k$  of the simulated 14-bit ADC is plotted as a solid line in Figure 2.2 (a). The  $INL$  of the ADC is 14.88 LSB resulting in a 10-bit linear device. With an additive noise of a 0.8 LSB standard deviation and an average sample density of 32 samples per code, the  $INL_k$  estimated by the proposed approach is plotted as a dashed line in Figure 2.2 (a). The true and estimated curves, in solid and dashed lines respectively, match very well and are difficult to be differentiated from each other. A zoomed in version of the two curves for codes from 8500 to 8600 is plotted in Figure 2.2(b) and we can see there are very little errors between the solid and dashed lines. The difference between the true and estimated  $INL_k$ , which reflects the error in prediction using the algorithm is plotted in Figure 2.2 (c). The error in prediction is a fraction of 1 LSB with the maximum error being less than 0.7 LSB. The estimated  $INL$  of the ADC is 14.74 LSB, with a 0.14 LSB error from the true  $INL$ .

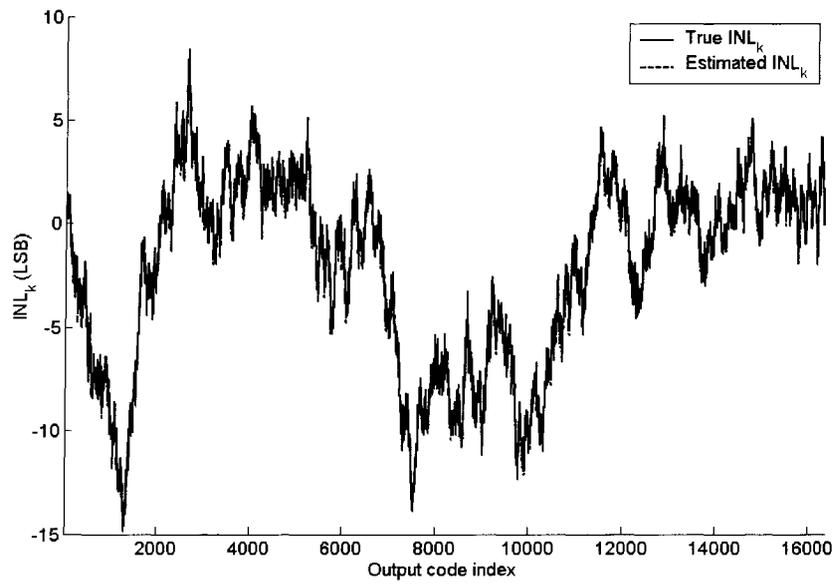


Figure 2.2 (a)

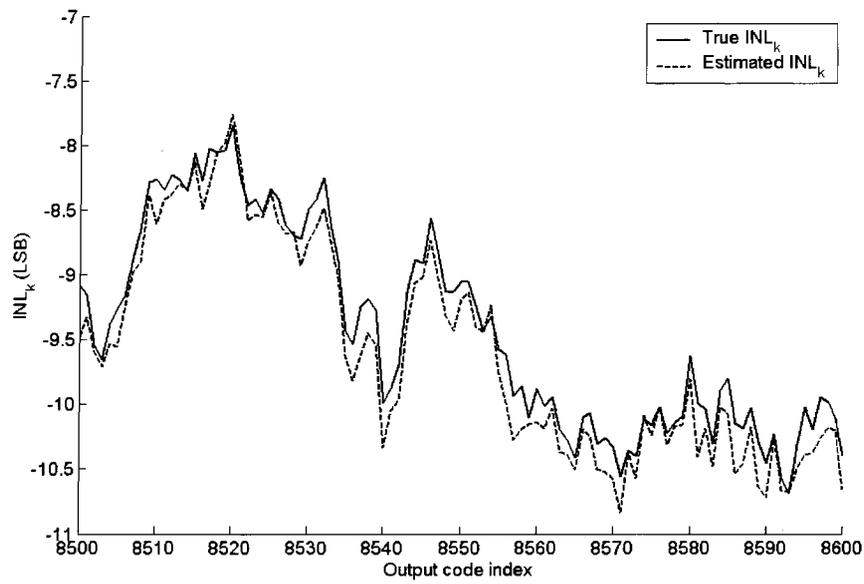


Figure 2.2 (b)

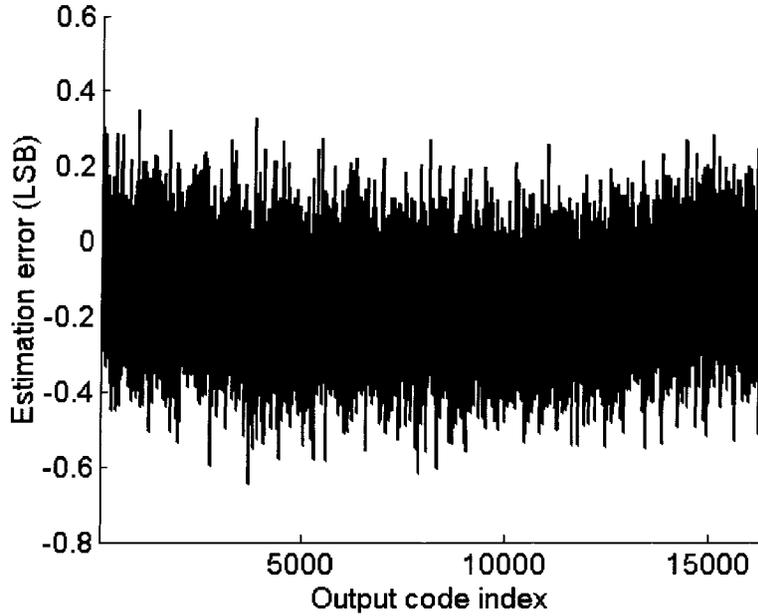


Figure 2.2 (c)

Figure 2.2  $INL_k$  estimation for a simulated 14-bit ADC. (a) True and estimated  $INL_k$  for all codes. (b) True and estimated  $INL_k$  zoomed in for codes from 8500 to 8600. (c) Error in  $INL_k$  estimation.

For the same ADC as shown in Figure 2.2, simulation under different combinations of the average number of samples per code,  $N_s$ , standard deviation of the noise,  $\sigma$ , and the number of basis functions,  $M_b$ , were performed and the results are summarized in Table 2.1. For each entry of the  $INL_k$  error reported in Table 2.1, the simulation was repeated 16 times and the average value of maximum  $INL_k$  errors are listed. It can be observed from Table 2.1 that when  $N_s$  is increased by 4 times, the estimation error is reduced by about 50%. When the standard deviation of the additive noise is increased by 4 times, the error in estimation is increased by 2 times. These statistics are in agreement with what is predicted by equation (2.44). When the number of basis functions,  $M_b$ , is increased from 5 to 20, the estimation

error is reduced by 40% and no further improvement is seen beyond the 0.8 LSB level, implying that the error in estimation is dominated by noise and error effects. From simulation results it was observed that using 20 basis functions to model the input nonlinearity is a reasonable choice.

Table 2.1. Maximum error in  $INL_k$  estimation for different  $N_s$ ,  $\sigma$ , and  $M_b$

1% input nonlinearity, $\alpha = 4\%$ , 100 ppm $V_{ref}$ drift					
$M_b = 20, \sigma = 3$ LSB		$N_s = 64, M_b = 20$		$N_s = 64, \sigma = 3$ LSB	
$N_s$	Est error (LSB)	$\sigma$ (LSB)	Est error (LSB)	$M_b$	Est error (LSB)
16	1.69	0.50	0.35	5	1.40
32	1.13	1.00	0.49	10	0.95
64	0.81	2.00	0.67	20	0.80
128	0.57	4.00	0.98	40	0.82

## VI. Test Results from A 16-bit SAR ADC

Commercially available 16-bit ADCs were tested to verify the performance of the SEIR method. The sample used as the device under test (DUT) was a laser trimmed 16-bit ADC with excellent linearity performance with typical  $INL$  of about 1.5 LSB, which is a known test challenge. The test hardware used for the verification of the proposed method is the same as used in the production test of the device.

### *Test setup*

Verification of the full performance of this ADC requires extreme attention to test hardware design. A 12-layer handler interface board is used with extensive ground, supply and

reference coverage. Extreme care is given to reduce ground loops and to obtain proper bypassing. High performance contactors, high precision resistors, high performance capacitors, and precision op-amps were used throughout the board. Latching relays were used to reduce temperature gradients generating metal to metal contact noise effects. The digital outputs were damped and buffered properly to avoid current surges. The test platform was a Teradyne A580 Advanced Mixed Signal Tester. The source generating both the linear and the synthetic nonlinear excitations was a 20-bit multi-bit delta-sigma DAC with 2ppm typical linearity error, 100 $\mu$ V/minute typical drift characteristics, and 2 kHz bandwidth. This source was a typical example demonstrating that an expensive signal generator is not always good enough to provide low drift, high speed and good linearity all at the same time. The DC offset of the nonlinear excitation was generated using an analog summing circuit. In the experiment, the capture of histogram data using nonlinear signals and identification of  $INL_k$  using the proposed method were done on different platforms. The tester setup, including the shape of input nonlinearity and offset between the two signals were not known to the identification algorithm. Only two sets of histogram bin counts were fed to the analysis program.

#### ***Test data collection and analysis***

The  $INL_k$  of the ADC was first tested by using the traditional histogram method using a highly linear ramp excitation generated by the tester. 32 samples per code were used to keep the test time reasonable. The  $INL_k$  plot is given on top in Figure 2.3. We will term this the “measured”  $INL_k$ . The corresponding measured  $INL$  is 1.66 LSB. This measured  $INL_k$  and  $INL$  will be used as a benchmark to evaluate the performance of the SEIR algorithm. Two nonlinear signals were synthetically generated by programming the source memory with a

nonlinear digital waveform. The DC offset was chosen to be 33 LSB. The histogram was obtained with the new nonlinear input and was analyzed using the proposed method. The estimated  $INL_k$  is plotted on bottom in Figure 2.3. The estimated  $INL$  using the nonlinear input signals is also 1.66 LSB. From Figure 2.3 we can see the  $INL_k$  estimated using linear and nonlinear input signals are really close. The difference between them is shown in Figure 2.4. The error in prediction is less than 1 LSB and is an acceptable solution as far as 16-bit converters are concerned.

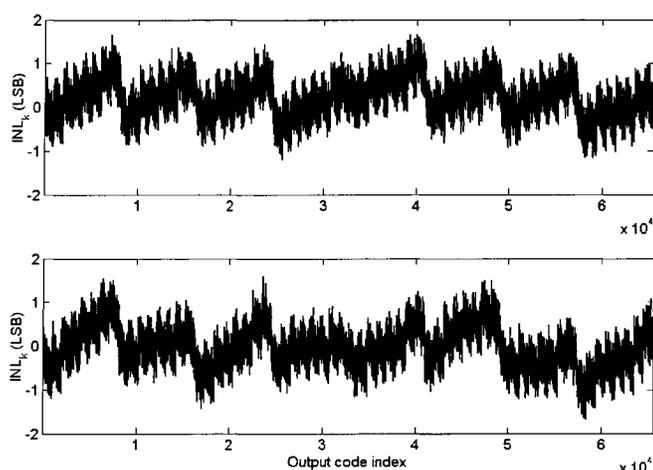


Figure 2.3  $INL_k$  estimation of a 16-b SAR DAC with true  $INL$  1.66 LSB. Top: estimated using the traditional histogram method with a linear stimulus. Bottom: estimated using the proposed approach with nonlinear stimuli.

The “low frequency” errors in  $INL_k$  estimation, as can be seen in Figure 2.4, come from the nonstationarity of the signal source. The nonstationarity errors could not be completely eliminated by the proposed algorithm since they introduce different nonlinearities into the two ramps signals, but we tried to minimize their effects in the experiments by interleaving the two ramps signals. The “high frequency” residue errors come from the additive noise in

the testing system. In a previous study on this device, the performance of all-code histogram test was compared to the performance of a reduced code test with a servo-loop. At each code, differences up to 0.7 LSB was found during the comparison, giving further justification to the premise that discrepancies indicating poor test capability do occur at the 16-bit level. This “high frequency” noise band can be further reduced by increasing the number of samples per code from 32 to a larger number. The results in Figure 2.3 and 4 were calculated by using the first 14 polynomial basis functions. The  $INL$  estimation using sinusoidal basis functions also gave similar performance. This supports that the SEIR method does not rely strongly on the class of basis functions used in the model.

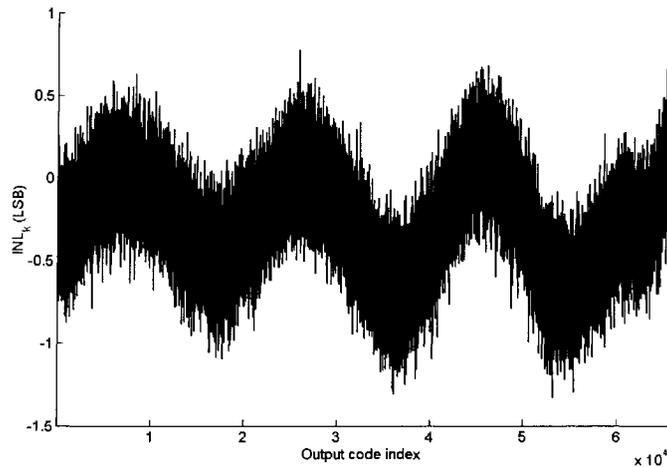


Figure 2.4 Difference between the estimated  $INL_k$  by using linear and nonlinear signals.

For the purpose of comparison,  $INL_k$  of the ADC was also estimated by using the traditional histogram method as in equation (2.26) with one of the two nonlinear signals. The result is plotted in Figure 2.5. This estimation of  $INL_k$  is significantly affected by the input nonlinearity as discussed in equation (2.27), which introduced an error of more than 300

LSBs. The results also indirectly indicate that the input signals are just nearly 7-bit linear. They are fairly linear for the real world, but for our 16-bit precision ADC, the amount of nonlinearity in the input is excessive. The inputs used in these tests were synthetically generated to be representative of real world quasi-linear analog ramp generators such as can be generated with simple integrators.

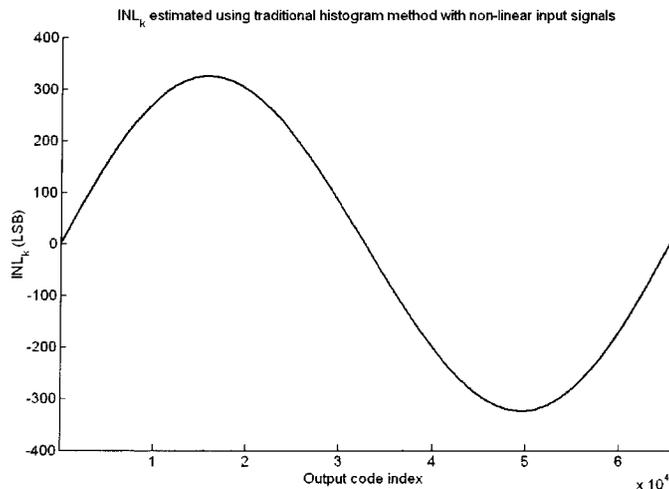


Figure 2.5  $INL_k$  of a 16-b SAR DAC with true  $INL$  1.66 LSB. Estimated using the traditional histogram method with a nonlinear stimulus.

The test time penalty of the SEIR algorithm was found to be insignificant. The actual test time for this 16-b ADC is 52 seconds, and the post-processing of the algorithm takes 1.2 seconds in Matlab to calculate all of the  $INL_k$  values from the collected bin counts. Once coded in the tester workstation, the algorithm is expected to complete well within 100 milliseconds. If a fast nonlinear source were used, the testing time performance would actually improve.

## VII. Conclusions

The SEIR method for accurate linearity testing of ADCs using nonlinear input signals has been introduced. Beyond a readily satisfied restriction that the nonlinearities of the excitation have no high spatial frequency components, no prior knowledge about the offset or nonlinearities in the input signals is required with this method. Using actual production test hardware, the SEIR method was shown to be able to test a high performance 16-bit ADC to well within  $\pm 2$  LSB *INL* specifications using only 7-bit linear inputs. The computation time required to implement this method is small and should not cause a significant degradation in test time compared to that required with existing approaches and may offer substantial reductions in test time in some applications. With the introduction of the SEIR method and corresponding extensions, the test hardware development paradigm could shift from one of highly linear source development to one of low drift and high-speed source development. The nonlinear low-drift input waveform and its shifted replica used in the proposed method can be readily generated on a DIB to reduce tester requirements and costs or on chip to support DFT or BIST features.

## References

- [1] M. Burns and G. W. Roberts, *An Introduction to Mixed-Signal IC Test and Measurement*, Oxford University Press, New York, USA 2000.
- [2] J. Doernberg, H.-S. Lee, and D. A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE J. Solid-State Circuits*, SC-19, pp.820-827, December 1984.
- [3] J. Blair, "Histogram measurement of ADC nonlinearities using sine waves," *IEEE Trans. Instrum. Meas.*, Vol. 43, pp.373-383, June 1994.

- [4] T. Kuyel, "Linearity Testing Issues of Analog to Digital Converters," in *Proc. 1999 Intl. Test Conf.*, pp. 747-756, 1999.
- [5] T. Kuyel and H. Bilhan, "Relating Linearity Test Results to Design Flaws of Pipelined Analog to Digital Converters," in *Proc. 1999 Intl. Test Conf.*, pp. 772-779, 1999.
- [6] T. Kuyel and F. Tsay, "Optimal Trim Techniques for Improving the Linearity of Pipeline ADCs," in *Proc. 2000 Intl. Test Conf.*, pp. 367-375, 2000
- [7] Y.-I. Park, S. Karthikeyan, W. M. Koe, Z. Jiang, and T.-C. Tan, "A 16-bit, 5MHz multi-bit sigma-delta ADC using adaptively randomized DWA," in *Proc. 2003 IEEE Custom Integrated Circuits Conf.*, pp. 115 – 118, Sept. 2003.
- [8] K. L. Parthasarathy, L. Jin, D. Chen, and R. L. Geiger, "A Modified Histogram Approach for Accurate Self-Characterization of Analog-to-Digital Converters," in *Proc. 2002 IEEE Intl. Sympo. Circuits Systems*, Vol. 2, pp. 376-379, 2002.
- [9] L. Jin, K. L. Parthasarathy, D. Chen, and R. L. Geiger, "A Blind Identification Approach to Digital Calibration of Analog-to-Digital Converters for Built-In-Self-Test," in *Proc. 2002 IEEE Intl. Sympo. Circuits Systems*, Vol. 2, pp. 788-791, 2002.
- [10] B. Provost and E. Sanchez-Sinencio, "Auto-Calibrating Analog Timer for On-Chip Testing," in *Proc. 1999 Intl. Test Conf.*, pp.541-548, 1999.
- [11] J.-L. Huang and et. al, "A BIST Scheme for On-chip ADC and DAC Testing," in *Proc. Design, Automation and Test in Europe Conf. & Exhib. 2000*, pp.216-220, 2000.
- [12] J. Wang, E. Sanchez-Sinencio, F. Maloberti, "Very Linear Ramp-Generators for High Resolution ADC BIST and Calibration," in *Proc. 2000 43<sup>rd</sup> IEEE Midwest Sympo. Circuits Systems*, Vol. 2, pp. 908-911, 2000.

- [13] K. L. Parthasarathy and R. L. Geiger, "Accurate Self Characterization and Correction of A/D Converter Performance," in *Proc. 2001 44<sup>th</sup> IEEE Midwest Sympo. Circuits Systems*, Vol. 1, pp. 276-279, 2001.
- [14] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*, John Wiley & Sons, Inc., 1997.
- [15] K. L. Parthasarathy, L. Jin, T. Kuyel, D. Chen, and R. L. Geiger, "A Histogram Based AM-BIST algorithm for ADC Characterization Using Imprecise Stimulus," in *Proc. 2002 45<sup>th</sup> IEEE Midwest Sympo. Circuits Systems*, Vol. 2, pp. 274-277, 2002.
- [16] K. L. Parthasarathy, L. Jin, T. Kuyel, D. Price, D. Chen, and R. L. Geiger, "Accurate Self Characterization and Correction of A/D Converter Performance," in *Proc. 2003 IEEE Intl. Sympo. Circuits Systems*, vol. 5, pp. V-537 - V-540, August 2003.
- [17] L. Jin, K. Parthasarathy, T. Kuyel, D. Chen and R. L. Geiger, "Linearity Testing of Precision Analog-to-Digital Converters Using Stationary Nonlinear Inputs," in *Proc. 2003 International Test Conference*, pp. 218 - 227, Charlotte, NC, Oct 2003.
- [18] IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters, June 2001.

## **Chapter 3**

### **SEIR Linearity Test of Precision A/D Converters in Non-Stationary Environments with Center-Symmetric Interleaving**

A manuscript submitted to the IEEE Transactions on Instrumentation and Measurement based on a paper published in the Proceedings of the 2005 International Test Conference

Le Jin, Degang Chen, and Randall Geiger

#### **Abstract**

This work describes an approach for ADC linearity testing that can tolerate environmental non-stationarity and use low-precision test signals. The effects of stimulus errors on ADC testing results will be identified and removed by exploiting the functional relationship of input signals. The effects of environmental non-stationarity will be suppressed by interleaving input signals with center-symmetric patterns. This approach can be applied to testing ADCs of very high performance, such as 16-bit or higher resolutions and more than 1 MSPS sampling rates, to which there is hardly a well-established solution for full-code testing. Simulation and experimental results show that a 16-bit ADC can be tested to 1-LSB accuracy by using input signals of 7-bit linearity in an environment with more than 100 ppm/minute non-stationarity. The proposed method can help control the cost of ADC production test, extend the test coverage of current solutions, and enable built-in self-test and test-based self-calibration.

## I. Introduction

The analog-to-digital converter (ADC) is one of the world's largest volume analog and mixed-signal (AMS) integrated circuit products and is viewed as one of the system drivers for AMS chip design [1]. Linearity test of high-performance ADCs is a well-known important and challenging problem, and the testing cost has essential meaning to the manufacturers because of the high volume. In a previous work, the authors introduced an ADC test algorithm using nonlinear signals with stimulus error identification and removal (SEIR) [2]. The authors also developed a signal generation strategy that can eliminate the effect of environment non-stationarity on the test results [3]. The combination of the two methods will provide a solution to production test of high-performance ADCs, utilizing low-linearity stimuli in a non-stationary environment. Simulation and experimental results show that the proposed technique can accurately test 16-bit ADCs using 7-bit linear signals in an environment with more than 100 ppm per minute non-stationarity. Since the combined strategy allows using nonlinear but fast signals in test, it is applicable to ADCs with a 16-bit or higher resolution and a sampling rate of more than 1 MSPS, which do not have a practical full-code test solution because of the prohibitively long test time required for conventional methods. It can also work in a test environment with stability worse than that of the application environment, but still provide accurate test results. Additionally, this approach can help reduce the test cost if combined with current solutions and enable testing of a wider range of specifications at a manageable cost.

## II. ADC Linearity Testing with SEIR

This section will briefly review the method, requirement and bottleneck of ADC linearity test and the SEIR algorithm as a solution for testing high-resolution ADCs.

### *Linearity test of high-performance ADCs*

The quasi-static linearity of ADCs is conventionally tested with the histogram approach by using a ramp or sine wave input signal with linearity at least one decade better than the specified resolution of the ADC under test [4-6]. The test of high-resolution ADCs is an increasingly challenging problem, since the resolution of ADCs is continuously going up along with the emerging demand for high-performance applications in communications, imaging, and industrial controls [7, 8]. The ADC testing capability is mainly determined by three enabling technologies: fast data capture, precision clock timing and linear stimulus generation [9]. The bottle neck of testing next generation high-performance ADCs lies in the linear signal generation, as present state-of-the-art technologies on clock timing and data capture can handle the upcoming ADCs.

A full-code histogram test for a high-resolution ADC requires a large number of samples, which implies long and expensive test time [10]. In some linearity test practices, only a reduced set of codes are tested to control the test cost. Due to long testing time, non-stationarity of the test environment will cause errors in linearity testing and become a problem too when the resolution of ADCs under test increases. A highly-linear signal source does not necessarily have good stationarity and vice versa [2]. Although the industry has spent a lot of efforts on designing and maintaining a stationary test environment, few discussions about the stationarity issue can be found in the literature. Because of these facts, there is lack of widely-adopted cost-effective approaches for testing high-performance ADCs

that are pushing the edge of current technologies. A precision linearity test can help validate the design of a high-performance ADC, reduce the number of wasted parts, and enable calibration, so it is necessary to develop methods for accurately characterizing linearity of high-resolution high-speed ADCs.

### ***SEIR algorithm***

The SEIR algorithm proposed in [2] uses two ramp signals with a constant offset  $\alpha$  to test an ADC,

$$x_1(t) = t + F(t) = t + \sum_{j=1}^M a_j F_j(t) \quad (3.1)$$

and 
$$x_2(t) = x_1(t) - \alpha, \quad (3.2)$$

where  $F_j(t)$ 's are basis functions used to parameterize the input nonlinearity  $F(t)$  with coefficients  $a_j$ 's. Feeding the two ramps into an ADC under test, we collect two sets of histogram data  $H_{k,1}$ 's and  $H_{k,2}$ 's and get two estimates for a transition level  $T_k$ ,

$$\hat{T}_{k,1} = \hat{t}_{k,1} + \sum_{j=1}^M a_j F_j(\hat{t}_{k,1}) = T_k + e_{k,1} \quad (3.3)$$

and 
$$\hat{T}_{k,2} = \hat{t}_{k,2} + \sum_{j=1}^M a_j F_j(\hat{t}_{k,2}) - \alpha = T_k + e_{k,2}, \quad (3.4)$$

where  $e_{k,1}$  and  $e_{k,2}$  are estimation errors, and transition times are estimated from the histogram data  $H_{k,1}$ 's and  $H_{k,2}$ 's as  $\hat{t}_{k,1} = \sum_{i=0}^k H_{i,1} / \sum_{i=0}^{N-1} H_{i,1}$  and  $\hat{t}_{k,2} = \sum_{i=0}^k H_{i,2} / \sum_{i=0}^{N-1} H_{i,2}$ . Taking the difference between (3.3) and (3.4) gives

$$e_{k,1} - e_{k,2} = \hat{t}_{k,1} - \hat{t}_{k,2} + \sum_{j=1}^M a_j [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] + \alpha. \quad (3.5)$$

There are  $N-1$  equations for  $k$  taking different values in (3.5) and these equations are linear in  $M+1$  unknown variables,  $a_j$ 's and  $\alpha$ . Therefore, we can robustly estimate the unknowns by using the least squares (LS) method to minimize the error energy as

$$\{\hat{a}_j, \hat{\alpha}\} = \arg \min \left\{ \sum_{k=0}^{N-2} \left[ \hat{t}_{k,1} - \hat{t}_{k,2} + \sum_{j=1}^M a_j [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] + \alpha \right]^2 \right\}. \quad (3.6)$$

With the knowledge of ramp nonlinearity  $a_j$ 's, we can remove their effects on histogram data and accurately identify ADC transition levels as

$$\hat{T}_k = \hat{t}_{k,1} + \sum_{j=1}^M \hat{a}_j F_j(\hat{t}_{k,1}). \quad (3.7)$$

Thus ADC's linearity performance can be estimated. The SEIR algorithm can use nonlinear signals for testing ADCs, with testing accuracy comparable to that of conventional methods using highly linear signals. Therefore it is promising for cost-effective production test and built-in self-test of precision ADCs.

### ***Implementation issues of SEIR algorithm***

One straightforward way to generating testing signals for the SEIR algorithm is to repeat the ramp signal  $x_1(t)$  twice and add an offset  $\alpha$  to the second one. There are two critical requirements on the test environment for using the SEIR algorithm, which are

- the same signal can be exactly repeated twice;
- the offset needs to be constant.

Violating any of the two requirements will introduce errors in the test results. Test errors due to mismatch between the two signals and due to non-constancy of the offset will be mathematically modeled and theoretically analyzed in this work, while other types of errors

related to the total number of samples, noise in the test environment and quantization errors were discussed in [2].

### **III. Non-Stationary Test Environment**

Although the test signals can be highly nonlinear in the SEIR algorithm, it is required to have two identical test signals and a constant offset. For high-precision ADC testing, changes in the test environment may cause errors that are not in agreement with the two critical requirements and will significantly degrade the testing accuracy. Causes of test environment non-stationarity include changes of temperature, humidity, and other physical environment parameters introduced by cycles of heating, ventilating and air conditioning systems, turning on and off of instruments and people walking by; changes due to aging of testing instruments; and power supply changes. It is expensive and difficult to maintain a stable environment for testing high-performance ADCs with 16-bit or higher resolutions. For the purpose of testing, the “common mode” non-stationarity that affects both the testing circuitries and the ADC does not introduce errors in linearity test results. Only the relative, “differential”, non-stationary effects among the signal generator, the offset generator, the adder and the ADC will cause errors. Typical examples of these effects in a test system include temperature difference between different functional blocks and errors introduced by internal voltage distribution networks.

A non-stationary test environment can cause errors in the two requirements and affect test accuracy of the SEIR algorithm by changing circuit electrical variables. Following are some examples. Increase of the temperature can change the reference voltage of the signal generator, which will result in different test signals  $x_1(t)$  and  $x_1'(t)$ . If ideally the signal

generator can give out linear ramps but its output is scaled by a steadily increasing reference, the true test signals are

$$x_1(t) = t(1 + \Delta \cdot t) = t + \Delta \cdot t^2$$

and

$$x_1'(t) = t(1 + \Delta \cdot t_0 + \Delta \cdot t) = t + \Delta \cdot t^2 + \Delta \cdot t_0 \cdot t,$$

where the reference voltage is normalized to 1 at the beginning of  $x_1(t)$ ,  $\Delta$  is the slope of reference drift, and  $x_1'(t)$  starts later than  $x_1(t)$  by  $t_0$ , which causes the reference voltage of  $x_1'(t)$  larger than that of  $x_1(t)$  by  $\Delta \cdot t_0$ . This leads to a non-constant difference between two signals,  $\Delta \cdot t_0 \cdot t$ . We will use  $N_1(t) = x_1(t) - x_1'(t)$  to represent the general difference between test signals. Periodical power supply fluctuation can produce a non-constant offset  $\alpha' = \alpha + \varepsilon \sin(\omega t)$ , where  $\varepsilon$  is the amplitude of the power supply-induced error and  $\omega$  is its frequency, usually equal to the frequency of power supply change. We will use  $N_2(t) = \alpha' - \alpha$  to represent the general non-constant part in the offset.

Shifting  $x_1'(t)$  down by  $\alpha'$ , we get the true second test signal as

$$x_2(t) = x_1(t) - \alpha - N(t), \quad (3.8)$$

where  $N(t) = N_1(t) + N_2(t)$  is a general error term in SEIR test introduced by environment non-stationarity that violates the two critical requirements. With  $N(t)$ , equation (3.3) and (3.4) should be rewritten as

$$\hat{T}_{k,1} = \hat{t}_{k,1} + \sum_{j=1}^M a_j F_j(\hat{t}_{k,1}) = T_k + e_{k,1} \quad (3.9)$$

and

$$\hat{T}_{k,2} = \hat{t}_{k,2} + \sum_{j=1}^M a_j F_j(\hat{t}_{k,2}) - \alpha - N(\hat{t}_{k,2}) = T_k + e_{k,2}. \quad (3.10)$$

In this case, if we neglect  $N(t)$  and apply the LS method as in (3.6), we get another estimate,

$$\{\hat{a}_j^*, \hat{\alpha}^*\} = \arg \min \left\{ \sum_{k=0}^{N-2} [\hat{t}_{k,1} - \hat{t}_{k,2}] + \sum_{j=1}^M a_j [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] + \alpha \right\}^2. \quad (3.11)$$

However, the estimates with “\*” in (3.11) are not equal to their true values because  $N(t)$  introduces errors in input identification, which can be seen from the following analysis,

$$\sum_{j=1}^M \hat{a}_j^* [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] + \hat{\alpha}^* \approx \hat{t}_{k,2} - \hat{t}_{k,1} \approx \sum_{j=1}^M a_j [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] + \alpha + N(\hat{t}_{k,2}), \quad (3.12)$$

where the first equality comes from (3.11) and the second comes from (3.9) and (3.10). Estimation error  $e_{k,1}$  and  $e_{k,2}$  are neglected as they are very small with an appropriate number of samples. The estimated coefficients can be broken down into two terms, the true value  $a_j$  and the error  $\delta_j$  contributed by  $N(t)$ , as

$$\hat{a}_j^* = a_j + \delta_j. \quad (3.13)$$

Substituting (3.13) into (3.12) and canceling identical terms on both sides, we get

$$\sum_{j=1}^M \delta_j [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] \approx N(\hat{t}_{k,2}). \quad (3.14)$$

Applying series expansion to basis functions, we can simplify (3.14) into the following form,

$$\begin{aligned} \sum_{j=1}^M \delta_j [F_j(\hat{t}_{k,1}) - F_j(\hat{t}_{k,2})] &\approx \sum_{j=1}^M \delta_j (\hat{t}_{k,1} - \hat{t}_{k,2}) \frac{d}{dt} F_j(t) \Big|_{t=\hat{t}_{k,2}} \\ &\approx -\sum_{j=1}^M \delta_j \alpha \frac{d}{dt} F_j(t) \Big|_{t=\hat{t}_{k,2}} \approx N(\hat{t}_{k,2}). \end{aligned} \quad (3.15)$$

Equation (3.15) is an approximation when the offset  $\alpha$  is much smaller than the input range of the ADC so that  $\hat{t}_{k,2} - \hat{t}_{k,1} \approx \alpha$ . Integrating (3.15) gives the input identification error as

$$F_\delta(t) = \sum_{j=1}^M \delta_j F_j(t) \approx -\frac{1}{\alpha} \int_0^t N(\tau) d\tau. \quad (3.16)$$

This error will finally become the transition level estimation error. If the estimates from (3.11) are used in (3.7) to calculate transition levels, we get

$$\hat{T}_k = \hat{t}_{k,1} + \sum_{j=1}^M \hat{a}_j^* F_j(\hat{t}_{k,1}) = \hat{t}_{k,1} + F(\hat{t}_{k,1}) + F_\delta(\hat{t}_{k,1}) = T_k + F_\delta(\hat{t}_{k,1}). \quad (3.17)$$

For instance, if  $N(t) = \Delta(t - 0.5)$ , the test error will be

$$F_\delta(t) = -\frac{1}{\alpha_0} \int_0^t \Delta(\tau - 0.5) d\tau = -\frac{\Delta}{2\alpha} (t^2 - t). \quad (3.18)$$

It gives a “bell” shape  $INL_k$  test error with a maxim absolute value occurring at the middle of the ADC input range. The input identification error, as well as the ADC transition level estimation error, is proportional to the relative error  $N(t)$  with respect to  $\alpha$ . Therefore, even if the absolute error is very small as compared to the input signal range, it can still seriously hurt the final linearity test accuracy. When designing an SEIR test, appropriately increasing the value of  $\alpha$  can reduce the effect of  $N(t)$ .

#### IV. Center-Symmetric Interleaving of Test Signals

To deal with environment non-stationarity and meet the two critical requirements of the SEIR algorithm, we propose to use an interleaving strategy in signal generation for testing high-performance ADCs. That means instead of repeating a signal twice and adding an offset to the second one, we will generate many copies of the same signal and add an offset to some of them according to a given pattern. In a low-stationarity test environment, this approach can significantly reduce the negative effects of environment changes on tests signals and equivalently generate two identical signals and a constant offset.

##### *Concept of interleaving*

Interleaving has been historically used in design and testing of electrical circuits. For designing matching-sensitive circuits that are susceptible to gradient effects, such as amplifier input stages, precision gain stages and feedback networks, the common-centroid

layout technique and extensions of this idea are ubiquitously adopted to get two or more matched electrical quantities [11, 12]. The basic idea of these approaches is to divide circuit components into many small unit cells and evenly place them on a piece of silicon such that gradient effects on electrical parameters of these components are averaged out. They can be viewed as interleaving in a two dimensional space and it is practically proven that these approaches can significantly improve the circuit performance. As a widely adopted practice, the ramp-based histogram test for ADC linearity is usually implemented by using a periodic triangular wave as the input. One of the major reasons for taking this approach is that with environment non-stationarity existing, it is easier to guarantee the linearity of each individual short and fast ramp, which could have different slopes from one to another but give an overall linear ramp, as compared to a long slow ramp. This method can effectively guarantee the accuracy of ADC linearity testing. Implementation of a ramp test signal as many short triangles prompts the use of interleaved signals in the SEIR test.

Both of the two test signals used in the SEIR algorithm are generated as triangular waves. Assume a single period of triangular wave takes the following form,

$$x_T(t) = \begin{cases} 2t + F_T(t), & 0 \leq t < 0.5; \\ 2(1-t) + F_T(t), & 0.5 \leq t < 1; \end{cases} \quad (3.19)$$

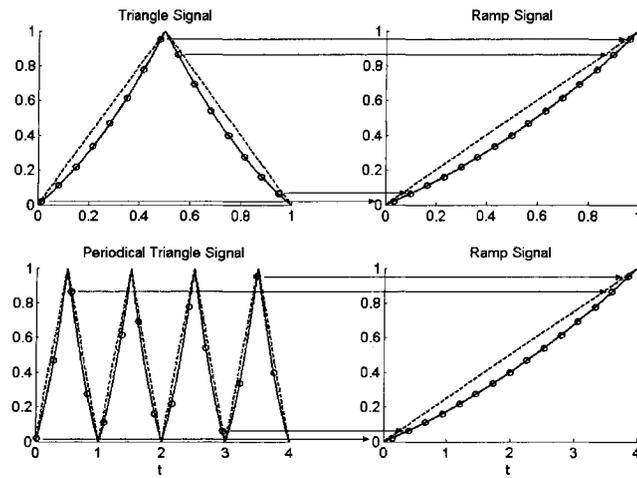
where  $2t$  and  $2(1-t)$  are the normalized desired output of the triangular wave generator for  $t < 0.5$  and  $0.5 < t$ , respectively, and  $F_T(t)$  is a general nonlinear component in the triangular wave. If we assume the input nonlinearity is a function of the desired output voltage, which is true for most generators used in quasi-static testing, the nonlinear component in (3.19) has the following property,

$$F_T(t) = F_T(1-t). \quad (3.20)$$

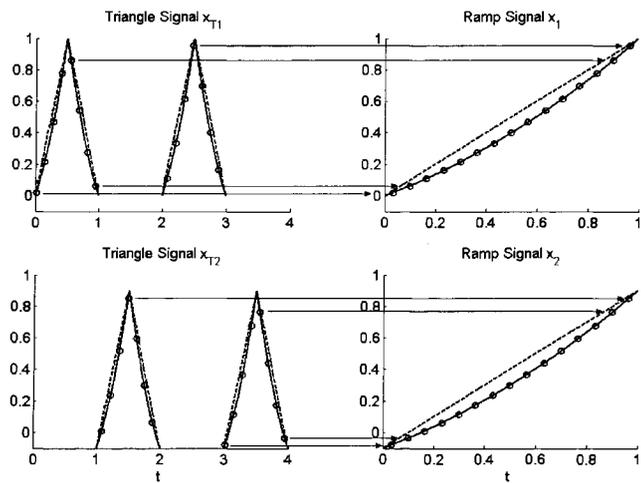
For ADC linearity testing, a ramp signal in (3.1) is equivalent to a triangle signal in (3.19), if

$$F_T(t) = F(2t), 0 \leq t \leq 0.5. \quad (3.21)$$

Let us look at the equivalence between triangle signals and ramp signals with the help of Figure 3.1. On top left of Figure 3.1(a) are an ideal triangle signal in dashed lines and a nonlinear triangle signal in solid lines, which can be described by (3.19) and its nonlinearity follows (3.20). For ADC testing, the triangle signal will be sampled at evenly spaced time instances, with samples represented by small circles. If we reorder these samples with respect to their voltage magnitudes (the vertical axis) and plot them with even spaces in time (the horizontal axis), we get a ramp signal as the top right of Figure 3.1(a) shows, while nonlinearities of the triangle signal and the ramp signal are related by (3.21). The horizontal arrows in Figure 3.1 indicate the reordering process. Since only the sampled voltages determine the ADC output codes and the timing order of samples are not important in ADC linearity testing, the triangle and ramp signals are equivalent on the sense that they give out the same histogram test result. The bottom part of Figure 3.1(a) shows the equivalence between a periodical triangle signal and a ramp signal in ADC linearity testing. Sometimes the reordering process is also called an unfolding process, and the ramp signal is called an unfolded signal, while the periodical signal is referred as folded. If the time spacing is fixed during the unfolding process, duration of the unfolded ramp is the same as that of the folded signal, 4 in the bottom part of Figure 3.1(a). As we mentioned that timing information is not critical to histogram testing, we can normalize the time axis and make the bottom ramp same as the top one, while not affecting the ADC quasi-static test result. Therefore the two ramp signals on the right of Figure 3.1(a) will give the same test result, and subsequently the periodical triangle signal is equivalent to the normalized ramp in the sense of testing.



(a)



(b)

Figure 3.1 (a) Equivalent triangle, periodical triangle and ramp signals. (b) Two nonlinear triangle signals with a constant offset in between and their ramp equivalences.

The SEIR test method requires two ramp signals with one shifted from the other by a constant offset. The two signals can not be generated and quantized by an ADC simultaneously. Generally speaking, the two desired input signals are generated as triangular

waves and interleaved with each other in the time domain. The interleaved signals can be represented as

$$s(t) = \sum_{j=0}^{2N_s-1} [x_T(t-j) - \alpha \cdot I_A(j)g(t-j)], \quad (3.22)$$

where  $I_A(j)$  is a characteristic function on an index set  $A$  that equals to 1 if  $j$  is an element of  $A$  and otherwise 0, and  $g(t)$  is a gate function that equals to 1 inside  $[0, 1]$  and vanishes outside the interval. The index set  $A$  specifies the time windows corresponding to the second signal, during which the offset will be applied. When the ADC under test is converting the signal  $s(t)$ , output codes generated during the time windows associated with  $j$ 's outside  $A$  will be counted into  $H_{k,1}$ , and codes associated with  $j$ 's in  $A$  will be counted into  $H_{k,2}$ . Based on the above discussion, the two desired test signals are

$$x_{d1}(t) = \sum_{j \notin A}^{2N_s-1} x_T(t-j), \quad (3.23)$$

and

$$x_{d2}(t) = \sum_{j \in A}^{2N_s-1} [x_T(t-j) - \alpha \cdot g(t-j)], \quad (3.24)$$

which are equivalent to (3.1) and (3.2), because samples on the two interleaved triangle signals can be unfolded and give two normalized ramp signals as shown in Figure 3.1(b). If two triangle signals are the same, two unfolded ramp signals will be the same as well and the offset between the two ramps is constant and equal to the offset between the original triangular waves.

### ***Center-symmetric interleaving in non-stationary environment***

Test environment non-stationarity will inevitably introduce errors in (3.22). Two identical signals and a constant offset as in (3.23) and (3.24) are practically impossible to achieve. Various types of non-stationary effects exist, including deterministic and stochastic time

dependent drifting, as well as random noise. Random noise usually does not degrade test performance in an unrecoverable way, since they are uncorrelated from each other and their effects can be averaged out by a reasonable number of measurements. We will focus on the non-stationary effects that have a strong correlation during a test window of about tens of seconds to several minutes. These error terms can be modeled as deterministic slowly-changing functions of time in a specific test window. And we assume these errors affect the test signal through a scaling effect in our following analysis, while additive errors can be treated in a similar style. Based on the above assumptions, the two real test signals with non-stationary effects are

$$x_{r1}(t) = \sum_{j \in \Lambda}^{2N_s-1} x_T(t-j)[1+e(t)], \quad (3.25)$$

and

$$x_{r2}(t) = \sum_{j \in \Lambda}^{2N_s-1} [x_T(t-j) - \alpha \cdot g(t-j)][1+e(t)], \quad (3.26)$$

where

$$e(t) = \sum_{i=1}^L b_i t^i + n(t) \quad (3.27)$$

represents the effects of environment non-stationarity.  $b_j$ 's are major coefficients of low-order polynomial error terms.  $n(t)$  includes noise and high-order errors which are neglected in our discussion since we assume the non-stationarity changes slowly in a test.

To meet the two critical requirements of the SEIR algorithm in non-stationary environments, we introduce a center-symmetric interleaving (CSI) strategy to cancel out the low-order error terms shown in (3.25) and (3.26). We first give the property of CSI, which is to determine the size and elements of the set  $\Lambda$ , and then explain a procedure to get such a set. The set  $\Lambda$  that can cancel up to  $L$ -th order non-stationary effects should have the property that up to the  $L$ -th

moment of  $j$ 's in  $\Lambda$  should be equal to that of  $j$ 's outside  $\Lambda$ , which can be mathematically expressed as

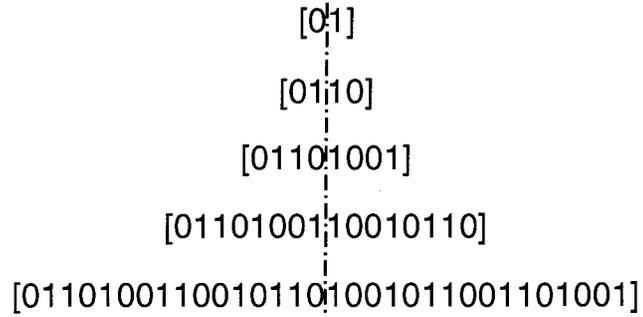
$$\sum_{j \notin \Lambda}^{2N_s-1} j^l - \sum_{j \in \Lambda}^{2N_s-1} j^l = 0, \quad l = 0, 1, \dots, L. \quad (3.28)$$

One procedure to get  $\Lambda$  is shown as follows. It starts from a pattern of one single element, [0], which will be extended to generate the elements of  $\Lambda$ . Extension of the pattern takes following steps:

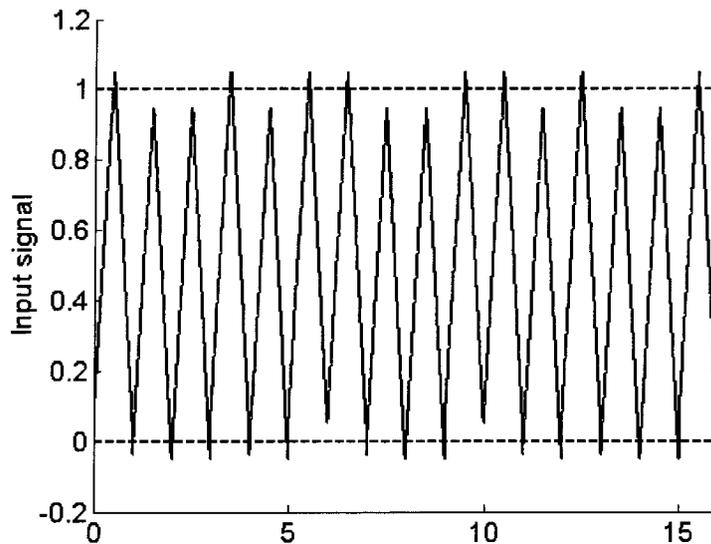
- Start from the beginning of the current level pattern.
- If meet a '0' in the current level pattern, append '01' to the next level pattern.
- If meet a '1' in the current level pattern, append '10' to the next level pattern.
- Finish at the end of the current level pattern.
- Repeat above steps with the next level pattern until the length of the pattern is equal to  $2N_s = 2 \cdot 2^L$ , where  $L$  is the level of the pattern.

The first 5 patterns are listed in Figure 3.2 (a). We name them the 0<sup>th</sup> to 4<sup>th</sup> level interleaving patterns. It can be observed that these patterns are symmetric or anti-symmetric with respect to the center as the dash-dotted line indicated, and when dividing a pattern into two in the middle, the two sub-patterns are symmetric or anti-symmetric with respect to their own centers as well. That's why we call them center-symmetric interleaving patterns, and this symmetric property leads to the capability of canceling non-stationary effects. The final pattern has a length of  $2N_s$ , because  $2N_s$  periods of triangular waves are used in an SEIR test for two signals. Index the pattern from 0 to  $2N_s - 1$ . The set  $\Lambda$  contains the indices of "1" elements in the generating pattern. The signal generated by using the 3<sup>rd</sup> interleaving pattern is plotted in Figure 3.2 (b), where a negative offset is added at  $j$  in  $\Lambda = \{1, 2, 4, 7, 8, 11, 13,$

14}. It can be verified that by using  $L^{\text{th}}$  level interleaving pattern,  $l^{\text{th}}$  moments of  $j$ 's inside and outside  $\mathcal{A}$  are equal to each other for  $l$  up to  $L$ , which is required in (3.28).



(a)



(b)

Figure 3.2 (a) First 5 patterns for  $\mathcal{A}$  generating. (b) Signal generated using the fifth pattern.

### ***Test signals with center-symmetric interleaving***

Now we show that the two test signals generated with CSI is nearly identical and the offset is nearly constant, as required by the SEIR algorithm. For investigating the effect of

interleaving on test signal generation, we first set  $\alpha=0$  and respectively take the average of triangular wave periods for the first and second signals. The two averaged signals are

$$x_1(t) = \frac{x_T(t)}{N_s} \sum_{j \in \Lambda}^{2N_s-1} [1 + e(t+j)]g(t), \quad (3.29)$$

and

$$x'_1(t) = \frac{x_T(t)}{N_s} \sum_{j \in \Lambda}^{2N_s-1} [1 + e(t+j)]g(t). \quad (3.30)$$

The difference between the averaged signals can represent the mismatch between the test signals,

$$N_1(t) = x_1(t) - x'_1(t) = \frac{x_T(t)}{N_s} \left[ \sum_{j \in \Lambda}^{2N_s-1} e(t+j) - \sum_{j \in \Lambda}^{2N_s-1} e(t+j) \right] g(t). \quad (3.31)$$

From (3.27), we get

$$\begin{aligned} & \sum_{j \in \Lambda}^{2N_s-1} e(t+j) - \sum_{j \in \Lambda}^{2N_s-1} e(t+j) \\ &= \sum_{i=1}^L b_i \left[ \sum_{j \in \Lambda}^{2N_s-1} (t+j)^i - \sum_{j \in \Lambda}^{2N_s-1} (t+j)^i \right] = \sum_{i=1}^L b_i \sum_{l=0}^i \binom{i}{l} t^{i-l} \left( \sum_{j \in \Lambda}^{2N_s-1} j^l - \sum_{j \in \Lambda}^{2N_s-1} j^l \right), \end{aligned} \quad (3.32)$$

where the choice numbers of  $i$  choose  $l$  are used and noise and high-order terms are neglected.

If  $L$ -th order CSI is used, we know that  $N_1(t) = 0$  based on (3.28) and (3.32). That means the CSI strategy cancels signal mismatch caused by up to  $L$ -th order non-stationary effects and makes two test signals nearly identical.

For studying the constancy of the offset, we set the signal to be 0 in (3.26) and take average of the real offset during time windows of the second signal,

$$\alpha' = \frac{\alpha}{N_s} \sum_{j \in \Lambda}^{2N_s-1} [1 + e(t+j)]g(t). \quad (3.33)$$

The difference between the averaged offset and the nominal offset is

$$N_2(t) = \alpha' - \alpha = \frac{\alpha}{N_s} \sum_{j \in \Lambda}^{2N_s-1} e(t+j)g(t). \quad (3.34)$$

Although  $e(t)$  is unknown, its effect on offset is attenuated by approximately  $2N_s=2 \cdot 2^L$  times if  $L$ -th level CSI is used, since each triangle only experiences a small,  $1/(2N_s)$ , portion of the total error. Assuming each signal consists of  $N_s = 32$  triangles and the signal source has 1000 ppm dominantly-linear drifting error in its reference voltage during a test, it will introduce a relative error of about 16 ppm in the offset. Based on (3.18) and the discussion thereafter, this will introduce about 2 ppm error in ADC linearity test, which is much smaller than 1 LSB at the 16-bit resolution level (16 ppm). In reality, most of the existing testing circuitries can maintain better than 1000 ppm stability during a test time of several minutes or shorter. Therefore with center-symmetric interleaving, the offset can be maintained sufficiently constant under a changing environment.

Based on above discussions, we can see that the CSI strategy can help generate identical test signals and a constant offset, two critical components required for the SEIR algorithm, under a non-stationary environment. At the end of Section IV, some conclusions are summarized and listed below.

- Two interleaved periodical triangle signals are equivalent to two unfolded ramp signals with same nonlinearity, if the original triangle signals are of the same shape.
- Offset between unfolded ramps are constant, if interleaved triangle signals are shifted from each other by a constant value.
- Identical test signals can be generated with center-symmetric interleaving. With  $L^{\text{th}}$  level CSI, effects of environmental non-stationarity on unfolded ramps can be cancelled to the  $L^{\text{th}}$  order.

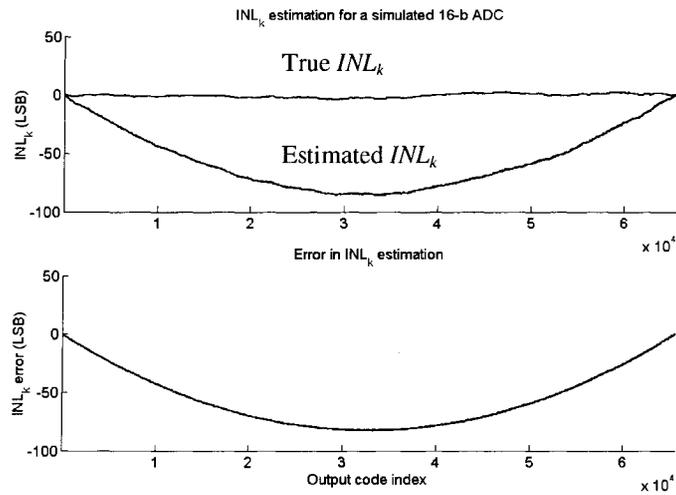
- Constant offset can be achieved with CSI. With  $L^{\text{th}}$  level interleaving, offset errors introduced by a changing environment can be reduced by about  $2^L$  times.
- By using the SEIR algorithm and  $L^{\text{th}}$  level CSI, the residue non-stationary error is limited to be of  $(L+1)^{\text{th}}$  or higher orders and in a given bound when an appropriate  $L$  is picked, if the system can be described by a well-behaved low-order function.

## V. Simulation and Experimental Results

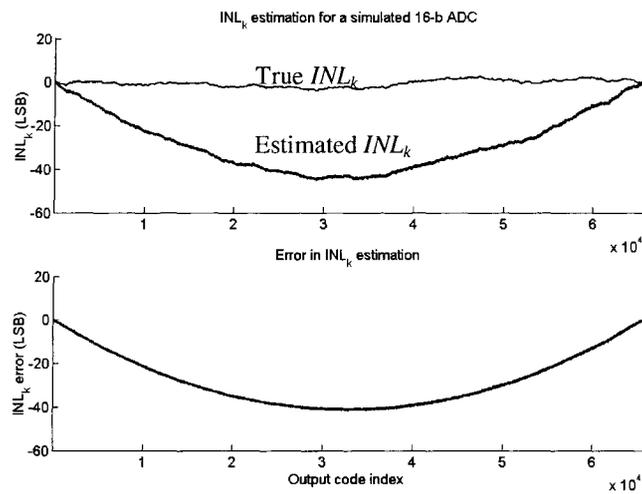
Simulations and experiments were run to validate the conclusions made earlier. They are all in agreement with the analysis on the CSI strategy and support its effectiveness.

### *Simulation Results*

The ADC under test was modeled by a 16-bit flash structure with resistance mismatch in the simulation. The SEIR test algorithm is not sensitive to the ADC architecture. We choose the flash structure because it has a large number of independent error sources so that we can validate the performance of the proposed method under challenging situations. To save simulation time, the number of average samples per code was set to 8, while it could take more than 50 samples per code in real production test applications for high-precision ADCs. In simulation, the additive noise at the ADC input had a standard deviation of 0.5 LSB. The input signals were composed of triangular waves with less than 7-bit linearity. 16 sinusoidal basis functions were used in the SEIR algorithm to identify the input nonlinearity.



(a)

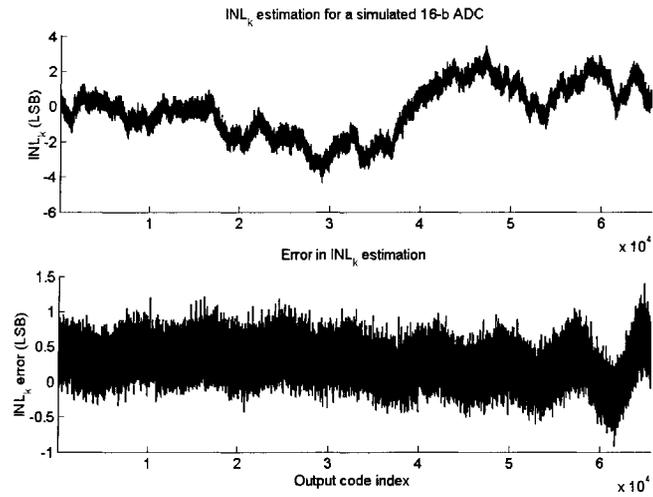


(b)

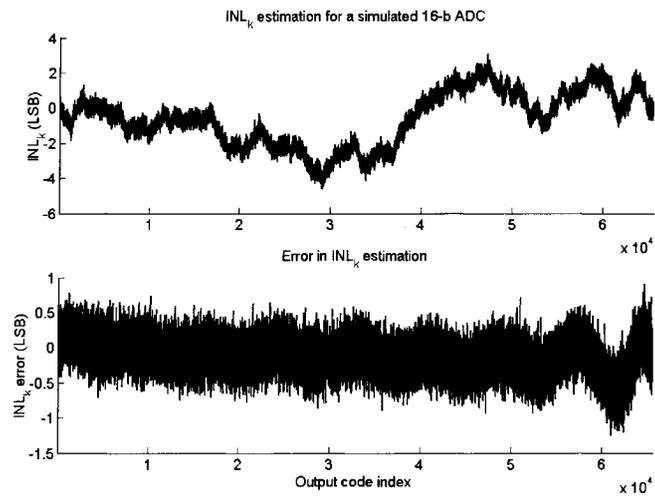
Figure 3.3  $INL_k$  estimation. (a)  $\alpha = 0.5\%$ , 100 ppm drifting. (b)  $\alpha = 1\%$ , 100 ppm drifting.

First we checked correctness of (3.16). We chose  $N_s = 8$ . The reference voltage had 100 ppm linear drifting during the test and was multiplied to the interleaved signal. To have a visible effect of the non-constant offset, the second signal was generated after the first signal completed. That means  $\Lambda = \{8, 9, 10, \dots, 15\}$ . The  $INL_k$  estimation results using the SEIR

method for a same ADC are plotted in Figure 3.3, for offset values as 0.5% and 1% of the total ADC input range, respectively. We can see that the  $INL_k$  estimation error, on the bottom in Fig. 3 (a) and (b), has a “bell” shape and the maximum error happens at the middle of the ADC input range. The maximum error is inversely proportional to offset  $\alpha$ , reduced by half when  $\alpha$  is doubled. All of these observations are in agreement with (3.18). Calculations show that the equivalent error in the offset is approximately a linear drift of 50 ppm, which is in agreement with the simulation setup, since the 100 ppm drift was applied to two test signals. Then we checked the effectiveness of CSI under the same test condition. The offset between two signals was 0.5% of the overall ADC input range. The 3<sup>rd</sup> level CSI as in Figure 3.2(b) was used. The top part of Figure 3.4(a) contains two curves, the true  $INL_k$  and the  $INL_k$  estimated using SEIR and CSI. The two curves match very well. The difference between them can hardly be seen and is plotted on the bottom part of Figure 3.4(a). The estimation error is dramatically reduced from more than 80 LSB in Figure 3.3(a) to 1 LSB. The residue errors mainly come from the noise effect due to the small number of samples. We run another simulation with 500 ppm linear drift to represent a worse test environment, and the results are shown in Figure 3.4(b). The estimation errors remain at about the same level, not increasing with the reference voltage drifting and are mainly due to noise.



(a)



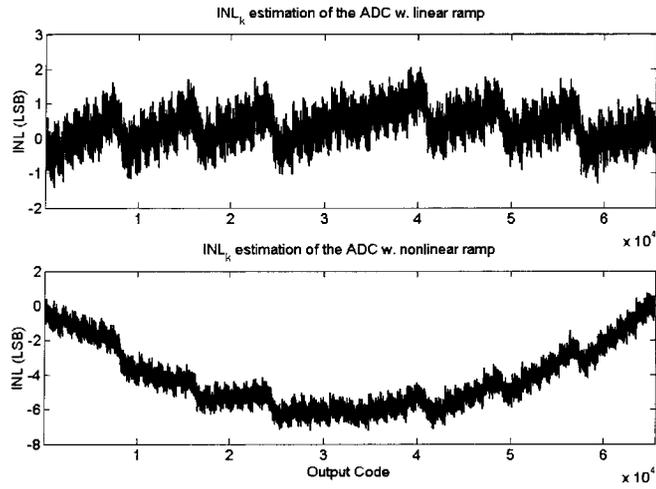
(b)

Figure 3.4  $INL_k$  estimation. (a)  $\alpha = 0.5\%$ , 100 ppm drifting. (b)  $\alpha = 0.5\%$ , 500 ppm drifting.

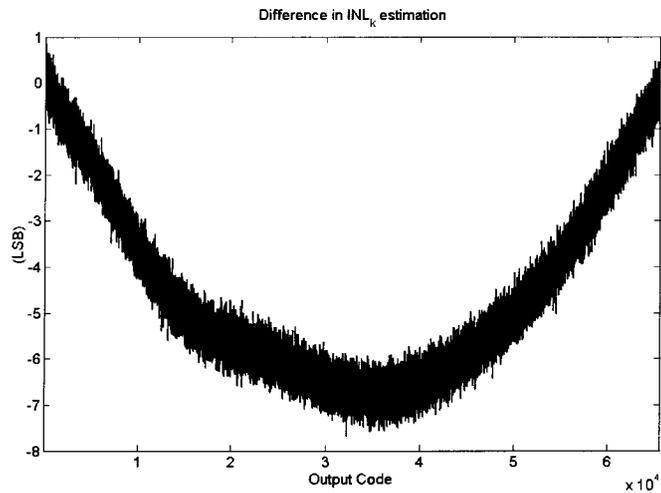
### ***Experimental Results***

Commercially available ADCs were tested to verify the performance of the proposed method combining the SEIR algorithm and the CSI strategy. The device under test was a laser trimmed 16-bit successive-approximation register ADC with typical  $INL$  of about 1.5 LSB,

which is a known test challenge.  $INL_k$  of the ADC was tested by both the traditional method and the proposed method, with 32 samples per code on average. The proposed method used 10 basis functions in input identification.



(a)



(b)

Figure 3.5  $INL_k$  measurement. Top of (a): test results with 20-bit linear signal. Bottom of (a): test results with 7-bit linear signals,  $\alpha = 0.1\%$ . (b): difference between results.

The first experiment was done without carefully arranging the two signals, simply generating the second signal after the first one. The offset was set as 0.1%. To do comparison, the ADC was first tested by using the conventional histogram method with a 20-bit linear signal and the tested  $INL_k$  is plotted on top of Figure 3.5 (a). This  $INL_k$  is used as a reference to determine the performance of the proposed test. The SEIR algorithm tested the ADC with the 7-bit linear signals and the tested  $INL_k$  is plotted on the bottom of Figure 3.5 (a). The difference between the two tested  $INL_k$  is plotted in Figure 3.5 (b). As can be seen from the plot, SEIR test results have errors of a “bell” shape and a maximum value of more than 7 LSB. This error is expected because there is non-stationarity existing in the test system.

Next the ADC was tested with an improved arrangement of signals, while all other setups are unchanged. Instead of letting the second signal go after the first one, the triangular waves of the two signals were evenly interleaved, which means  $A = \{1, 3, 5, \dots\}$ . Figure 3.6 gives the results.  $INL_k$  testes with a 20-bit linear signal is used as a benchmark and plotted on top of Figure 3.6(a). The SEIR result is on the bottom. This time test errors of the SEIR algorithm with 7-bit linear signals were reduced to about the 2-LSB level, and did not have a “bell” shape. That means evenly interleaving two signals can reduce the non-stationary effects, but it is still not good enough for testing the 16-bit part.

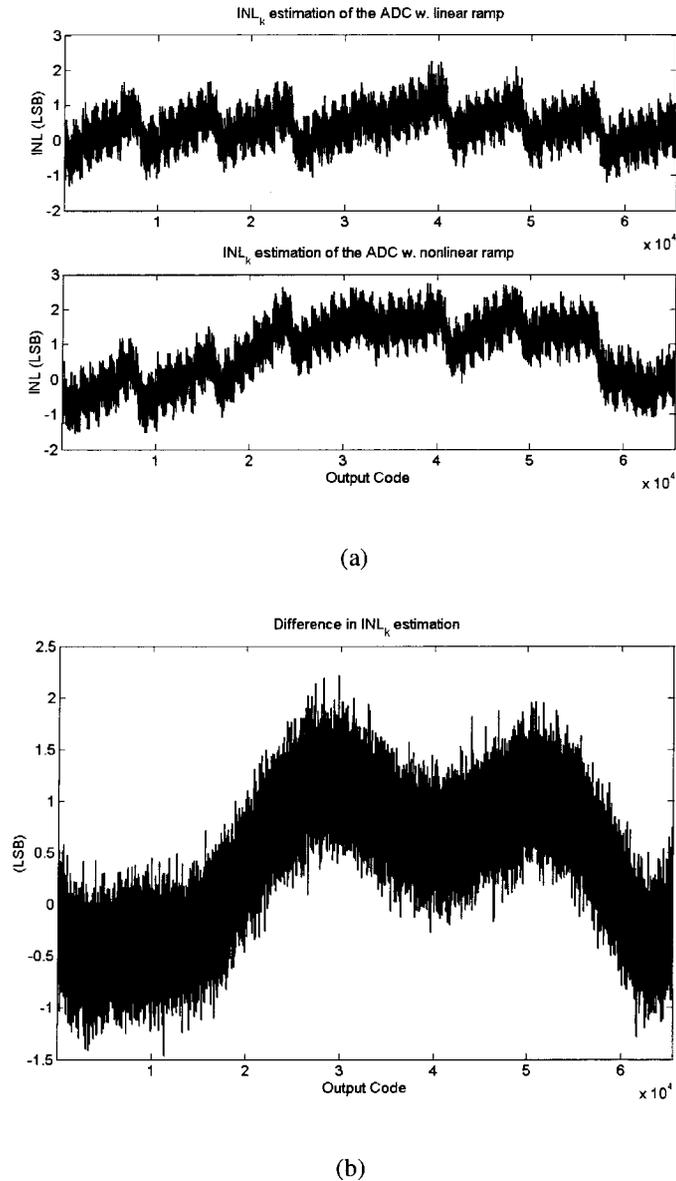


Figure 3.6  $INL_k$  measurement. Top of (a): test results with 20-bit linear signal. Bottom of (a): test results with 7-bit linear signals,  $\alpha = 0.1\%$ . (b): difference between results.

Finally the CSI method developed in this work was used. This time the DC offset was set to a smaller value, 0.05%, which may lead to larger test errors as compared to the 0.1% offset.  $N_s$  was set to 32 and  $A = \{1, 2, 4, 7, 8, 11, 13, 14, 16, 19, 21, 22, 25, 26, 28, 31, 32, 35, 37, 38, 41,$

42, 44, 47, 49, 50, 52, 55, 56, 59, 61, 62} was used for canceling up to the fifth order reference drifting errors. The test results are plotted in Figure 3.7. As our old convention, the ADC was first tested by using a highly linear signal with the results plotted on top of Figure 3.7(a). The corresponding measured  $INL$  is 1.66 LSB. Then histogram data were obtained with the 7-bit linear input signals and analyzed using the SEIR algorithm with 10 basis functions. The estimated  $INL_k$  is plotted on bottom of Figure 3.7(a). The estimated  $INL$  with nonlinear signals is 1.77 LSB. We can see that when using the proposed CSI method, the  $INL_k$  estimated using linear and nonlinear signals are really close. The difference between the  $INL_k$  estimates is shown in Figure 3.7 (b) and they are mostly less than 1 LSB. That means the proposed approach can effectively make the offset between two signals a constant in non-stationary environments and is an acceptable solution as far as 16-bit converters are concerned.

In Figure 3.7 (b), the difference between testing results mainly come from two sources. First, the high frequency errors were introduced by the additive noise, which gives a band of about  $\pm 0.5$  LSB. Second, the low frequency error component was contributed by the residue of non-constant errors in the offset, which are not completely cancelled by the signal arrangement approach. Based on (3.16), this kind of error is inversely proportional to the offset amount. Since the offset value is only 0.05% in the last test, we can increase the offset amount to further reduce test errors to well below the noise error level.

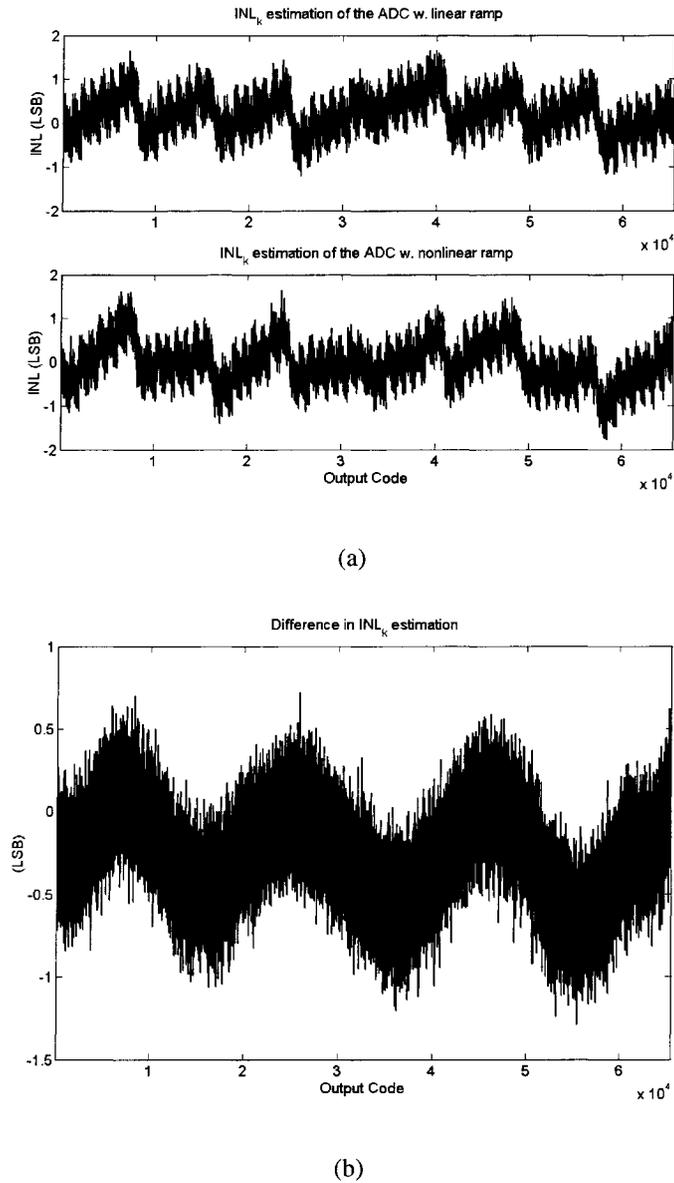


Figure 3.7  $INL_k$  measurement. Top of (a): test results with 20-bit linear signal. Bottom of (a): test results with 7-bit linear signals,  $\alpha = 0.05\%$ . (b): difference between results.

The simulation and experimental results presented in this section show that combining the SEIR algorithm with the CSI technique can eliminate the effect of test environment non-stationarity and give out accurate linearity test results for high-resolution ADCs.

## VI. Conclusions

In this paper, a test strategy that can eliminate the effects of input nonlinearity and environment non-stability errors on the test results of high-resolution ADCs is introduced. Using the SEIR algorithm along with the proposed CSI technique, 16-bit ADCs were accurately tested with only 7-bit linear input signals in an environment with more than 100 ppm non-stationarity in the test window. This strategy is promising to solve test problems that are very challenging, such as full characterization of ADC linearity at more than 16-bit resolution levels, since both the signal linearity and environment stability are no longer required to be better than ADC specifications. This strategy can also help control the cost of existing test solutions by allowing the use of cheap instruments. Furthermore, the strategy has the potential to be used in on-chip test environment, where accurate test devices may not be available.

## References

- [1] *International Technology Roadmap for Semiconductors*, 2003 edition, available online: <http://public.itrs.net>.
- [2] L. Jin, K. Parthasarathy, T. Kuyel, D. Chen and R. Geiger, "Accurate testing of analog-to-digital converters using low linearity signals with stimulus error identification and removal," *IEEE Trans. Instrum. Meas.*, vol. 54, pp. 1188 – 1199, June 2005.
- [3] L. Jin, K. Parthasarathy, T. Kuyel, R. Geiger and D. Chen, "High-Performance ADC Linearity Test Using Low-Precision Signals in Non-Stationary Environments," in *Proc. 2005 International Test Conference*, Nov 2005.

- [4] J. Doernberg, H.-S. Lee, and D.A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE J. Solid-State Circuits*, SC-19, pp. 820-827, December 1984.
- [5] J. Blair, "Histogram measurement of ADC nonlinearities using sine waves," *IEEE Trans. Instrum. Meas.*, vol. 43, pp. 373-383, June 1994.
- [6] IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters, Dec 2000.
- [7] E. Siragusa and I. Galton, "A Digitally Enhanced 1.8-V 15-Bit 40-MSample/s CMOS Pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 39, pp 2126- 2138, Dec 2004.
- [8] A. Roy, S. Sunter, A. Fudoli, and D. Appello, "High Accuracy Stimulus Generation for A/D Converter BIST," in *Proc. 2002 International Test Conference*, pp. 1031-1039, Oct. 2002.
- [9] T. Kuyel, "Linearity Testing Issues of Analog-to-Digital Converters," in *Proc. 1999 International Test Conference*, pp. 747-756, Sept 1999.
- [10] S. Bernard, M. Comte, F. Azais, Y. Bertrand, and M. Renovell, "A New Methodology for ADC Test Flow Optimization," in *Proc. 2003 International Test Conference*, vol. 1, pp. 201-209, Sept 2003.
- [11] A. Hastings, *The Art of Analog Layout*, Prentice Hall, New Jersey, 2000.
- [12] X. Dai, C. He, H. Xing, D. Chen, and R. Geiger, "An N<sup>th</sup>-Order Central Symmetrical Layout Pattern for Nonlinear Gradients Cancellation," in *Proc. 2005 IEEE Intl. Sympo. Circuits and Systems*, pp. 4835 – 4838, May 2005.

## **Chapter 4**

# **Code-Density Test of Analog-to-Digital Converters Using Single Low-Linearity Stimulus Signal**

A paper submitted to the 2006 International Test Conference

Le Jin, Degang Chen and Randall Geiger

### **Abstract**

High-precision ADC testing is a challenging problem because of its stringent requirement on test signal's linearity. This work introduces a histogram-based method using a single nonlinear stimulus signal for testing linearity of some widely used high-resolution ADCs, including cyclic and pipeline architectures. The proposed algorithm exploits the architecture information in data processing. It first identifies and removes the input errors using the series expansion and least squares method, and then accurately measures ADC linearity. Simulation and experimental results show that 16-bit ADCs can be tested to 1 LSB accuracy by using a 7-bit linear signal. This approach provides a solution to both the production and on-chip testing problems of high-resolution ADCs, since it does not require high-linearity stimulus signals and its computational complexity is reasonably low.

### **I. Introduction**

Consumer electronic devices are integrating more and more functions, such as mixed-signal and RF, into a single system. This trend reflects the society's persistent demand for high

quality of life. It is enabled by the fast-evolving semiconductor technologies, and meanwhile introduces new challenges to the whole IC industry. As the digital technologies are no longer solving all the problems, new needs in design and test of analog, mixed-signal and RF circuits are emerging. Novel technology revolutions such as system-on-a-chip (SoC) design style and built-in self-test (BIST) are proposed to take advantage of the rapidly-growing integration possibility. Along with the production of very large scale digital ICs, testing of digital circuits has been systematically studied for decades and many mature methodologies were developed. On the other side, because the analog and mixed-signal circuit design has never received as much attention as digital design, testing of analog and mixed-signal (AMS) circuits is still in its development stage. Many methods proposed in 1980's or even earlier are still being used. Nowadays the analog test cost of a mixed-signal device can take a significant portion of its total production cost and this ratio is expected to become even higher in the future [11]. The *International Technology Roadmap for Semiconductors (ITRS)* identified mixed-signal testing as one of the most daunting SoC challenges.

The analog-to-digital converter (ADC) is one of the world's largest volume analog and mixed-signal (AMS) integrated circuit products and is viewed as one of the system drivers for AMS chip design [1]. Linearity test of high-performance ADCs is a well-known important and challenging problem. A precision linearity test can help validate the design of a high-performance ADC, reduce the number of wasted parts, and enable calibration, so it is necessary to have methods for accurately characterizing linearity of high-resolution high-speed ADCs. Also the testing cost has essential meaning to the manufacturers because of the high volume.

The ADC testing capability is mainly determined by three enabling technologies: fast data capture, precision clock timing and linear stimulus generation [8]. The bottle neck in testing of next generation high-performance ADCs is the linear signal generation, as the present state-of-the-art technologies on timing and data capture can handle the testing need of upcoming ADCs. The code-density test method [2, 3] is widely adopted for testing ADCs' static linearity in the industry, because its implementation is straightforward and its computational complexity is low. The code-density method uses a ramp or sine wave as the stimulus signal, with linearity at least one decade better than the specification of the ADC under test. The input linearity requirement makes the test of high-resolution ADCs an increasingly challenging problem, since the resolution of ADCs is continuously going up along with the emerging demand for high-performance applications in communications, imaging, and industrial controls [12, 13]. Furthermore, a full-code histogram test for a high-resolution ADC requires a large number of samples, which implies long and expensive test time [14]. Due to long testing time, the nonstationarity of the test environment will cause errors in linearity testing as well. A highly-linear signal source does not necessarily have good stationarity and vice versa [5]. Because of the above facts, there is lack of widely-adopted cost-effective approaches for testing high-performance ADCs that are pushing the edge of current technologies.

In addition to the approaches for testing ADC linearity by using analog and mixed-signal automated test equipments (ATEs), other test approaches have been developed for measurement of ADC linearity. An on-chip ramp generator was designed and achieved 11-bit linearity [15]. However, for testing ADCs with 16-bit or higher resolutions by using the conventional histogram method, more than 20-bit linear signals are needed. To the best of the

authors' knowledge, none of the existing on-chip signal generators can fulfill this requirement. This is why there is little industrial adoption of BIST techniques.

If the requirement on highly linear signals can be removed, the testing problem for high-speed high-resolution ADCs becomes much more tractable. It can be shown that one nonlinear signal is generally insufficient for testing an ADC. At 2003 and 2005's International Test Conference, the authors presented an ADC testing method using two nonlinear signals with a stimulus error identification and removal (SEIR) algorithm [5], and a strategy that minimizes the effect of environment nonstationarity on the test results [6]. The combination of the two methods provides a solution to production test of high-performance ADCs, utilizing low-linearity stimuli in a realistic time-varying environment. This methodology is a general I/O based testing method. It takes the ADC under test as a black box and does not make use of any information on the ADC architecture.

Our investigation shows that exploiting extra knowledge on the ADC architecture can help further simplify the SEIR testing algorithm. This work introduces such an approach using a single nonlinear stimulus signal for testing ADCs with some widely adopted architectures, including cyclic and pipeline. It first identifies and removes the input errors based on the ADC's structure characteristics, and then accurately measures the ADC's linearity. Simulation and experimental results show that 16-bit ADCs can be tested to 1 LSB accuracy by using a 7-bit linear signal. This approach provides a solution to both the production and on-chip testing problems of high-resolution ADCs, since it does not have stringent requirement on the stimulus signal and its computational complexity is reasonably low.

## II. Characteristics of Pipeline ADCs

Among different ADC architectures, the pipeline ADC is a balanced combination of the speed, accuracy, and power consumption. It has very wide applications in communications, CCD imaging, and data acquisition. A general block diagram of the pipeline ADC is drawn in Figure 4.1 [7]. It consists of a front-end sample-and-hold (SHA) circuit,  $k$  conversion stages, and some digital circuits for output code generation.

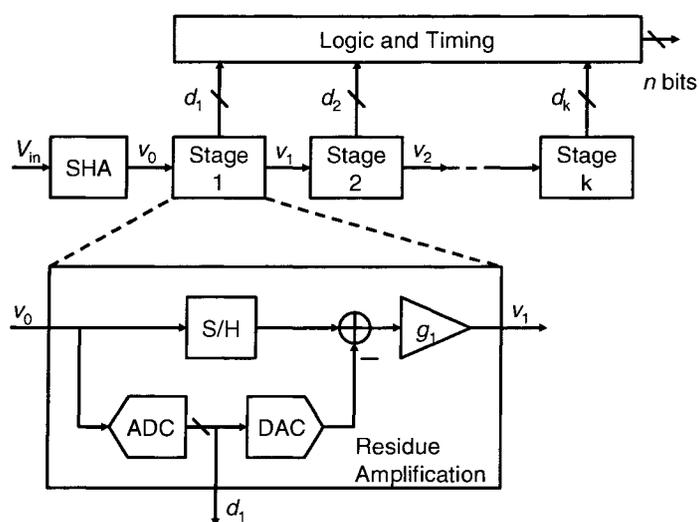


Figure 4.1 Block Diagram of an  $n$ -bit Pipeline ADC.

### *Conversion Stage Modeling*

The sampled input voltage,  $v_0$ , will be quantized by a low-resolution sub-ADC in the first stage, and the residue is amplified and sent to the second stage. The subsequent cascaded conversion stages will process the residue of the previous stage in a similar way. The output codes of these stages will be appropriately assembled to give a quantized value of the input with very high accuracy. Although the pipelined stages will introduce latency between the

input and the output, the quantization speed is determined by a single stage and can be very fast.

Different sources contribute to the final errors in the ADC, including distortion of the front-end SHA circuit, mismatches and linearity errors in the sub-ADCs and sub-DACs, gain error and nonlinearity of the residue amplifier, and so on. Some of the error sources do not cause significant problems in practical applications. For instance, the existing front-end SHA circuits can have very low distortion at extremely high speed, especially for CCD imaging applications where input voltages are static, so they are usually viewed as linear. Other error sources will cause problems such as non-monotonicity, missing codes, and nonlinear transfer characteristics. Sub-radix design and self-calibration techniques are suggested for taking care of them to achieve high-performance [10].

Common terminologies for ADC parameters and its linearity specifications used in the paper will follow the definitions in [3]. Using the one-bit-per-stage architecture as an example, the transfer function of the first conversion stage of a pipeline ADC can be summarized into a mathematical model as

$$d_1 = \begin{cases} 1, & v_{o1} \leq V_{in}; \\ 0, & V_{in} < v_{o1}, \end{cases} \quad (4.1)$$

and

$$v_1 = \begin{cases} g_1(V_{in} - v_{p1}), & d_1 = 1; \\ g_1(V_{in} - v_{n1}), & d_1 = 0, \end{cases} \quad (4.2)$$

where  $V_{in}$  is the input voltage,  $d_1$  is the one-bit digital output of the first stage,  $v_1$  is the residue of the first stage and the input to the second conversion stage,  $v_{o1}$  is the offset voltage,  $v_{p1}$  and  $v_{n1}$  are input voltages generating a zero residue, and  $g_1(\cdot)$  is a transfer function

representing nonlinear effects of the first stage's residue amplifier. It is assumed the front-end SHA is linear, so  $v_0 = V_{in}$  in Figure 4.1.

Ideally, the offset voltage is zero, and the input voltage equal to  $\pm V_{ref}/2$  should generate a residue voltage equal to zero. Mismatch errors will cause them to deviate from the desired values. The transfer function  $g_1(\cdot)$  is supposed to be a straight line, but the actual gain will drop as the output voltage increases [9]. Figure 4.2 shows a realistic residue transfer curve, where dashed lines compose the ideal curve, and the errors in the offset, zero-residue input voltages and the amplifier are exaggerated for a better visual effect.

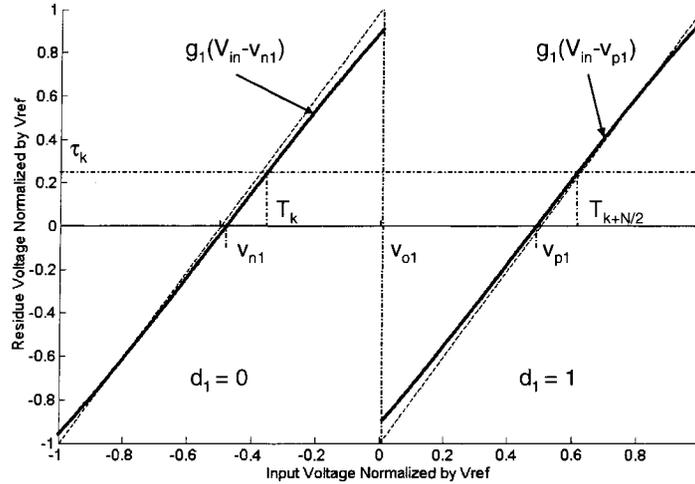


Figure 4.2 Transfer curve of residue amplification.

### ***Transition Level Characterization***

As we discussed earlier, the residue voltage of the first stage,  $v_1$ , will be quantized by the following  $2^{\text{nd}}$  to  $k$ -th conversion stages. To  $v_1$ , these stages work as an  $n-1$  bit sub-ADC with transition levels  $\tau_k$ ,  $k=1, 2, \dots, N/2-1$ , where  $N=2^n$ . It is easy to show that when  $v_1$  goes across a specific  $\tau_k$ , corresponding  $V_{in}$  crosses a transition level of the overall ADC  $T_k$ ,

because a change in the output of the  $2^{\text{nd}}$  to  $k$ -th stages is obviously a change in the output of the whole ADC. Since the first stage's output can have 0 and 1 two values,  $v_1 = \tau_k$  is associated with two input voltages  $V_{in} = T_k$  and  $T_{k+N/2}$  that are smaller and larger than  $v_{o1}$ , respectively, as marked with dash-dotted lines in Figure 4.2. Substituting these relations into (4.2), we get

$$\tau_k = g_1(T_k - v_{n1}), \quad d_1 = 0, \quad (4.3)$$

and

$$\tau_k = g_1(T_{k+N/2} - v_{p1}), \quad d_1 = 1, \quad (4.4)$$

for  $k$  from 1 to  $N/2 - 1$ . Taking the inverse function of  $g_1(\cdot)$  on above equations gives

$$g_1^{-1}(\tau_k) = T_k - v_{n1} = T_{k+N/2} - v_{p1}. \quad (4.5)$$

We can rewrite the second equality in (4.5) as

$$T_{k+N/2} - T_k = v_{p1} - v_{n1}. \quad (4.6)$$

It means that the difference between  $T_{k+N/2}$  and  $T_k$  is a constant  $v_{p1} - v_{n1}$ .

Equation (4.6) is a general relationship between the upper half transition levels,  $T_k$  with  $k > N/2$ , and lower half transition levels,  $T_k$  with  $k < N/2$ , of a pipeline ADC. It is applicable to all  $k = 1, 2, \dots, N/2-1$ . This result is not surprising because the residue amplification and quantization for input voltages smaller and larger than the offset  $v_{o1}$  are identical, except for the constant voltage shift  $v_{n1}$  and  $v_{p1}$ . Based on the above observation, we can conclude that (4.6) is also a correct description for many other popular architectures such as cyclic ADCs and successive approximation register (SAR) ADCs with a binary-weighted internal DAC, whose transfer function are repeated for small and large input voltages. Therefore, the

algorithm that will be developed in the following section is applicable to these types of ADCs as well.

### III. ADC Testing with Single Nonlinear Stimulus Signal

We are going to introduce an algorithm uses a nonlinear signal to test high resolution ADCs with pipeline, cyclic or other similar architectures. The input nonlinearity will be first identified based on the ADC's inherent property. Then the linearity performance of the ADC itself can be accurately estimated.

#### *Input Signal Modeling and Histogram Test*

The histogram method uses a stimulus signal to excite the ADC under test and characterize its performance based on the code-density information. Ramp (triangular) signals and sine waves are often used as the stimulus, for simplicity of the conventional testing algorithm [2, 3]. The conventional algorithm requires that the input signal should have linearity one decade better than the resolution of the ADC to guarantee accurate test results, or the error of the input signal will be mistakenly identified as part of the ADC's error [5]. This requirement on input linearity is very challenging for high-resolution ADCs.

We model the input signal  $s(t)$  as a linear ramp plus a nonlinear term  $F(t)$ ,

$$s(t) = t + F(t). \quad (4.7)$$

The amplitude and the offset of the input signal do not directly affect the linearity test results, so the linear component's coefficient is normalized to one and the offset assumed to be zero in (4.7). Testing the ADC with this input signal, we can get a set of histogram count  $H_k$ 's for code  $k$  from 0 to  $N - 1$ . The transition levels of the ADC can be estimated using the histogram counts as

$$\hat{T}_k = t_k + F(t_k) \quad (4.8a)$$

$$= T_k + e_k, \quad (4.8b)$$

where

$$t_k = \frac{\sum_{j=0}^{k-1} H_j}{\sum_{j=0}^{N-1} H_j} \quad (4.9)$$

in (4.8a) is the measured time at which the output code's transition between  $k - 1$  and  $k$  happens, and (4.8b) explicitly gives the estimation error  $e_k$ . However, the nonlinear component  $F(t)$  is unknown, so (4.8a) cannot practically give an estimated value of  $T_k$ . If we assume the test signal is linear,  $F(t)$  will be taken as ADC nonlinearity and cause errors in the test result. If we would like to use a nonlinear signal in ADC testing, we have to accurately identify  $F(t)$  first.

### ***Test Using Single Nonlinear Signal***

Based on our discussion in Section II, we know that the difference between the true  $T_{k+N/2}$  and  $T_k$  is a constant. So the difference between the corresponding estimated values of them should be a constant as well with some estimation error effects. Plugging (4.8b) into (4.6) gives

$$\hat{T}_{k+N/2} - \hat{T}_k = v_{p1} - v_{n1} + e_{k+N/2} - e_k. \quad (4.10)$$

Substituting (4.8a) into (4.10), we get

$$t_{k+N/2} - t_k = F(t_k) - F(t_{k+N/2}) + v_{p1} - v_{n1} + e_{k+N/2} - e_k. \quad (4.11)$$

If the difference between  $t_{k+N/2}$  and  $t_k$  is not a constant, it is because of the input nonlinearity. In [5], we have successfully shown that series expansion over a set of basis functions and the least squares (LS) method can be used to identify the input error in two identical nonlinear

signals with a constant offset in between. We will use these techniques again in this paper to estimate the input error in a single stimulus signal using the ADC's architecture property.  $F(t)$  can be expanded as

$$F(t) = \sum_{j=1}^M \alpha_j F_j(t), \quad (4.12)$$

where  $\{F_j(t), j=1, 2, \dots, M\}$  is a set of complete functions, and  $\alpha_j$ 's are unknown coefficients. This expansion can be used in (4.11). With some simple re-ordering operations, we can write (4.11) as

$$e_{k+N/2} - e_k = t_{k+N/2} - t_k + \sum_{j=1}^M \alpha_j [F_j(t_{k+N/2}) - F_j(t_k)] - \Delta v, \quad (4.13)$$

where  $\Delta v = v_{p1} - v_{n1}$ . The left hand side of (4.13) contains only error terms. Ideally they should be zero. We have an abundant number of,  $N/2 - 1$ , equations like (4.13) for  $k$  from 1 to  $N/2 - 1$ , because  $N$  can be as large as tens of thousands for high-resolution ADCs. Usually the input nonlinearity can be concisely described by a reasonable small number of basis functions, so  $M$  is much smaller than  $N/2 - 1$ . So we have plenty of equations linear in a small number of variables. Therefore, the LS method can be used to estimate  $\alpha_j$ 's and  $\Delta v$ . If we are going to minimize the target function of the total energy of estimation errors, the corresponding estimated values are [16],

$$\{\hat{\alpha}_j's, \Delta \hat{v}\} = \arg \min \left\{ \sum_{k=1}^{N/2-1} \left( t_{k+N/2} - t_k + \sum_{j=1}^M \alpha_j [F_j(t_{k+N/2}) - F_j(t_k)] - \Delta v \right)^2 \right\}. \quad (4.14)$$

Details of the LS algorithm are provided in Section VI. Substituting these estimations into (4.8a), we can now get the ADC transition levels as

$$\hat{T}_k = t_k + \sum_{j=1}^M \hat{\alpha}_j F_j(t_k), \quad (4.15)$$

for all  $k$  from 1 to  $N - 1$ . Based on these estimated transition levels, we can calculate the linearity specifications of interest, such as INL and DNL, according to their definitions. Because the input nonlinearity is identified and removed, the linearity test result has very high accuracy.

### *Discussions*

Equation (4.8) is valid and consistent with (4.7), if the input signal is monotonic. This condition can be achieved by most existing signal generator structures. It is reasonable to assume that the monotonicity will not be violated if the signal is allowed to be nonlinear. Actually the proposed algorithm doesn't require the input signal to be monotonic, since the histogram-based method is essentially working with the cumulative distribution function (CDF) of the input voltage, which is inherently monotonic. But we will make the monotonicity assumption in this paper, only for convenience.

The sampling instances need to be uniformly distributed along the time axis for (4.9) to be valid. This is assumed to be true in the conventional histogram test and the proposed method. The current timing techniques can usually achieve very high accuracy at sampling speeds of pipeline and cyclic ADCs with medium and high-resolutions. For low-resolution ADCs with very high sampling rate, timing jitter may cause significant problems, but they are out of the scope of this paper. Combining the assumptions of a monotonic signal and uniformly distributed sampling in time, the shapes of the CDF of the input voltage and its time domain functional form are identical, neglecting the additive noise in the input signal. This makes our analysis in this work applicable to both cases.

The offset voltage of the first stage determines the transition level of the most significant bit (MSB),  $T_{N/2}$ , which is not used in the algorithm above. The offset does not directly affect the values of other transition levels, but some transition levels may not appear if the offset is too big. Therefore, the summation in (4.14) should only be applied to those  $k$ 's for which both  $T_k$  and  $T_{k+N/2}$  are detected during the histogram test. Since  $N$  is much larger than  $M$ , there will still be enough number of equations for the LS method to accurately identify the input nonlinearity.

#### IV. Simulation Results

A behavioral level model of pipeline ADCs was built in simulation for validating the performance of the proposed algorithm. The model includes the gain error, mismatch errors, amplifier nonlinearity, and other common nonideal effects in a real integrated circuit. 16-bit ADCs generated with this model have about 4 LSB INL, which is a reasonable value for current products. A noise with 2-LSB standard deviation was added to the test signals in simulation. This will introduce a noise band of larger than  $\pm 6$  LSB, which is sufficient to represent the noise in a typical test environment. The total number of samples is equivalent to about 16 samples per code on average.

##### *Single ADC Testing*

True  $INL_k$  of a simulated 16-bit ADC is plotted on the top of Figure 4.3. It was measured with an ideal clean ramp signal without any noise. The true INL is 3.41 LSB. The simulated ADC was tested with a 6-bit linear ramp signal. The nonlinear error of the test signal was first characterized using the parameterization and LS algorithm introduced in Section III. The  $INL_k$  of the ADC was then calculated and plotted on the bottom of Figure 4.3. The estimated INL is 3.62 LSB, which is very close to the true value. The true and tested  $INL_k$  curves match

very well to each other and the difference between them is given in Figure 4.4. All the  $INL_k$  test errors are less than 1 LSB at the 16 bit level. This confirms that proposed algorithm can accurately estimate and remove the input error and test 16-bit ADCs' performance using a single nonlinear signal with only 6-bit linearity.

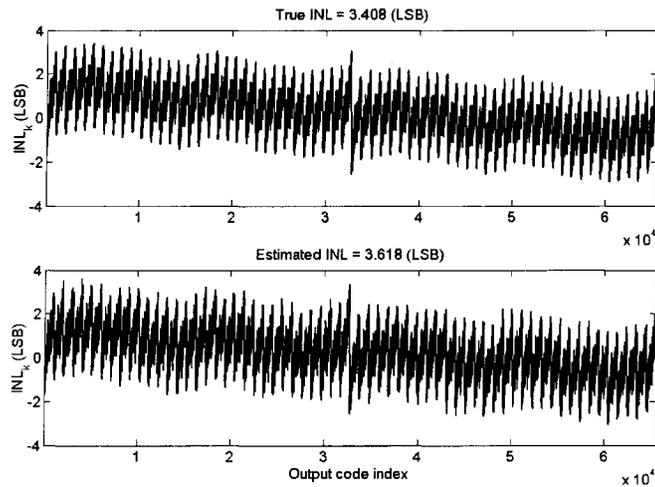


Figure 4.3 True and estimated  $INL_k$ .

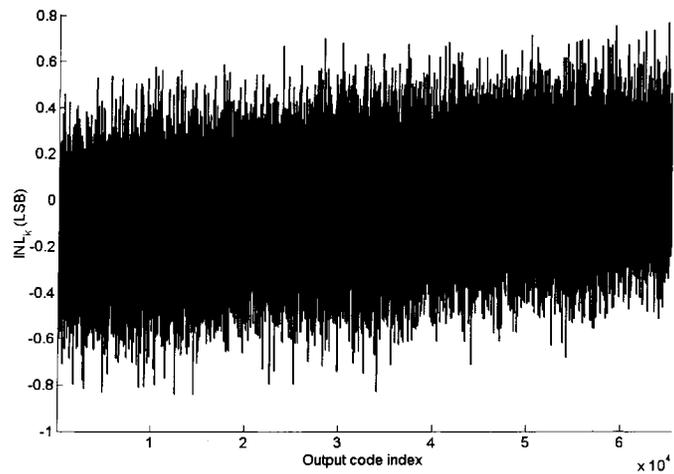


Figure 4.4 Difference between  $INL_k$  measurement results using ideal and nonlinear ramps.

Low-order even and odd nonlinear components are synthesized for the input ramp. The stimulus signal has more than 1000 LSB errors at the 16-bit level, as shown in Figure 4.5. This is an exaggerated nonlinearity for validating the performance the proposed strategy. Signals with much better linearity can be easily generated in reality.

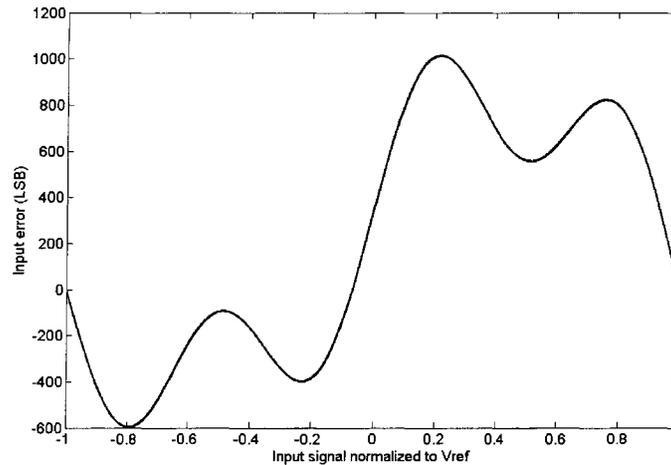


Figure 4.5 Input error of the nonlinear ramp signal.

### ***Multiple ADC Testing***

The simulation was repeated another 64 times to better characterize the performance of the algorithm. For saving simulation time, 14-bit pipeline ADCs were used as the device under test in these runs. Additive noise has a standard deviation of 2 LSB at the 14-bit level. A sample density of 16 hits per code was still used to reduce the noise and quantization effects. The estimated INL using the single 6-bit linear signal are plotted with respect to the true values in Figure 4.6. The true INL values of the simulated ADCs range from 2 to 5 LSB. It can be observed that pairs of measured and true INL values are very well distributed along the 45 degree line in the figure. That means the estimated INL values accurately track the

true values for ADCs with different linearity performance. The residue INL estimation errors are given in Figure 4.7 for the 64 ADCs. All of the errors are less than 0.5 LSB. These statistical data show that the proposed algorithm can consistently test high-resolution ADCs using a single low-linearity signal and is robust to different ADC performance levels.

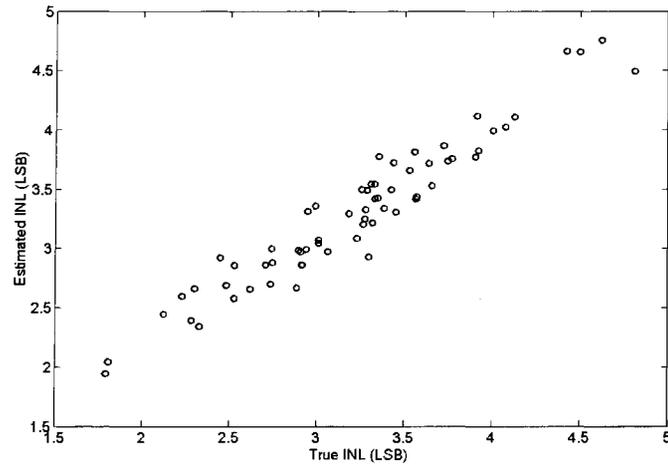


Figure 4.6 Measured INL vs. True INL.

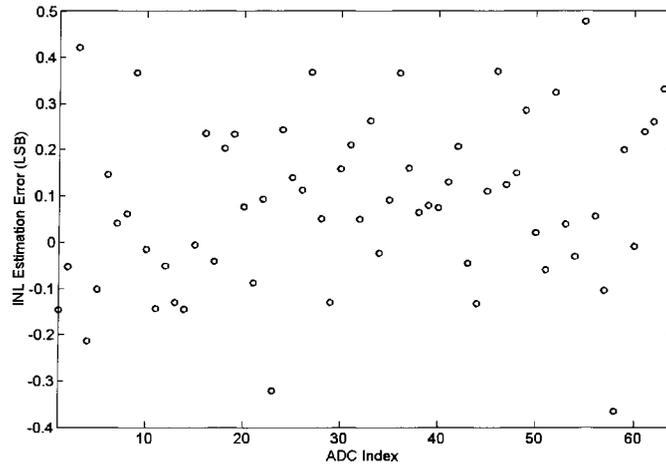


Figure 4.7 Residue INL estimation errors.

## V. Experimental Results

Data on 16-bit commercial ADCs collected at Texas Instruments are presented below to validate the performance of the proposed algorithm. The ADCs under test have about 1 to 2 LSB INL, which is a known testing challenge. The  $INL_k$  of the ADC is first tested with a linear ramp signal generated by a 20-bit high-precision sigma-delta DAC using the conventional histogram algorithm. The measured result is plotted on the top of Figure 4.8. This result will be used as an accurate reference in the following comparison.

A nonlinear signal is then generated by programming the input to the precision DAC. This signal is used to test the same ADC again. The algorithm proposed in Section III is applied to the captured histogram data.  $INL_k$  measured with the nonlinear signal is plotted on the bottom of Figure 4.8. To study the robustness of the proposed methodology, the detailed experimental setup was unknown to the test program and the input error was independently identified by the proposed algorithm.

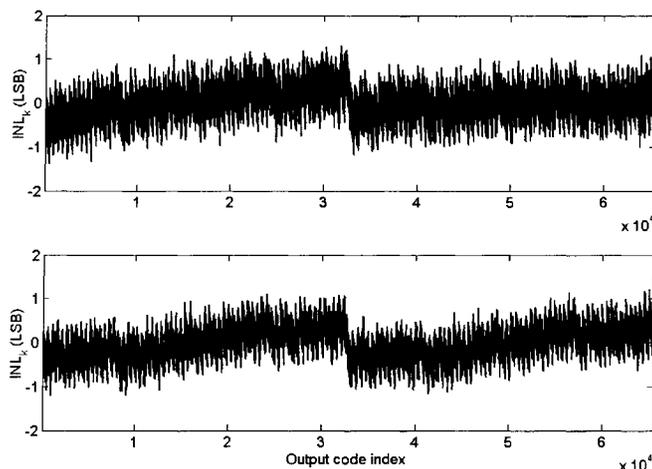


Figure 4.8 Measured  $INL_k$ . Top:  $INL=1.39$  LSB w. 20-bit linear signal. Bottom:  $INL=1.21$  LSB w. 7-bit linear signal.

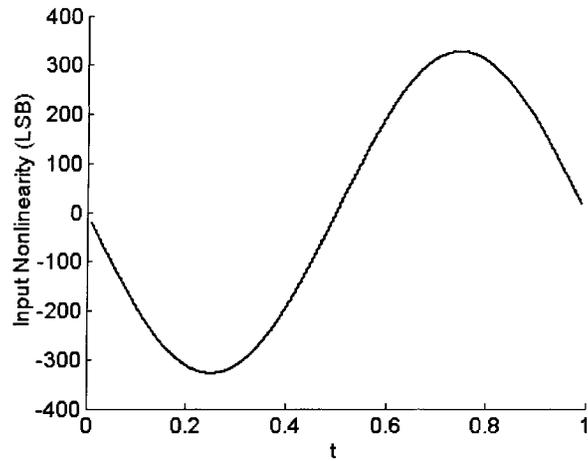


Figure 4.9 Synthesized input Nonlinearity.

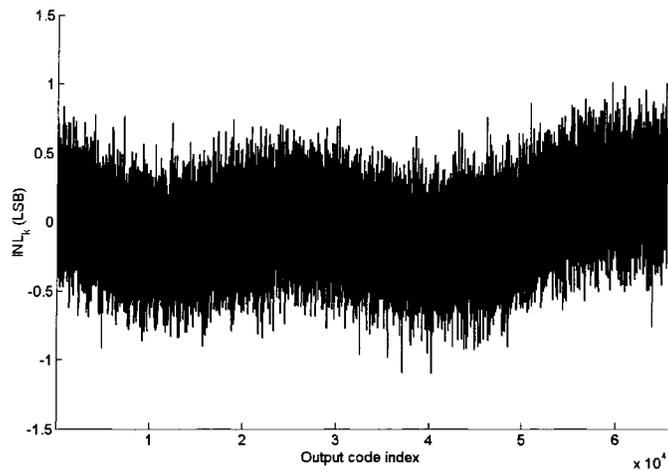


Figure 4.10 Difference between  $INL_k$  measurement results using linear and nonlinear signals.

The input nonlinearity is plotted in Figure 4.9. This signal has more than 300 LSB errors at the 16-bit level, so it is only 7-bit linear. The two  $INL_k$  plots measured with the 20-bit linear signal and the 7-bit linear signal are very close to each other. The estimated INLs are 1.39 and 1.21 LSB, respectively. The difference between the two  $INL_k$  curves is plotted in Figure 4.10. At different codes, the difference is almost always smaller than 1 LSB, and a significant

part of it comes from the noise in the measurement system, which can be reduced by increasing the total number of samples. So the 7-bit linear signal is successfully used to test 16-bit ADC's linearity and 1 LSB accuracy is achieved by using the proposed input error removal algorithm.

The experiment was repeated on a different ADC. The  $INL_k$  curves measured with the same linear and nonlinear signals are plotted on the top and bottom of Figure 4.11, respectively, and the INL is measured as 1.66 and 1.45 LSB. The two  $INL_k$  results track with each other very well. Difference of estimation for the 2<sup>nd</sup> ADC is given in Figure 4.12. All the errors are less than 1 LSB at the 16-bit level. This further confirms the capability of the proposed algorithm for testing high-resolution ADCs with single nonlinear signal.

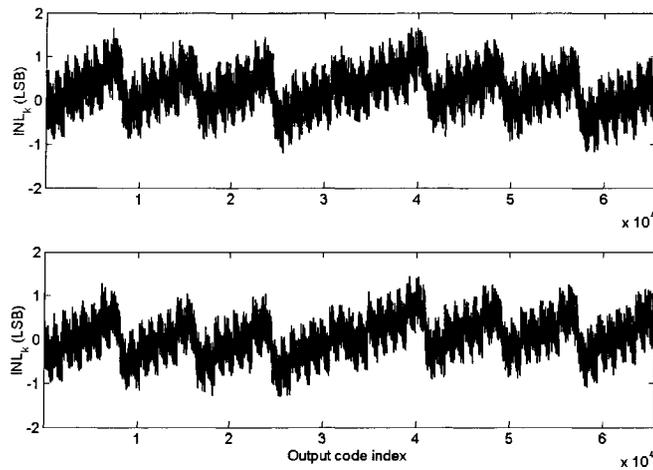


Figure 4.11 Measured  $INL_k$ . Top:  $INL=1.66$  LSB w. 20-bit linear signal. Bottom:  $INL=1.45$  LSB w. 7-bit linear signal.

To see the effect of input noise,  $INL_k$  of the ADC was tested twice by using a linear signal, with a sampling density of 16 samples per code. The difference between the two

measurement results is given in Figure 4.13. There is an error band of about 1.5 LSB due to the input noise. With a sampling density of 32 samples per code, this band is expected to be reduced to about 1 LSB. Comparing Figure 4.10 and 4.12 to Figure 4.13, it can be shown that the test accuracy of the algorithm using a single nonlinear signal is very close to the conventional method, with major errors introduced by input noise.

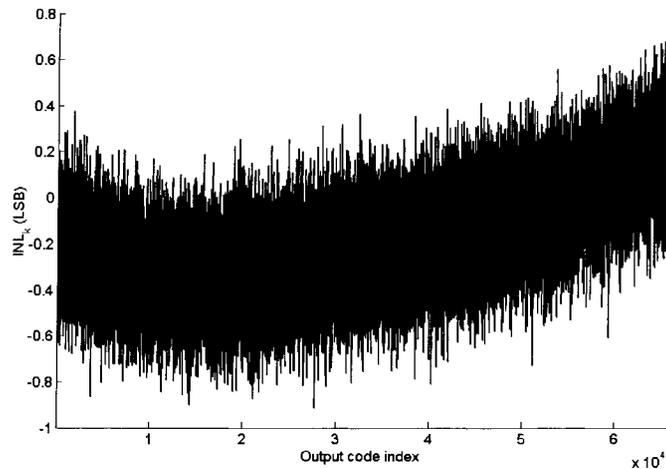


Figure 4.12 Difference between  $INL_k$  measurement results using linear and nonlinear signals.

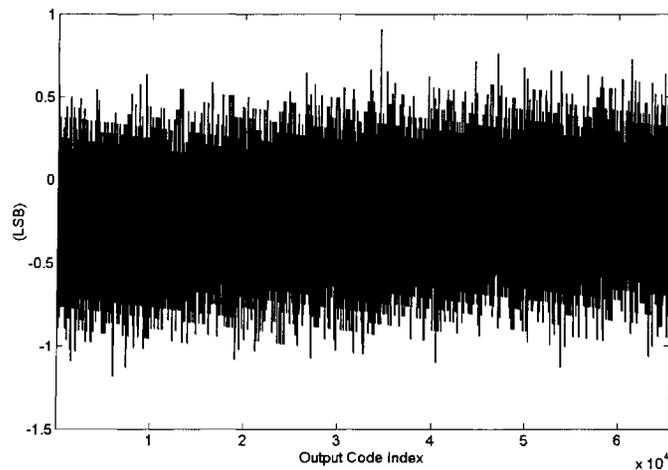


Figure 4.13 Difference between two  $INL_k$  measurement results using linear signal.

## VI. Limitations and Further Discussions

Some implementation issues of the algorithm will be discussed in this section. Discussions on the computation overhead of the proposed algorithm will also be provided shortly.

It can be observed from (4.11) that if the input nonlinearity error  $F(t)$  has repeating components that are coupled for lower and upper transition levels,  $T_k$  and  $T_{k+N/2}$ , these components will cancel each other in the expression and can not be accurately identified using the LS method in (4.14) and cause errors in final ADC test results in (4.15). This type of error components do not usually appear in common input signals and can be purposely avoided in the test environment by various ways. We can pay attention to design of the signal generator or filter to eliminate these repeating components. We can also carefully arrange the relative voltage level between the test signal and the ADC's input range so that the input error  $F(t)$  and the ADC  $INL_k$  do not have coupled periodical components. Or we can repeat the test and add an offset to the input signal so that the input error component is not coupled with  $INL_k$  in the new test and can be accurately identified. We are also investigating some ADC design techniques that can help solve this problem by making some hardware modifications to the converter structure, while introducing little overhead and error sources.

The basis functions in (4.12) can be chosen as common functions such as sine waves

$$F_j(t) = \sin(j\pi t), j = 1, 2, \dots, M; \quad (4.16)$$

or polynomials

$$\begin{aligned} F_1(t) &= t(t-1), \\ F_2(t) &= t(t-1)(t-1/2), \\ F_3(t) &= t(t-1)(t-1/2)(t-1/4), \\ F_4(t) &= t(t-1)(t-1/2)(t-1/4)(t-3/4), \\ &\dots \end{aligned} \quad (4.17)$$

where  $t$  is the normalized test time between 0 and 1. Other forms of basis functions can be used as well, primarily determined by the signal generator's characteristics. For example, exponential functions may better describe the behavior of some circuits. If appropriate basis functions are chosen, only a small number of basis functions are needed. This can reduce the computational complexity of the LS algorithm.

Details of the LS method in (4.14) will be given for readers' convenience of practically implementing the algorithm. Write (4.13) for all  $k$  from 1 to  $N/2 - 1$  in a matrix form,

$$\vec{E} = \vec{\Delta t} - \vec{\Delta F} \times \vec{\alpha}, \quad (4.18)$$

where

$$\vec{\Delta t} = [t_{1+N/2} - t_1, t_{2+N/2} - t_2, \dots, t_{N-1} - t_{N/2-1}]^T,$$

$$\vec{\Delta F} = \begin{bmatrix} F_1(t_1) - F_1(t_{1+N/2}) & \dots & F_M(t_1) - F_M(t_{1+N/2}) & 1 \\ F_1(t_2) - F_1(t_{2+N/2}) & \dots & F_M(t_2) - F_M(t_{2+N/2}) & 1 \\ \dots & \dots & \dots & \dots \\ F_1(t_{N/2-1}) - F_1(t_{N-1}) & \dots & F_M(t_{N/2-1}) - F_M(t_{N-1}) & 1 \end{bmatrix}, \quad \vec{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_M, \Delta v]^T,$$

and

$$\vec{E} = [e_{1+N/2} - e_1, e_{2+N/2} - e_2, \dots, e_{N-1} - e_{N/2-1}]^T.$$

The theory of LS method shows that [16] the estimated value of  $\vec{\alpha}$  in the form of

$$\hat{\alpha} = (\vec{\Delta F}^T \vec{\Delta F})^{-1} \vec{\Delta F}^T \vec{\Delta t} = [\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_M, \Delta \hat{v}]^T \quad (4.19)$$

minimizes the total error energy  $\vec{E}^T \vec{E}$ . Equation (4.19) gives the same results as those in (4.14). The proof of (4.19) can be done by setting the derivative of the total energy with respect to the variable vector equal to zero or with linear vector space methods. The details are omitted in this paper.

The computational complexity of the proposed method can be assessed from (4.19).  $(\overrightarrow{\Delta F}^T \overrightarrow{\Delta F})^{-1}$  is a  $M+1$  by  $M+1$  matrix, so its inversion takes proportional to  $(M+1)^3$  computations. Other matrix and vector multiplications take proportional to  $M*N$  computations. So the total computational complexity of the proposed algorithm is  $O(M*N)$ , which is linear in  $N$ . The conventional histogram method's complexity is  $O(N)$ . The computation overhead introduced by the proposed method is reasonable, as a nonlinear input signal is used, and can be managed by the current digital signal processing technique.

If the stimulus signal is not required to be highly linear, it can have many desired properties for testing. First, the signal can be continuous, which means an infinite resolution. Many simple circuits can do this, such as a MOS transistor current source charging a capacitor. Second, the input error can be composed of only low-spatial frequency components, because low-pass filters can be used to remove its high-frequency components. So a small number of basis functions are sufficient to precisely characterize the signal's nonlinearity errors. Furthermore, the signal can be generated very fast, since we don't need to wait the circuit to completely settle for achieving a high linearity. Monotonicity is also very easy to achieve for a nonlinear signal, and has already been assumed in earlier discussions.

## VII. Conclusions

A histogram-based ADC linearity testing approach using a single nonlinear stimulus signal is introduced. This approach is applicable to some widely used high-resolution ADC architectures, including cyclic and pipeline. The proposed algorithm exploits the ADC architecture knowledge in data processing. It first identifies and removes the input errors using the series expansion and least squares method, and then accurately measures the

ADC's linearity. Simulation and experimental results show that 16-bit ADCs can be tested to 1 LSB accuracy by using a 7-bit linear signal. This approach provides a promising solution to both the production and on-chip testing problems of high-resolution ADCs, since it does not require high-linearity stimulus signals and its computational complexity is reasonably low. The proposed idea can be extended to develop similar algorithms for other AMS circuits testing.

### Acknowledgements

This work is supported by the Semiconductor Research Corporation (SRC). The authors would like to thank Turker Kuyel and Kumar Parthasarathy with Taxes Instruments for their help on collecting the experimental data.

### References

- [1] *International Technology Roadmap for Semiconductors*, 2003 edition, available online: <http://public.itrs.net>.
- [2] J. Doernberg, H.-S. Lee, and D.A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE J. Solid-State Circuits*, SC-19, pp. 820-827, December 1984.
- [3] J. Blair, "Histogram measurement of ADC nonlinearities using sine waves," *IEEE Trans. Instrum. Meas.*, vol. 43, pp. 373-383, June 1994.
- [4] IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters, Dec 2000.
- [5] L. Jin, K. Parthasarathy, T. Kuyel, D. Chen and R. Geiger, "Accurate testing of analog-to-digital converters using low linearity signals with stimulus error identification and removal," *IEEE Trans. Instrum. Meas.*, vol. 54, pp. 1188 – 1199, June 2005.

- [6] L. Jin, K. Parthasarathy, T. Kuyel, R. Geiger, and D. Chen, "High-performance ADC linearity test using low-precision signals in non-stationary environments," in *Proc. 2005 International Test Conference*, pp. 1182 – 1191, Nov. 2005.
- [7] D. Johns and K. Martin, *Analog Integrated Circuit Design*. New York, NY: John Wiley & Sons, 1997.
- [8] T. Kuyel, "Linearity Testing Issues of Analog-to-Digital Converters," in *Proc. 1999 International Test Conference*, pp. 747-756, Sept 1999.
- [9] C.R. Grace, P.J. Hurst, and S.H. Lewis, "A 12-bit 80-MSample/s pipelined ADC with bootstrapped digital calibration," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1038-1046, May 2005.
- [10] A.N. Karanicolas, H.-S. Lee, and K.L. Bacrania, "A 15b 1Ms/s digitally self-calibrated pipeline ADC," *IEEE J. Solid State Circuits*, vol. 28, pp. 1207-1215, Dec 1993.
- [11] G. W. Roberts, "DFT Techniques for Mixed-Signal Integrated Circuits," *Circuits And Systems In The Information Age*, IEEE Press, pp. 251-271, June 1997.
- [12] E. Siragusa and I. Galton, "A Digitally Enhanced 1.8-V 15-Bit 40-MSample/s CMOS Pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 39, pp 2126- 2138, Dec 2004.
- [13] A. Roy, S. Sunter, A. Fudoli, and D. Appello, "High Accuracy Stimulus Generation for A/D Converter BIST," in *Proc. 2002 International Test Conference*, pp. 1031-1039, Oct. 2002.
- [14] S. Bernard, M. Comte, F. Azais, Y. Bertrand, and M. Renovell, "A New Methodology for ADC Test Flow Optimization," in *Proc. 2003 International Test Conference*, vol. 1, pp. 201-209, Sept 2003.

- [15] B. Provost and E. Sanchez-Sinencio, "On-Chip Ramp Generators for Mixed-Signal BIST and ADC Self-Test," *IEEE J. Solid-State Circuits*, vol. 38, pp. 263-273, Feb 2003.
- [16] D.G. Luenberger, *Optimization by Vector Space Methods*. New York, NY: John Wiley and Sons, 1997.

## Chapter 5

# Linearity Test of Analog-to-Digital Converters Using Kalman Filtering

A paper submitted to the 2006 International Test Conference

Le Jin, Degang Chen and Randall Geiger

### Abstract

This work introduces an efficient code-density linearity testing algorithm for ADCs that can achieve high accuracy within short test time. The proposed algorithm uses Kalman filtering to suppress the effect of errors in the histogram counts. Appropriate versions of the algorithm for ADCs of flash and pipelined architectures are introduced respectively. Simulation results show that this approach can reduce the  $INL_k$  estimation error by over 50% and achieve desired accuracy with a much smaller number of samples as compared to the conventional algorithm. Simulation and experimental results show that the proposed algorithm can significantly shorten the linearity test time by a factor of 10 or higher. Therefore, it can enable test and help maintain the quality of high-performance ADCs, and reduce the production test time and cost for medium and low resolution ADCs.

### I. Introduction

Integral nonlinearity (INL) and differential nonlinearity (DNL) are two critical specifications on static performance of analog-to-digital converters (ADCs). Accurate INL and DNL test

results can indicate the quality of an ADC and provide information of nonideality in design and manufacture, so they are always tested for ADC products on the market. The test cost of INL and DNL is a big concern of ADC manufacturers for the following reasons. First is that the test cost itself is high. Nowadays the analog test cost of a mixed-signal device can take a significant portion of its total production cost and this ratio is expected to become even higher in the future [11]. Cut off the test cost can significantly reduce the total production cost. The second reason is that the ADC has a high volume. It is one of the world's largest volume analog and mixed-signal (AMS) integrated circuit products [1]. Test cost reduction of INL and DNL has essential meaning to manufacturers as the multiplication effect of the large volume is considered. Furthermore, for current and upcoming high-resolution ADCs, time for full-code INL and DNL test, which is directly related to the cost, is prohibitively long because of the large number of variables to be accurately measured. Reduced-code test methods are often adopted for high-resolution parts, but they do not provide complete characterization of the ADC under test. There is lack of a full-code linearity test strategy with a sufficiently low cost for production test of high-resolution ADCs. Because of these reasons, a testing algorithm that can achieve a desired level of accuracy in a short period of time is very useful and important, especially for ADCs of 16-bit and up resolutions that have a large number of transition levels to be tested with very high accuracy on a test facility.

The code-density test method [2, 3] is widely adopted for testing ADCs' static linearity in the industry, because its implementation is straightforward and its computational complexity is low. The conventional code-density method requires input signals with better linearity than the resolution of the ADC under test and a significant amount of samples to average out the noise effect for accurate measurement results. The requirement on the input signal's linearity

has been studied and new methods using nonlinear signals are proposed [6], but not many approaches are proposed for improving efficiency of the code-density test method as authors are aware of. A maximum likelihood estimator (MLE) method related to the code-density test was studied in [5], but its computational complexity is high and not suitable for high-resolution ADCs.

This work introduces a modified code-density algorithm that uses the Kalman filter to reduce the effect of errors on the histogram data. Simulation and experimental results show that the proposed algorithm can achieve the same level of accuracy as that of the conventional code-density algorithm but using a significantly smaller number of samples, which means shorter test time and lower test cost. The additional computations introduced by the modified algorithm is very few and negligible with current days' computing power. The new method can be used to enable testing of high-resolution ADCs with better coverage and reduce the time and cost of testing medium-resolution ADCs.

The rest part of the paper is organized as follows. Section II will briefly review the code-density test method and discuss its estimation errors in details. Section III describes the modified algorithms using Kalman filtering to improve the test accuracy for flash and pipeline ADCs. Simulation results are presented in Section IV, and experimental results are given in Section V.

## **II. Code-Density Test for ADC Linearity**

This section provides some background information on ADC linearity specifications and the code-density test method with discussions on its performance.

### ***ADC Linearity Specifications***

Terminologies used in this paper will follow [4] with modifications in indexing for simplicity of formulation. An ideally linear ADC is expected to have its transition levels evenly spaced over the input range. The differential nonlinearity for code  $k$  of an  $n$ -bit ADC is the relative error in the  $k$ -th code bin width and can be written as

$$DNL_k = (N - 2) \frac{T_k - T_{k-1}}{T_{N-2} - T_0} - 1 (LSB), \quad (5.1)$$

where  $N=2^n$ ;  $T_k$ ,  $k=0, 1, \dots, N-2$ , is the  $k$ -th transition level of the ADC between code  $k$  and  $k+1$ ; and the unit of  $DNL_k$  is called a least significant bit (LSB) and equal to the averaged code bin width

$$Q = \frac{T_{N-2} - T_0}{N - 2} = 1 (LSB). \quad (5.2)$$

The integral nonlinearity for code  $k$  is

$$INL_k = (N - 2) \frac{T_k - T_0}{T_{N-2} - T_0} - k (LSB). \quad (5.3)$$

Equations (5.1) and (5.3) show that the offset and gain of transition levels do not affect values of  $DNL_k$  and  $INL_k$ . INL and DNL of the ADC are defined as

$$DNL = \max\{|DNL_k|, k = 1, 2, \dots, N - 2\} \quad (5.4)$$

and

$$INL = \max\{|INL_k|, k = 0, 1, \dots, N - 2\}, \quad (5.5)$$

respectively. They indicate the maximum deviations of an ADC from its ideally linear counterpart. From (5.1) and (5.3) we can also see that

$$INL_k = INL_{k-1} + DNL_k. \quad (5.6)$$

To test linearity performance of an ADC, its transition levels are usually measured in a relative sense and used to calculate INL and DNL based on equations above. Other static parameters such as the offset voltage and gain error are measured separately.

### ***Code-Density Test of ADC Linearity***

The code-density method uses a linear ramp or sinusoidal signal as a stimulus to the ADC under test. We assume our test signal is an ideal ramp in following discussions, but the developed algorithms, analyses and conclusions can be applied to the sine wave test as well.

The ADC takes samples  $y_i$  on the stimulus signal at a constant rate and generates output codes  $c_i = k$  if  $T_{k-1} < y_i \leq T_k$ . The number of occurrences of code  $k$  is recorded as a histogram count  $H_k$ . This count can be used as an estimate of the  $k$ -th code bin width, since a larger bin will naturally contain more samples and the two quantities are proportional to each other if the samples are evenly spaced on the voltage axis, as in the linear ramp test situation. The code-density method estimates  $DNL_k$  and  $INL_k$  as

$$D\hat{N}L_k = \frac{H_k}{\bar{H}} - 1 \quad (5.7)$$

and

$$I\hat{N}L_k = \sum_{j=1}^k H_j / \bar{H} - k, \quad (5.8)$$

respectively, where

$$\bar{H} = \sum_{j=1}^{N-2} H_j / (N-2) \quad (5.9)$$

is the averaged histogram count.

By comparing to definitions in (5.1) and (5.3), (5.7) and (5.8) are equivalent to estimating transition levels of the ADC under test as

$$\hat{T}_k = \sum_{j=0}^k H_j / N_s = \sum_{i=1}^{N_s} v_{ki} / N_s, \quad (5.10)$$

where  $N_s$  is the total number of samples, and  $v_{ki}$  is a characteristic function such that

$$v_{ki} = \begin{cases} 1, & c_i \leq k; \\ 0, & k < c_i. \end{cases} \quad (5.11)$$

The estimated  $T_k$  is proportional to the number of samples that give out an output code smaller than or equal to  $k$ .

#### ***Estimation Error in Code-Density Method***

Because of the noise in the signal generator and ADC circuitries, the sampled voltage is deviated from the stimulus signal as

$$y_i = x_i + n_i, \quad (5.12)$$

where  $x_i$  is the ideal ramp signal level at the  $i$ -th sampling instance and  $n_i$  is an additive noise.

The output code  $c_i$  and consequently  $v_{ki}$  become random variables as well. It is usually assumed that  $n_i$  is white and takes a normal distribution  $N(0, \sigma^2)$ , so the probability for the  $i$ -th output code being smaller than or equal to  $k$ , notated as  $p_{ki}$ , can be calculated as

$$\begin{aligned} p_{ki} &= P(v_{ki} = 1) = P(c_i \leq k) \\ &= \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{T_k} \exp\left(-\frac{(t-x_i)^2}{2\sigma^2}\right) dt = \Phi\left(\frac{T_k - x_i}{\sigma}\right), \end{aligned} \quad (5.13)$$

where  $\Phi(x)$  is the cumulative distribution function (CDF) of the standard normal distribution  $N(0, 1)$ . Further discussions on the above modeling and calculation can be found in [5] and its references.

As noting that  $v_{ki}$  has a Bernoulli distribution [6] and independent from each other with respect to  $i$  by our assumption, the mean and variance of the estimator in (5.10) can be written out as [7]

$$E\{\hat{T}_k\} = \sum_{i=1}^{N_s} p_{ki} / N_s \quad (5.14)$$

and

$$\text{Var}\{\hat{T}_k\} = \sum_{i=1}^{N_s} p_{ki} q_{ki} / N_s^2, \quad (5.15)$$

where  $q_{ki}=1-p_{ki}$ .

Using (5.3) and (5.8), we can further show that the  $INL_k$  estimation error is equal to the transition level estimation error expressed in LSB as

$$e_k = \hat{INL}_k - INL_k = (\hat{T}_k - T_k) / Q(\text{LSB}), \quad (5.16)$$

if the two terminal transition levels  $T_{N-2}$  and  $T_0$  are specifically measured with high accuracy, which is usually done for offset and gain error measurements. We don't have a closed-form expression of (5.14), but it can be numerically verified that this mean is very close to  $T_k$ . Therefore the estimated  $INL_k$  is unbiased with  $E\{e_k\}=0$ . The estimated value has a variance

$$\text{Var}\{e_k\} = \text{Var}\{\hat{T}_k\} / Q^2 \cong A \frac{\sigma}{D_s} (\text{LSB}^2), \quad (5.17)$$

where  $A$  is a constant,  $\sigma$  is expressed in LSB, and  $D_s$  is called the sampling density, defined as the average number of hits of an ADC output code in the test and equal to  $N_s/N$ . The second equality comes from (5.15) and is a good approximation for  $k$  not close to 0 or  $N-1$ . The value of  $A$  can be numerically calculated as about 0.6.

Equation (5.17) gives an analytical relationship between testing accuracy of the code-density method and two critical parameters. It can be intuitively interpreted that the test will be

accurate if the noise is small and the number of samples is large. As we can see that (5.7), (5.8) and (5.10) are very straightforward estimations based on the histogram counts, the performance of the code-density method is not optimal and can be improved by some sophisticated approaches.

### III. ADC Linearity Test with Kalman Filtering

The Kalman filter estimates values of the state variables of a dynamic system from noisy measurements of the variables. It can give optimal linear estimation, in the sense of minimum Mean Square Error (MSE), for signals under Gaussian noise. It has extremely wide applications in various disciplines. The ADC linearity test problem can be put in such a way that we can straightforwardly apply the existing theory of Kalman filtering and make significant improvement to testing accuracy. The flowchart of the proposed method is shown in Figure 5.1.  $H_k$ ,  $\hat{INL}_k$  and  $\tilde{INL}_k$  are the histogram count, estimated  $INL_k$  in (5.8), and the optimal estimation of  $INL_k$  using the Kalman filter, respectively.

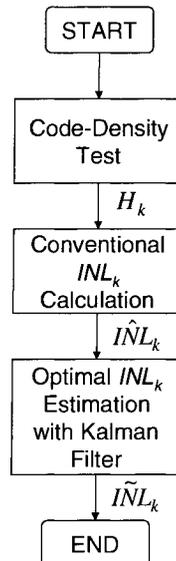


Figure 5.1. Flowchart of ADC testing with Kalman filtering.

### ***Problem Formulation***

Writing (5.6) and (5.16) together, we get  $INL_k$  of the ADC under test in the following form

$$INL_k = INL_{k-1} + DNL_k; \quad (5.18a)$$

$$\hat{INL}_k = INL_k + e_k. \quad (5.18b)$$

$INL_k$  can be viewed as a state variable of the dynamic system described by (5.18a), and we have a noisy measurement of it using the code-density method,  $\hat{INL}_k$  in (5.18b). Kalman filtering can be applied on (5.18) to find the optimal estimation of  $INL_k$ ,  $k=0, 1, \dots, N-2$ , based on the code-density test result under the following two conditions that

1)  $DNL_k$  is a Gaussian random variable of zero mean satisfying

$$E\{DNL_k DNL_l\} = C \cdot \delta_{kl}, \quad (5.19)$$

where  $\delta_{kl}$  equals 1 only if  $k=l$  and equals to 0 otherwise; and

2)  $e_k$  is Gaussian with zero mean and satisfies

$$E\{e_k e_l\} = R \cdot \delta_{kl}. \quad (5.20)$$

The condition on  $e_k$  is generally true based on Section II. The assumption on  $DNL_k$  could be violated, and we will discuss appropriate modifications to it later. We further assume  $DNL_k$  and  $e_k$  are independent which is reasonable under realistic situations.

### ***INL<sub>k</sub> Test for Flash ADCs with Kalman Filtering***

As a complete understanding of the Kalman filter requires much background information outside the scope of this paper, we will not provide details on the derivation and proof of the general Kalman filter algorithm. Interested readers are referred to [12] for rigorous descriptions and more references. We will provide detailed recursive estimation steps so that

the algorithm can be practically implemented after reading this paper. Some necessary discussions and comments are given as well.

The first condition on  $DNL_k$  is not always true for different types of ADCs, but it is a good description of ADCs with a flash architecture [9]. In a flash ADC, mismatch errors of the resistors and offset errors of comparators contribute to  $DNL_k$ . These errors are usually due to random process variations and managed to be uncorrelated using some well-known design techniques, such as common-centroid layout. Therefore  $DNL_k$ 's are independent of each other in a flash ADC, and the variance  $C$  is the summation of the mismatch error variance and offset error variance.

For flash ADCs, the Kalman filter algorithm can be described by the following equations [8] with initial conditions  $INL_0 = 0$  and  $P[0] = 0$ , which are generally true by (5.3) and (5.8),

$$R_e[k] = P[k] + R; \quad (5.21a)$$

$$K_p[k] = P[k]R_e^{-1}[k]; \quad (5.21b)$$

$$P[k+1] = P[k] + C - K_p[k]R_e[k]K_p[k]; \quad (5.21c)$$

$$\tilde{INL}_k = \tilde{INL}_{k|k-1} + K_p[k](\hat{INL}_k - \tilde{INL}_{k|k-1}); \quad (5.21d)$$

$$\tilde{INL}_{k+1|k} = \tilde{INL}_k. \quad (5.21e)$$

$\tilde{INL}_k$  in (5.21d) is the optimal estimation of  $INL_k$  given all the histogram information. It is calculated from  $\tilde{INL}_{k|k-1}$ , the predicted value of  $INL_k$  based on the histogram information for codes less than or equal to  $k-1$ , with correction introduced by the measured  $\hat{INL}_k$ . The difference between the measured and predicted values,  $\hat{INL}_k - \tilde{INL}_{k|k-1}$ , is called the measurement innovation. It indicates how good the prediction is but is also affected by the

measurement error  $e_k$ .  $K_p[k]$  in (5.21d) is the Kalman gain. It determines how much of the innovation should be used as correction to the predicted value. It can be shown that  $P[k]$  is the variance of the final prediction error of  $\tilde{I}NL_{k|k-1}$ .

$\tilde{I}NL_k$  and  $\tilde{I}NL_{k|k-1}$  are recursively calculated using (5.21d) and (5.21e) for all  $k$  values from 0 to  $N - 2$ . The Kalman gain  $K_p[k]$  is also recursively determined from (5.21a) to (5.21c) and targeted to minimize the mean square error of the final estimation  $\tilde{I}NL_k$ . An intuitive interpretation of the Kalman gain is as follows. If the measurement error is comparatively small, consequently  $R$  small, the measured value is better representing the true value, and vice versa. So when  $R$  is small,  $K_p[k]$  is close to 1 and the final estimation is closer to the measured value. When  $R$  is large, the Kalman gain is smaller than 1 and the final estimation is more dominated by the predicted value.

### ***INL<sub>k</sub> Test for Pipeline ADCs with Kalman Filtering***

The  $DNL_k$  usually has some repetitive patterns for ADCs with other architectures. One of the most widely adopted architectures is the pipelined structure, for which  $DNL_k$  and  $DNL_{k+N/2}$  are strongly correlated. ([10] provides a very good description of pipeline ADCs.) Under the case of such ADCs, we can first identify the deterministic component in  $DNL_k$  and make the residual part independent of each other.

Taking the pipeline ADC as an example, we use the average of  $DNL_k$  and  $DNL_{k+N/2}$  as an approximation of their deterministic component,

$$DNL'_k = DNL'_{k+N/2} = \frac{D\hat{N}L_k + D\hat{N}L_{k+N/2}}{2}. \quad (5.22)$$

The true  $DNL_k$  can be written as

$$DNL_k = DNL'_k + d_k, \quad (5.23)$$

where  $d_k$  is a random component of zero mean, contributed by the random errors in the ADC and the test errors in the code-density method, satisfying

$$E\{d_k d_l\} = C' \cdot \delta_{kl}. \quad (5.24)$$

The value of  $d_k$ , instead of  $DNL_k$  in the flash case, will be predicted by Kalman filtering based on the code-density test results. Replacing (5.23) into (5.18a), we can see that the Kalman filtering steps need to be modified as

$$R_e[k] = P[k] + R; \quad (5.25a)$$

$$K_p[k] = P[k]R_e^{-1}[k]; \quad (5.25b)$$

$$P[k+1] = P[k] + C' - K_p[k]R_e[k]K_p[k]; \quad (5.25c)$$

$$\tilde{INL}_k = \tilde{INL}_{k|k-1} + K_p[k](\hat{INL}_k - \tilde{INL}_{k|k-1}); \quad (5.25d)$$

$$\tilde{INL}_{k+1|k} = \tilde{INL}_k + DNL'_{k+1}. \quad (5.25e)$$

The major difference between (5.25) and (5.21) is in the prediction step. For the flash ADC, the  $DNL_k$  is modeled as completely random with zero mean, so the best prediction of  $INL_{k+1}$  is  $\tilde{INL}_k$ , (5.21e). For the pipelined ADC, the deterministic component should be added to get the prediction as in (5.25e), which is intuitively reasonable and mathematically correct. For other architectures of ADCs, similar variations as (5.25) can be found when appropriate circuit level models are used.

### ***Performance Analysis for Proposed Algorithm***

We use the flash ADC equations as an example to study how Kalman filtering can help improve test accuracy. The variance of the final prediction error,  $P[k]$ , can be calculated from

(5.21) by letting  $k$  goes to infinity so that the variables settle to their steady state values, notated with a subscript “s”,

$$\begin{aligned} R_{es} &= P_s + R, \\ K_{ps} &= P_s R_{es}^{-1}, \\ P_s &= P_s + C - K_{ps} R_{es} K_{ps}. \end{aligned} \quad (5.26)$$

Solving above equations gives the final prediction error variance as

$$P_s = \frac{C + \sqrt{C^2 + 4RC}}{2}, \quad (5.27)$$

where the negative solution is discarded. This is the variance of the error between  $\tilde{INL}_{k|k-1}$  and  $INL_k$ .  $P_s$  is smaller than  $R$  when  $C$  is much smaller than  $R$ . This is usually true for flash ADCs, since the standard deviation of  $e_k$  could be about 1 LSB or larger for median and high resolution ADCs while the standard deviation of  $DNL_k$  is much less than 1 LSB. This means that even the predicted value of  $INL_k$  can have better accuracy than the code density results.

The final estimation error can be calculated as [12],

$$\begin{aligned} R_e &= \text{var}\{\tilde{INL}_k - INL_k\} \\ &= P_s (1 - K_{ps}) = \frac{P_s R}{P_s + R} = \frac{\sqrt{C^2 + 4RC} - C}{2}. \end{aligned} \quad (5.28)$$

This variance is strictly smaller than  $R$ , which means using the Kalman filter can effectively improve the accuracy of code density testing. Note that for the case of pipeline ADCs, the  $C$  in (5.27) should be replaced with  $C'$ .

Equations (5.21) and (5.25) show that the computational complexity of the Kalman filter algorithm is proportional to  $N$ . For each  $INL_k$  value, it takes some basic operations. This complexity is acceptable with current data processing power and does not introduce any significant computation overhead. While the proposed algorithm can achieve the same

performance as the conventional algorithm with much fewer sample and consequently shorter testing time, this algorithm has obvious benefits for testing high-resolution ADCs.

#### IV. Simulation Results

Simulations have been done to validate the effectiveness of the proposed algorithms on various types of ADC architectures and with different test parameters. All results show that the advantage of using Kalman filtering in linearity test is significant.

##### *Single ADC Test*

The proposed algorithm was first compared with the conventional code-density algorithm for testing a 16-bit flash ADC.  $DNL_k$  of the flash ADC has zero mean and a standard deviation of 0.5% LSB. The true INL is 1.39 LSB. The input signal is an ideal ramp plus a white Gaussian noise with zero mean and a standard deviation of 0.5 LSB. The average number of samples per code bin is 32. The  $INL_k$  estimation results are given in Fig. 5.2 and 3. On the top of Figure 5.2, true  $INL_k$  and estimated values with the conventional code-density algorithm are plotted. The estimation error is plotted on the bottom with a standard deviation of about 0.1 LSB. The INL is estimated as 1.68 LSB. Figure 5.3 shows the results when Kalman filtering is used, with true and estimated  $INL_k$  on top and estimation error on bottom. The estimation error has a standard deviation of 0.04 LSB. The INL is estimated as 1.50 LSB. Comparing Figure 5.2 to Figure 5.3, we can see that Kalman filtering significantly reduced the estimation error, by more than 50%, and the estimated INL is closer to the true value, achieved 0.1 LSB accuracy at 16-bit level.

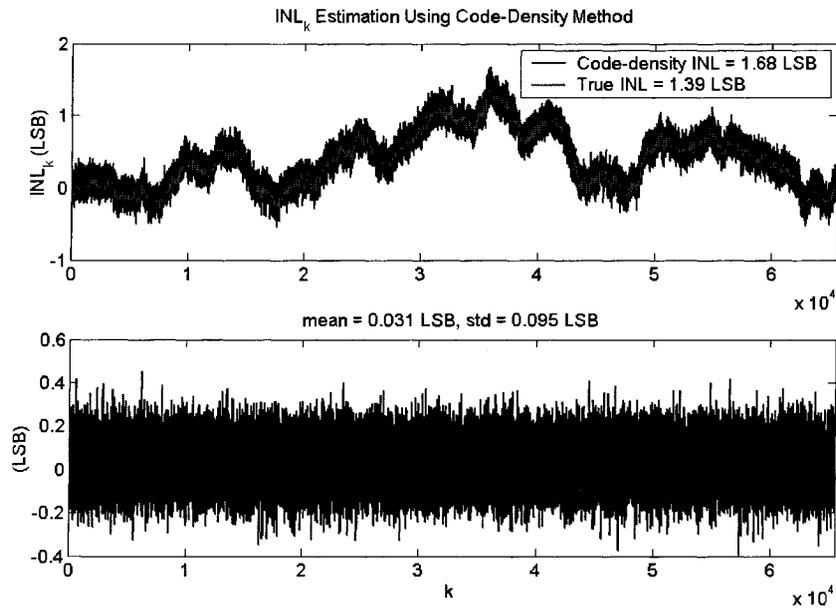


Figure 5.2.  $INL_k$  estimation using code-density method.

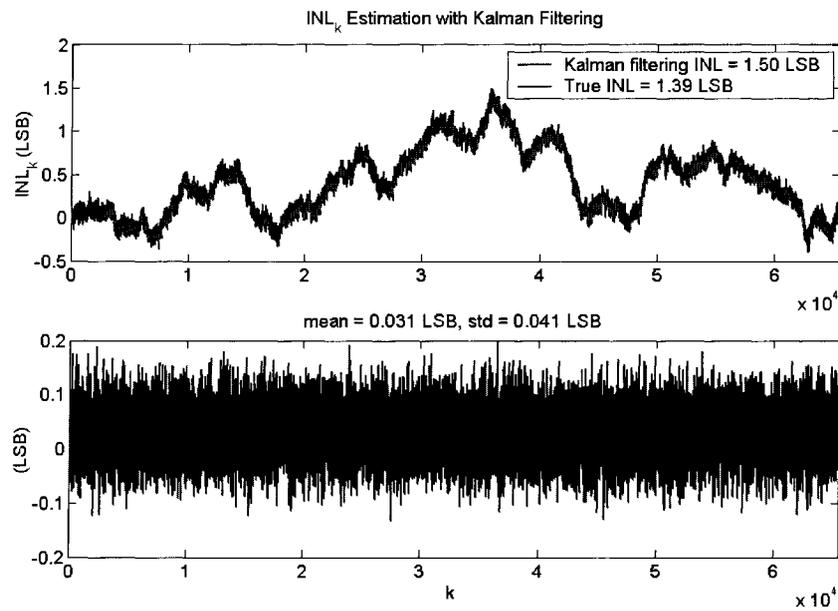


Figure 5.3.  $INL_k$  estimation with Kalman filtering.

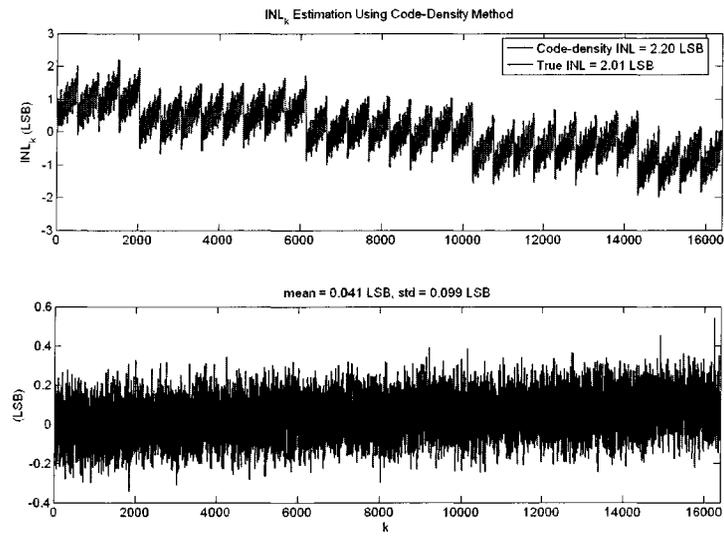


Figure 5.4.  $INL_k$  estimation using code-density method. Top: True and estimated  $INL_k$ ; Bottom:  $INL_k$  estimation error.

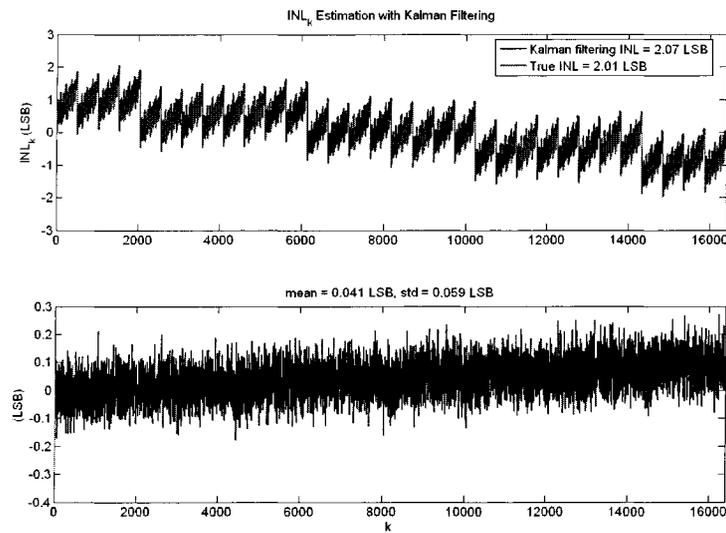


Figure 5.5.  $INL_k$  estimation with Kalman filtering. Top: True and estimated  $INL_k$ ; Bottom:  $INL_k$  estimation error.

A 14-bit pipeline ADC with linear errors was then tested using two methods in simulation. Its true INL is 2.01 LSB. The input noise has a standard deviation of 0.5 LSB at the 14-bit level.

The average number of samples per bin is 32. The operation as discussed in (5.23) was applied to the first three stages of the ADC. The results using conventional and Kalman filtering methods are plotted in Figure 5.4 and 5, respectively. We can see that the  $INL_k$  estimation error with Kalman filtering is much smaller. The standard deviation of the estimation error with the conventional method is 0.1 LSB and is reduced to 0.06 LSB when Kalman filtering is used. The estimated INL with Kalman filtering, 2.07 LSB, is also closer to the true value than the conventional method, 2.20 LSB.

### Statistical Result

Firstly, the relationship between the accuracy improvement of the Kalman filtering method and the number of samples was studied. For saving simulation time, 14-bit flash ADCs were used. The standard deviation of these ADCs'  $DNL_k$  is 0.8% of LSB. The INL of these ADCs ranges from 0.4 LSB to 2 LSB. Input noise has a standard deviation of 0.5 LSB.

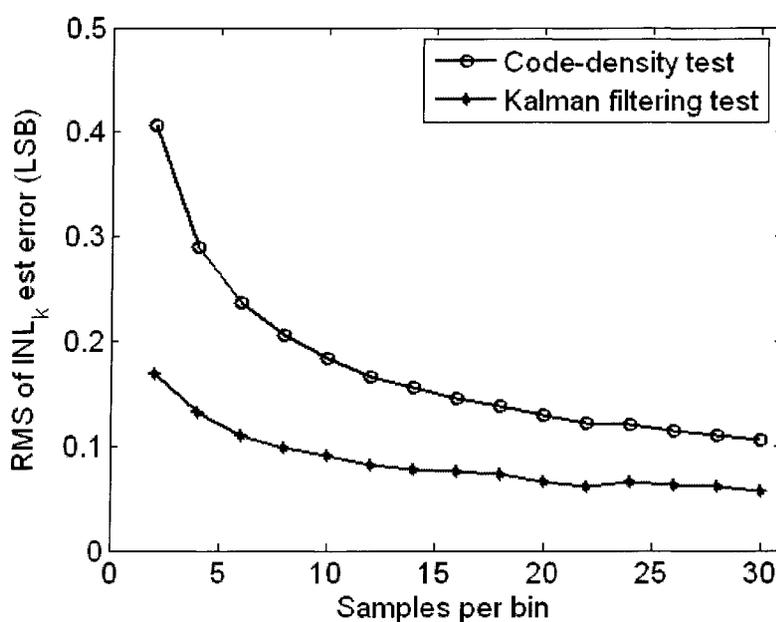


Figure 5.6. RMS value of  $INL_k$  estimation error vs. number of samples.

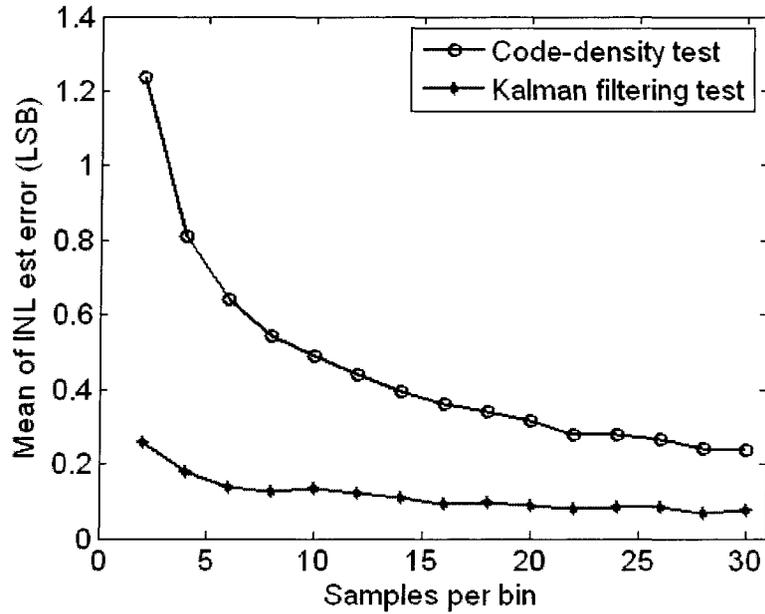


Figure 5.7. Mean value of INL estimation error vs. number of samples.

For each different number of samples, the RMS value of  $INL_k$  estimation error was collected from 128 ADCs and the averaged values are plotted in Figure 5.6. The Kalman filtering method can reduce the test error by 50% and can achieve the same accuracy level by using only about 1/10 of the number of samples as compared to the conventional code-density method. The mean values of INL estimation error are compared in Figure 5.7 for the two methods. We can see significant improvement in test results introduced by the proposed algorithm as well.

The relationship between the accuracy improvement and the input noise was investigated as well. 14-bit flash ADCs with same error characteristics as in the previous paragraph were test with 32 samples per bin on average. The RMS values of the  $INL_k$  estimation error and the

mean values of INL estimation error are plotted as a function of the standard deviation of the input noise in Figure 5.8 and 9, respectively. Each data point is an average from 128 ADCs.

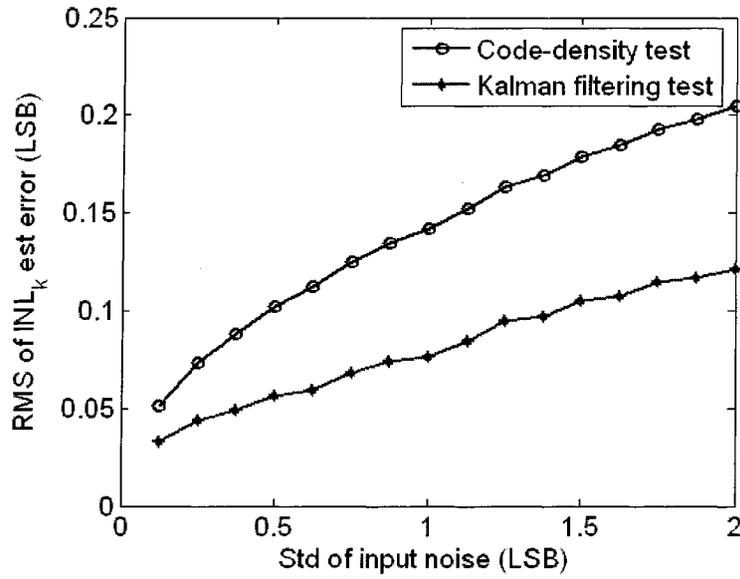


Figure 5.8. RMS value of  $INL_k$  estimation error vs. standard deviation of input noise.

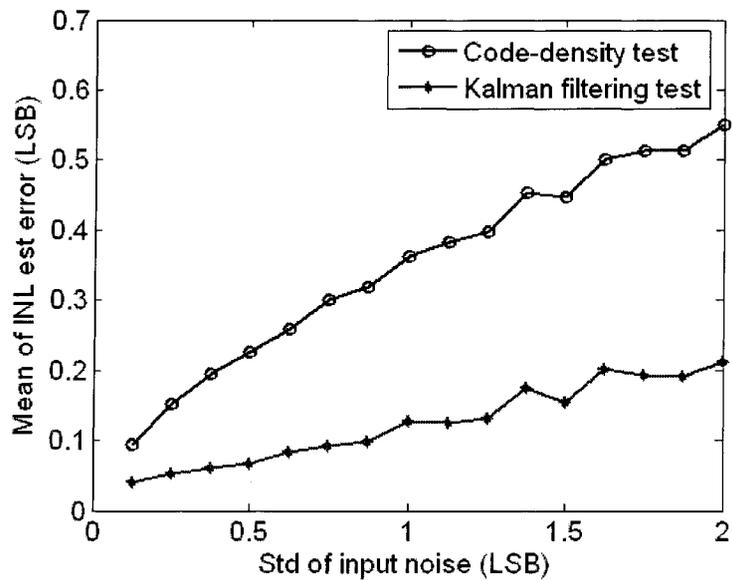


Figure 5.9. Mean value of INL estimation error vs. standard deviation of input noise.

The  $INL_k$  estimation error was obviously smaller with the modified Kalman filtering algorithm and the INL estimation error was reduced even more percentage wise. The proposed Kalman filtering algorithm can improve the test performance of the code-density method and achieve a high accuracy level under different noise situations with a reasonable number of samples.

### ***Discussion***

The results in Figure 5.6 are in agreement with (5.17). When the sampling density was multiplied by 4, the  $INL_k$  estimation errors can only be reduced by 2 times, see those circles in Figure 5.6. This means increasing the number of samples and consequently the test time can not efficiently improve testing accuracy. We can achieve such improvement by applying a short piece of code during the data processing stage. Kalman filtering is very cost-effective. The capability of the proposed Kalman filtering algorithm for significantly reducing linearity test time can be seen from Figure 5.6 and 7. For achieving 0.2 LSB accuracy in INL test, the Kalman filtering algorithm only requires a few samples while the conventional code-density needs more than 30 samples per code bin on average. This can save 90% of test time and make it possible for full-code test of some highest-resolution ADCs.

Analysis of (5.17) and Figure 5.8 show that reducing noise also is not efficient in improving test accuracy. Reducing the standard deviation of the noise by a factor of 4 can only reduce the standard deviation of the  $INL_k$  estimation error by a factor of 2. And the noise can not be unlimitedly reduced, because the  $kT/C$  noise always exists in an ADC and is determined by design. For ADCs with higher than 16-bit resolutions, it is difficult to reduce the noise to less than the half LSB level. Figure 5.8 and Figure 5.9 show that the proposed algorithm can

achieve 0.1 LSB test accuracy under such a noise level with a few samples per code, while the conventional algorithm cannot even with tens of samples.

## V. Experimental Results

The experiment platform was a Teradyne A580 Advanced Mixed Signal Tester. Linear ramps were generated by a 20-bit multi-bit delta-sigma DAC for the code-density test. To avoid sudden jump in the output of the waveform generator, triangular waveforms were implemented with appropriate coding of the input to the DAC.

Experiments were taken on 16-bit commercial ADC for validating the proposed algorithm with Kalman filtering. A typical  $INL_k$  plot of the ADC used in the test is given on the top of Figure 5.10. This result was measured with 32 samples per code on average and will be used as a reference to evaluate the results from histogram data collected with a smaller sample density.

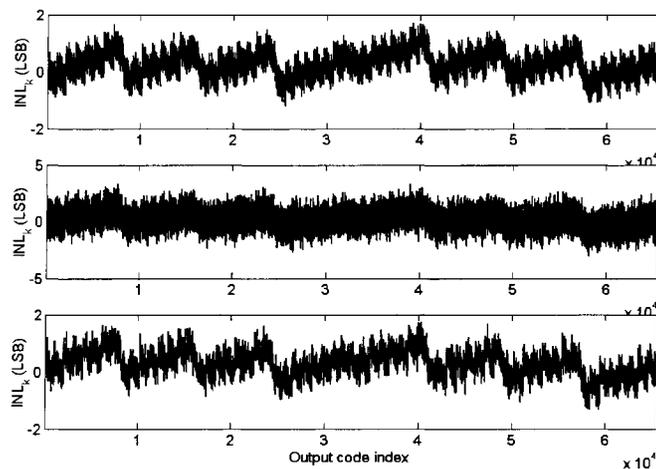


Figure 5.10.  $INL_k$  measurement for a 16-bit ADC.

$INL_k$  of the same ADC was then tested with  $D_s$  equal to one sample per code and the result is plotted in the middle of Figure 5.10, which does not track the top curve very well. Significant noise can be observed. The difference between the results using 32 samples per code and one sample per code is plotted on the top of Figure 5.11. The difference has an error band of about 2 LSB. The result with 32 samples per code is pretty accurate. The difference mostly came from the one-sample-per-code result, because the additive noise in the system and quantization effects could not be well averaged out by the small number of samples.

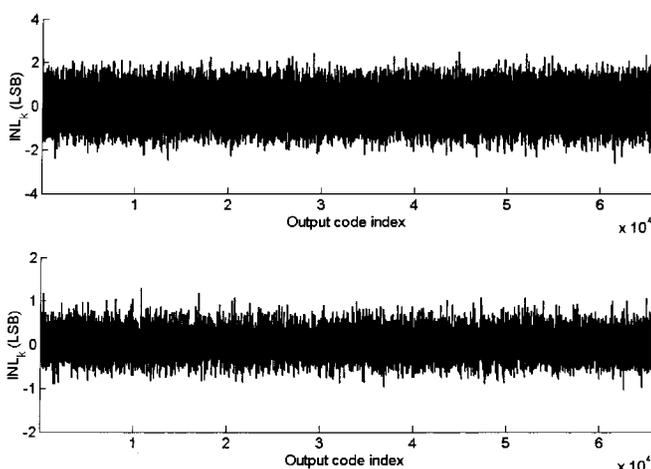


Figure 5.11. Estimation error in  $INL_k$  test.

Kalman Filtering was applied to the one-sample-per-code result as discussed in Section III. The values of  $R$  and  $C$  in (5.21) were first characterized for the system and the ADC under test from previous measurement results. The new  $INL_k$  is plotted on the bottom of Figure 5.10. It matches with the 32-samples-per-code result very well. Even the small details are almost identical in the two plots. The difference between them is plotted on the bottom of Figure 5.11, which are nearly all less than 1 LSB at the 16-bit level this time. By comparing the

results, we conclude that the proposed algorithm can reduce the linearity test error by more than 50% with a very small sample density.

More experiments with the same setup were done to further investigate the improvement introduced by Kalman filtering, as shown in Figure 5.12. The ADC was tested with one sample per code 32 times. For each run, the results estimated with and without Kalman filtering are compared to the reference result from the 32-samples-per-code test and the difference are taken as errors. The top of Figure 5.12 shows the improvement in INL estimation, where the INL estimation errors of one-sample-per-code tests are drawn in circles and the estimation errors of one-sample-per-code tests with Kalman filtering are drawn in diamonds. It can be observed that the errors were reduced from about 2 LSB to less than 1 LSB. The bottom of Figure 5.12 shows the improvement in the variance of  $INL_k$  estimation errors for different runs. The variance was reduced from about  $0.4 \text{ LSB}^2$  to  $0.07 \text{ LSB}^2$  using Kalman filtering.

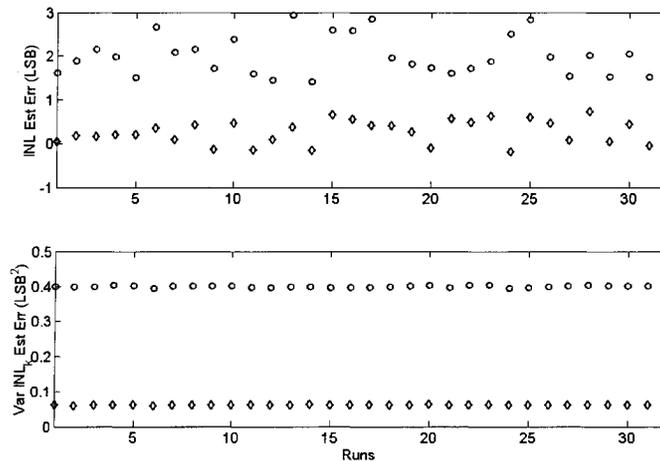


Figure 5.12. Performance comparison: with and without Kalman Filtering.

Some more performance validation was done for the algorithm. Two samples per code were used and the results are summarized in Figure 5.13. As the convention in Figure 5.12, circles represent results without Kalman filtering and diamonds represent results with filtering. The performance improvement with Kalman filtering is still obvious. Because of the increase in the number of samples, both the INL estimation errors and variances of  $INL_k$  errors became smaller. The variance of  $INL_k$  errors was reduced by a factor of two when the number of samples doubles, which is in agreement with the discussion in Section II.

Kalman filtering can significantly reduce the errors in ADC test results. From the experimental results, we can conclude that the measured ADC specifications from 32 samples per code and from one or two samples per code with Kalman filtering are very close to each other. That means we can dramatically reduce the test time, by a factor of 10 or more. Since the data processing using Kalman filter does not introduce any hardware overhead and a little computation overhead which is easy to get in nowadays digital technologies, it can help engineers effectively cut down the test cost.

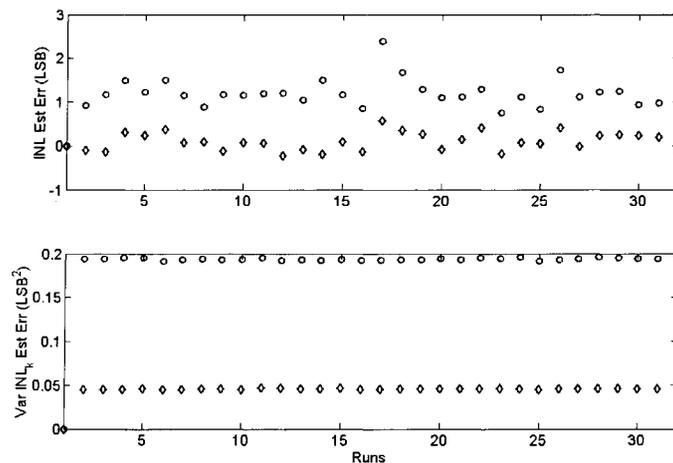


Figure 5.13. Results with two-samples-per-code tests.

## VI. Conclusions

The performance of the code-density method for ADC linearity testing is analyzed in this work and the optimization algorithm using Kalman filtering for improving test accuracy of the conventional method is introduced for flash and pipeline ADCs. Simulation and experimental results show that the proposed algorithms can effectively reduce the variance of  $INL_k$  test errors when using the same number of samples as the conventional algorithm. This method can effectively reduce the test time, by 10 times or more, and provide accurate test results of ADC static linearity. The proposed algorithm provides an enabling solution to the problem of cost-effective test of high-resolution ADCs with 16-bit or higher resolutions. It can also significantly reduce the test cost for high-volume medium and low-resolution ADCs.

## Acknowledgements

The authors would like to thank Turker Kuyel and Kumar Parthasarathy with Taxes Instruments for their help on collecting the experimental data.

## References

- [1] International Technology Roadmap for Semiconductors, 2003 edition, available online: <http://public.itrs.net>.
- [2] J. Doernberg, H.-S. Lee, and D.A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE J. Solid-State Circuits*, SC-19, pp. 820-827, December 1984.
- [3] J. Blair, "Histogram measurement of ADC nonlinearities using sine waves," *IEEE Trans. Instrum. Meas.*, vol. 43, pp. 373-383, June 1994.
- [4] *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*, Dec 2000.

- [5] P.D. Capofreddi and B.A. Wooley, "The efficiency of methods for measuring A/D converter linearity," *IEEE Trans. Instrum. Meas.*, vol. 48, pp. 763 – 769, June 1999.
- [6] L. Jin, K. Parthasarathy, T. Kuyel, D. Chen and R. Geiger, "Accurate testing of analog-to-digital converters using low linearity signals with stimulus error identification and removal," *IEEE Trans. Instrum. Meas.*, vol. 54, pp. 1188 – 1199, June 2005.
- [7] G. Casella and R.L. Berger, *Statistical Inference*. Pacific Grove, CA: DUXBURY, 2001.
- [8] D.G. Luenberger, *Optimization by Vector Space Methods*. New York, NY: John Wiley and Sons, 1997.
- [9] D. Johns and K. Martin, *Analog Integrated Circuit Design*. New York, NY: John Wiley & Sons, 1997.
- [10] A.N. Karanicolas, H.-S. Lee, and K.L. Bacrania, "A 15b 1Ms/s digitally self-calibrated pipeline ADC," *IEEE J. Solid State Circuits*, vol. 28, pp. 1207-1215, Dec 1993.
- [11] G. W. Roberts, "DFT Techniques for Mixed-Signal Integrated Circuits," *Circuits And Systems In The Information Age*, IEEE Press, pp. 251-271, June 1997.
- [12] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," [http://www.cs.unc.edu/~welch/media/pdf/kalman\\_intro.pdf](http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf).

## **Chapter 6**

# **Testing of Precision DACs Using Low-Resolution ADCs with Dithering**

A paper submitted to the 2006 International Test Conference

Le Jin, Hosam Haggag, Randall Geiger, and Degang Chen

### **Abstract**

The bottleneck of DAC testing is the fast and accurate measurement devices. Production testing of high-resolution DACs with Giga Hertz clock rates is a challenging problem, and there is no widely adopted approach for on-chip testing of precision DACs in an SoC system. This work presents a new approach for testing high-resolution DACs. High speed data acquisition is achieved with flash ADCs; sufficient resolution is provided by dithering; and high test accuracy is guaranteed by the proposed data processing algorithm. This method provides a potential solution to both the production and on-chip DAC testing problems. Simulation results show that the static linearity of 14 bit DACs can be tested to better than 1 LSB accuracy, and dynamic performance of more than 85 dB SFDR can be tested with 1 dB accuracy, using 6-bit ADCs and dithering. Experimental results included in the paper also affirm the performance of the algorithm in testing high-resolution DACs using 6-bit ADCs.

## I. Introduction

There has been explosive growth in the consumer electronics market during the past decades. PDA's, portable multimedia players, digital cameras and video recorders are prevalent in our daily life. Their functions can even be combined in a very small cell phone. As the IC industry shifts from PC-centric to consumer electronics-centric, digital technologies are no longer solving all the problems. Electronic devices integrating more functions, such as mixed-signal and RF, have become new challenges to the IC industry. System-on-a-chip (SoC) design and built-in self-test of mixed-signal circuits are two enabling technologies behind the integration of these functions and are of great interest to the industry and the academia.

When digital testing has been studied for long time, testing of analog and mixed-signal circuits is still in its development stage. Many methods proposed in 1980's or even earlier are still being used by engineers and researchers. Existing solutions for testing analog and mixed-signal (AMS) circuits have two major problems. First, the test cost is high. This has become a strategic problem to many large circuit manufacturers and led to serious discussions. Second, it is more and more challenging to improve the test capability of existing methods to keep up with the performance of the fast-evolving mixed-signal products demanded on the market. Furthermore, there is lack of effective methods for on-chip test of mixed-signal integrated systems. The *International Technology Roadmap for Semiconductors (ITRS)* identified mixed-signal testing as one of the most daunting SoC challenges [1].

The digital-to-analog converter (DAC) serves as the interface between the digital processing functions and analog signals. As the SoC design style getting more and more popular and

requirements for high-quality AMS circuitries continuously going up, the demand for high-performance DACs is growing rapidly. The *ITRS* indicates that "... digital-to-analog conversion performance becomes increasingly important as it opens the door to new high-volume but low-cost applications." World's leading AMS integrated circuits companies, such as Analog Devices, Texas Instruments and National Semiconductors, are all manufacturing high-speed high-resolution DACs for applications such as wireless communications and digital signal processing. The best commercial parts, such as AD9779 from ADI and DAC5687 from TI, have 16-bit resolutions and more than 500 MSPS update rates. The next generation products with better performance are currently under working and will come to the market very soon [10].

Along with the advancement in DAC performance, there are consequently new needs in DAC design and testing. It is well known that DAC testing is more challenging than ADC testing, as DACs usually have higher resolutions and speeds than ADCs. Measurement devices used in conventional DAC testing methods should have better performance and run faster than the device under test (DUT) to provide accurate characterization results. It is a nontrivial task to manufacture sufficiently fast and accurate instruments for testing the current and future highest-performance DACs. For example, to get the linearity of a 16-bit DAC at a 500 MSPS rate, the tester should have better than 18-bit accuracy, about 110 dB in dynamic range, over a frequency range of at least 1 GHz. State-of-the-art data acquisition techniques have difficulties to achieve such performance.

Our work is targeting at providing cost-effective solutions to the high-performance DAC testing problem. It is supported by the Semiconductor Research Corporation (SRC) at the member companies' request. We have come up with and investigated a novel method of

using low-accuracy instruments to test high-performance DACs. Our study shows that high-resolution DACs can be accurately tested by using low-resolution ADCs with appropriate voltage dithering. We discussed our ideas and researches with engineers from the industry at the SRC annual review, where leading mixed-signal IC companies were well represented, and received very positive evaluations [8]. Some of our results were also summarized in a paper presented at TECHCON 2005. The paper shows that 14-bit DACs can be tested to 1-LSB accuracy by using 6-bit ADCs [11] and it received an award judged by a panel of industry experts in the area of analog and mixed-signal test. Because of the availability of very high-speed low-resolution ADCs, this approach provides a potential solution to the testing problem for high-speed high-resolution DACs.

## **II. Existing Methods for DAC Testing**

Bench test plays important roles in design development, parameter tuning, debugging, and product validation stages of a DAC, while production test measures the specifications, sifts good, bad and marginal parts, and enables calibration for improving the performance of a DAC. An efficient testing method with high accuracy, short test time and low cost is very necessary for both of the two cases.

There are many well developed and widely adopted methods existing for bench and production test of different types of DACs. Quasi-static linearity and low-frequency dynamic performance of medium and low-speed DACs can be measured by using sigma-delta or dual-slope ADCs [9]. These types of ADCs can have very high accuracy, for instance, more than 20 bit linearity or 120 dB SFDR, but their sampling speed is inherently not high as limited by their respective architectures. High-frequency spectral test of communications DACs is usually done by using spectrum analyzers. Spectrum analyzers' dynamic range is affected by

their nonlinearity and distortion and is usually limited to less than 90 dB or lower for some specific measurements. Notch filters are sometimes used to kill the dominant fundamental component to reduce the nonlinearity and distortion. Also spectrum analyzers will need long time to generate a complete spectrum over a wide frequency range with a small resolution bandwidth, and they do not provide any time domain information of the measured signal.

Some other DAC testing approaches have been studied and reported. An on chip pass-or-fail testing approach for DACs using accurate reference voltages and a precision gain amplifier was presented by Arabi, Kaminska, and Sawan [2]. An approach of using a DAC's static nonlinearity to characterize its intermodulation errors was introduced by Vargha, Schoukens, and Rolain [3]. This approach is useful if the intermodulation errors are mainly from static nonlinearities, which is true at low frequencies. Rafeeqe and Vasudevan proposed an improved built-in self-test (BIST) scheme for DACs using an accurate sample-and-subtract circuit, a linear VCO, and a stable clock counter [4]. An overall review of existing built-in self-test approaches for DACs can be found in their paper.

In spite of these efforts, testing of high-speed precision DACs remains as a problem. It puts stringent requirements on the testing instruments. Linearity and stability of measurement devices should be better than the resolution of a DAC under test. It is also desirable to have a test structure that runs as fast as the DAC under test to conduct real-time testing and reduce the total test time. Production testing of DACs with higher than 1 GSPS update rate and better than 90 dB SFDR is a coming challenge in the very near future. Furthermore, the problem of on-chip testing for high-performance DACs is of interest and still open. Calibration techniques have proven to be effective in significantly improving a DAC's performances [5]. For effective calibration, an accurate characterization of the DAC is

always necessary and often times carried out by using precision instruments such as off-chip high-resolution ADCs [6, 7]. If a highly accurate and stable DAC testing circuitry can be built on-chip, it will enable integrated self-calibration for DACs in an SoC design.

### III. Linearity Specifications of DACs

The widely used terminologies for characterizing a DAC's static linearity are the same as those used for ADCs, the integral nonlinearity (*INL*) and the differential nonlinearity (*DNL*). Various definitions for *INL* and *DNL* exist, one slightly different from another. We will use the definition based on a fit line connecting the smallest and largest DAC output voltages. The *INL* is defined as the largest difference between the true transfer curve and the fit line of a DAC, and the *DNL* as the maximum error of the true increments between two consecutive outputs with respect to their averaged value. By using this definition, an  $n$ -bit DAC's *INL* at code  $k$  can be written as

$$INL_k = (N - 1) \frac{v_k - v_0}{v_{N-1} - v_0} - k \text{ (LSB)}, k = 0, 1, \dots, N - 1, \quad (6.1)$$

where  $N = 2^n$  and  $v_k$  is the output voltage associated with  $k$ . The unit LSB, standing for the least significant bit, is the averaged voltage increment,

$$1 \text{ LSB} = \frac{v_{N-1} - v_0}{N - 1}. \quad (6.2)$$

$INL_0$  and  $INL_{N-1}$  are equal to 0 under this definition, which is a straightforward result of the fit line definition. The expression of *INL* is

$$INL = \max_k \{ |INL_k| \}. \quad (6.3)$$

Definitions of code-wise and overall *DNL* are

$$DNL_k = (N-1) \frac{v_k - v_{k-1}}{v_{N-1} - v_0} - 1 \text{ (LSB)}, k = 1 \dots N-1, \quad (6.4)$$

and 
$$DNL = \max_k \{ |DNL_k| \}. \quad (6.5)$$

One of the widely used dynamic performance specifications for DACs is the spurious free dynamic range (SFDR), when a digital sine wave at a specific frequency is used as the input. The SFDR is defined as the difference between the amplitude of the fundamental component and that of the maximum spurious component in the spectrum of the DAC output.

*INL* and *DNL* of a DAC are usually tested by measuring the output voltages  $v_k$  and calculated as in (6.1) to (6.5). The SFDR of a DAC can be measured by sampling the output waveform with a high-accuracy digitizer and applying FFT to the sampled output sequence. It is obvious that to get both of these specifications we need accurate measurement of the DUT output.

#### IV. Test Precision DACs Using Low-Resolution ADCs with Dithering

This work proposes a DAC testing approach with two goals, short test time and high accuracy. Flash ADCs have the fastest conversion rate among the data acquisition devices, so it is used in our approach to quantize the output voltage of high-speed DACs. Flash ADCs' resolutions are usually less than 8 bit because of the architecture limitation. The concern of using a low-resolution ADC in high-performance DAC testing is that it will introduce large quantization errors and its transition levels are not accurate. A dithering technique will be used to increase the resolution of the test, while final accuracy of the test result will be guaranteed by an effective data processing algorithm applied to the DAC output quantized by the low-resolution ADC with dithering.

### *Test Setup and Data Capture*

The proposed strategy uses a low-resolution measurement ADC (m-ADC) and a dithering DAC (d-DAC) to test a high-performance DAC, usually called the device under test (DUT), see the block diagram in Figure 6.1. The d-DAC can be specifically designed for the testing purpose, or simply another device from the same product family of the DUT. The output of the d-DAC will be scaled by a small factor  $\alpha$  and added to the output of the DUT as a dithering component. The dithered output of the DUT will be quantized by the m-ADC.

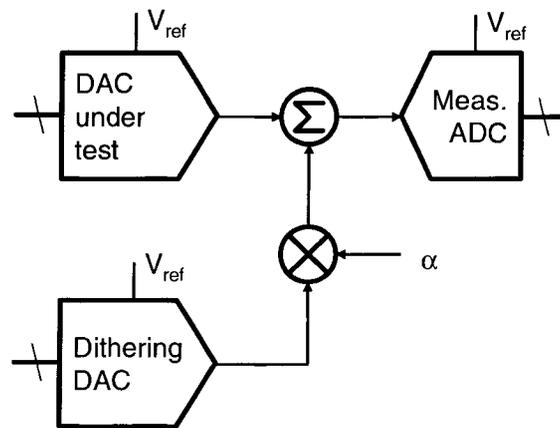


Figure 6.1 Block diagram of the proposed method.

In the test, the DUT will repeatedly generate a waveform of interest. During each period of the waveform, the d-DAC will provide a distinct but constant dithering voltage to be added to the DUT output. The m-ADC will quantize a certain number of periods of the waveform with different dithering levels. Because of the different dithering levels, the m-ADC's output codes associated with one output voltage of the DUT will be slightly different from one run to the next. Specifically, the output code associated with a voltage right smaller than an ADC

transition level will increase when the dithering level increases. See Figure 6.2, where the DUT output waveform is triangular in this example.

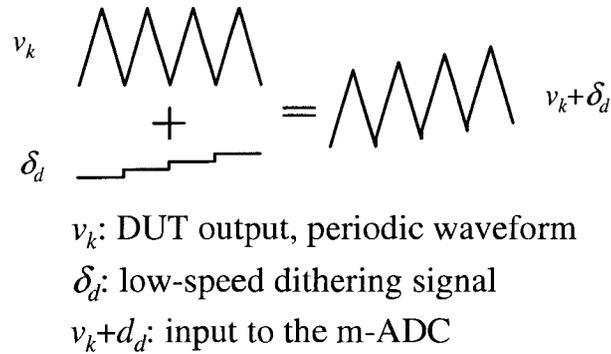


Figure 6.2 DUT's output with dithering.

The output codes of the m-ADC can be put into a two-dimensional structure as shown in Table 6.1, where  $N_d$  and  $N_D$  are the numbers of output levels of the d-DAC and the DUT, respectively. Assuming a 6-bit ADC is used, the output code will range from 0 to 63. Each column in Table 6.1 is associated with one d-DAC input, or alternatively speaking a dithering level, and collected from one period of the waveform generated by the DUT, a ramp in this example. On the other hand, each row in the table comes from one DUT output voltage  $v_k$  with different dithering levels. The scaling factor  $\alpha$  is chosen for the d-DAC so that the dithered voltages associated with any one specific DUT output will cover at least one complete code bin of the m-ADC. A dithering range of 3 *LSB* of the m-ADC is enough to guarantee this feature for a low resolution ADC, for which the *DNL* is usually much less than 0.5 *LSB*. Given this property, the output codes of the m-ADC associated with any input code to the DUT, a row in Table 6.1, will always consist of at least 3 distinct codes. These output codes will be used to calibrate the m-ADC and test the DUT.

Table 6.1. Output of the m-ADC vs. Input to the d-DAC

d-DAC input \ DUT output	1	2	3		$d$	...				$N_d$
$v_1$	2	2	2	3	3	3	3	4	4	4
$v_2$	2	2	3	3	3	3	3	4	4	4
$V_3$	2	2	3	3	3	3	4	4	4	4
...										
$v_k$	40	41	41	41	41	42	42	42	42	43
...										
$v_{ND-1}$	60	60	60	61	61	61	61	61	62	62
$v_{ND}$	60	60	61	61	61	61	61	62	62	62

### ***Proposed DAC Test Method***

Without dithering, a DUT output voltage  $v_k$  will be quantized by the m-ADC as a code  $j$  if  $T_j < v_k \leq T_{j+1}$ , see Figure 6.3 (a).  $T_j$  is the transition level of the m-ADC between code  $j-1$  and  $j$ . Based on this output, we can only have a rough estimation of the DUT output as

$$\hat{v}_k = T_j = v_k + q_k, \quad (6.6)$$

where  $q_k$  is the quantization error introduced by the m-ADC.  $q_k$  can be as large as hundreds of LSBs for the DUT, since the m-ADC's resolution is much lower than that of the DUT. This is why low-resolution ADCs are not used to test precision DACs. In our approach, dithering is used to improve the resolution and minimize the quantization error.

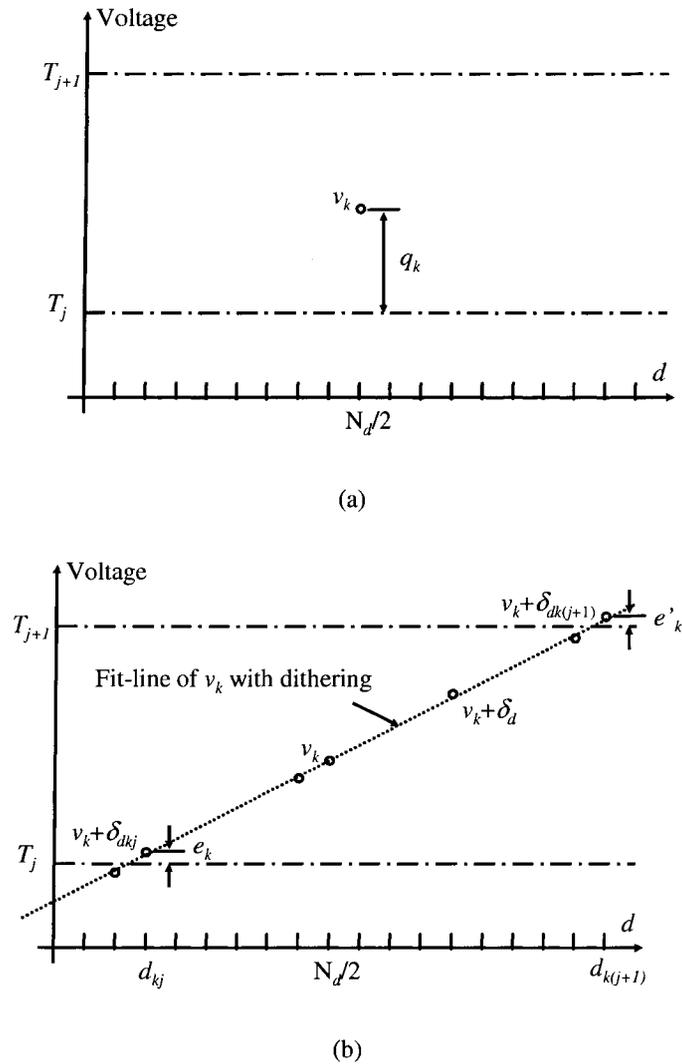


Figure 6.3 DAC testing with dithering.

As in Figure 6.2, the m-ADC will quantize many dithered copies of the DUT output voltage,  $v_k + \delta_d$ , where  $\delta_d$  is the  $d$ -th dithering level, and the associated output codes form a row in Table 6.1. The quantized output may change as the dithering voltage increases. At a specific dithering level,  $\delta_{dkj}$ , the output code of the m-ADC changes from  $j-1$  to  $j$ , as the dithered

voltage changes from less than  $T_j$  to larger than  $T_j$ , see Figure 6.3 (b). Then we have a new estimate of the DUT output as

$$\hat{v}_k = T_j - \delta_{dkj}, \quad (6.7)$$

where  $\delta_{dkj} = d_{kj}/N_d - 1/2$  is the dithering level. For linearity test, the unit of the dithering voltage does not affect the final accuracy, so we use their linearly code-dependent part and normalize it with  $N_d$ , and 1/2 is taken off for representing differential voltage dithering. It is assumed in the discussion that the dithered voltages associated with  $v_k$  are uniformly spaced over the whole dithering range, a small interval around  $v_k$ . This is reasonable for a small scaling factor  $\alpha$ , because any nonideality in the d-DAC is dramatically scaled down and becomes negligible as compared to the errors of the DUT. Further discussions on this assumption will be provided later in the performance analysis section.  $e_k$  in Figure 6.3 (b) is the error between the estimate of  $v_k$  in (6.7) and its true value. In this case,  $e_k$  is limited by the step size of the dithering voltages and can be made very small by a sufficient number of steps in a fixed range.

We have shown that we can effectively increase the resolution of testing by dithering, but test accuracy is not guaranteed because we do not know the exact value of  $T_j$  in (6.7). If we appropriately set the dithering range, there will be more than one transition in the m-ADC's output codes associated with  $v_k$ . In Figure 6.3 (b), the output code changes from  $j$  to  $j+1$  at the dithering level  $\delta_{dk(j+1)}$ . This gives us another estimate as

$$\hat{v}_k = T_{j+1} - \delta_{dk(j+1)}, \quad (6.8)$$

where  $\delta_{dk(j+1)} = d_{k(j+1)}/N_d - 1/2$ . For each  $v_k$ , we can have at least two equations like (6.7) and (6.8). There are totally  $2*N_D$  such equations in  $N_D$   $v_k$  variables and  $N_{ADC}-1$   $T_j$  variables, for  $k$

$= 1, 2, \dots, N_D$ . Since the m-ADC's resolution is lower than that of the DUT,  $N_{ADC}-1$  is smaller than  $N_D$ . Therefore, the DUT's output voltages and m-ADC transition levels can be simultaneously solved from the  $2N_D$  linear equations, under the Least Squares sense when necessary.

The DUT's linearity specifications can be calculated from the estimated  $v_k$ 's using the equations or methods discussed in Section III. The estimation errors in (6.7) and (6.8),  $e_k$  and  $e_k'$ , are bounded by the dithering step size and can be reduced by applying a small dithering increment between two consecutive dithering levels. If we make this increment much smaller than 1 LSB of the DUT, the final test result based on the estimated values will have very high accuracy.

#### ***Implementation of the Proposed Algorithm***

It is inefficient to solve  $2N_D$  equations simultaneously, especially when  $N_D$  is large. From investigating the equations' structure, we find that  $v_k$ 's and  $T_j$ 's can be calculated from the equations by applying a series of simple algorithmic operations.

We can first calculate one value of the m-ADC's  $j$ -th code bin width,  $W_j = T_{j+1} - T_j$ , from (6.7) and (6.8) as

$$\hat{W}_j^{(k)} = \delta_{dk(j+1)} - \delta_{dkj}. \quad (6.9)$$

For some different  $v_k$ 's, we may have other values for  $W_j$ . The final estimate is the average over these values,

$$\hat{W}_j = \text{mean}_k \{ \hat{W}_j^{(k)} \}. \quad (6.10)$$

Transition levels of the m-ADC can then be calculated by taking cumulative summations of these code bin widths as

$$\hat{T}_j = \sum_{i=0}^{j-1} \hat{W}_i. \quad (6.11)$$

Without affecting the linearity of the m-ADC,  $T_0$  is chosen to be 0 in (6.11). The DUT output voltage can be calculated from (6.7) and (6.8) as

$$\hat{v}_k = \underset{j}{\text{mean}}\{\hat{T}_j - \delta_{dkj}\}, \quad (6.12)$$

where the average is taken over all the  $T_j$ 's covered by the dithered voltages of  $v_k$ .

The proposed DAC testing strategy with low-resolution ADCs can be summarized as following steps.

- DAC under test generates periodic waveform with different dithering levels;
- ADC quantizes the dithered waveform;
- Estimate ADC transition points using (6.9)-(6.11);
- Calculate DAC output voltages using (6.12);
- Characterize DAC performance based on the measured waveform as discussed in Section III.

## V. Performance Analysis and Other Issues

This section provides some performance analysis and implementation considerations of the proposed high-performance DAC testing strategy.

### *Performance Analysis*

An intuitive observation of the proposed algorithm is that the test result will be more accurate, if the m-ADC has higher resolution or the d-DAC can provide more distinct dithering levels with high resolution and linearity. Detailed analysis is in agreement with this observation and the test accuracy of the proposed method can be summarized by the following equation as

$$A_{test} = n_{ADC} + ENOB_{dith} - \log_2 \alpha, \quad (6.13)$$

where  $A_{test}$  is the desired test accuracy in bit,  $n_{ADC}$  is the m-ADC's resolution,  $ENOB_{dith}$  represents the effective number of bits of the d-DAC in linearity, and  $\alpha$  is the scaling factor in m-ADC's LSB. In (6.13), we assume the d-DAC has a sufficient resolution so that the error introduced by dithering is dominantly dependent on d-DAC's linearity and the effect of quantization noise is neglected. This assumption is reasonable as the resolution is comparatively easy to get, but the linearity of a DAC is limited by the design and fabrication technologies. In (6.7) and (6.8), we assume the d-DAC is linear while it is actually not, so nonlinearity of the d-DAC will affect the final test accuracy. Therefore, the d-DAC can only provide accuracy improvement equal to its linearity in (6.13). However, if the d-DAC can be accurately characterized or calibrated, the d-DAC can improve the test accuracy even more and the  $ENOB$  term in (6.13) can be replaced by the d-DAC's resolution  $n_{d-DAC}$ .

Using the above equation, we can determine the requirement on the test devices for specific test accuracy. For example, if we have a 6-bit ADC and the dithering range is 4 LSB at the 6-bit level, we need following d-DAC linearity to achieve 14-bit accuracy

$$ENOB_{dith} = A_{test} - n_{ADC} + \log_2 \alpha = 14 - 6 + 2 = 10 \text{ (bit)}. \quad (6.14)$$

Eq. (6.14) means a 6-bit ADC can provide 14-bit test accuracy if 10-bit linear dithering is available.

### ***Circuit Implementation***

Assuming the DUT and the d-DAC are fully-differential current-steering DACs, a practical realization of the proposed test scheme is shown in Figure 6.4. This circuit can also be used with single-ended circuits after some simple modifications. The scaling of d-DAC's output

and the dithering summation can be physically implemented with the two  $\pi$  networks of resistor  $R_{+/-}$ ,  $R_{s+/-}$  and  $R_{d+/-}$ . If the d-DAC and the DUT are the same product,  $R_{+/-}$  and  $R_{d+/-}$  are also the same. To correctly match DACs' output impedance and set the scaling factor, the resistance values need to be appropriately chosen such that

$$\begin{aligned} R_{+/-} \parallel (R_{s+/-} + R_{d+/-}) &= R_0; \\ R_{+/-} / (R_{s+/-} + R_{+/-}) &= \alpha, \end{aligned} \quad (6.15)$$

where  $R_0$  is the specified load resistance of the DACs. The above conditions can uniquely determine the nominal values of  $R_{+/-}$ ,  $R_{s+/-}$  and  $R_{d+/-}$ . Since resistive networks are usually very linear, it will not introduce extra nonlinear errors in dithering, which is necessary to guarantee the m-ADC characterization.

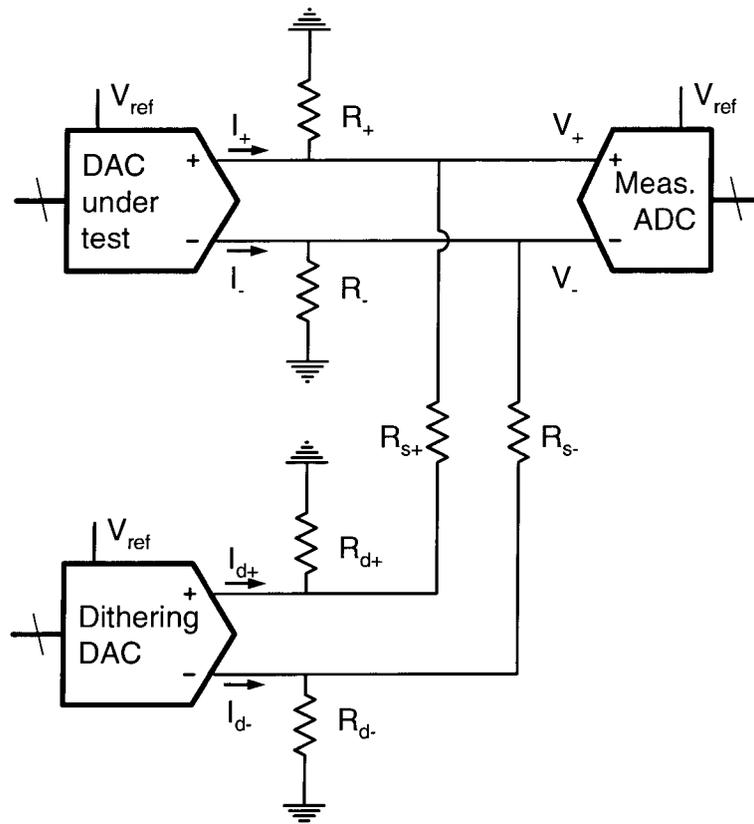


Figure 6.4 Circuit implementation of the proposed test scheme.

### ***Data Storage and Transition Identification***

If the number of required dithering levels is large for high test accuracy from (6.13), the size of Table 6.1 can be very large correspondingly. For instance, if the DUT has 16384 output voltages to be tested and the d-DAC needs to generate 1024 dithering levels, there will be approximately 16 M output codes to be stored. This storage requirement is nontrivial but still doable for production testing, but it is usually too much for an on-chip testing application. To reduce the storage requirement, the output codes from the m-ADC can be saved in a histogram style. This operation does not hurt the testing performance as we will see shortly, since the dithering information we need are all contained in the histogram data.

Table 6.2. Histogram storage for m-ADC's output codes

DUT output	$m$	$H_m$	$H_{m+1}$	$H_{m+2}$	$H_{m+3}$
$v_1$	2	3	4	3	0
$v_2$	2	2	5	3	0
$v_3$	2	2	4	4	0
...					
$v_k$	40	1	4	4	1
...					
$v_{ND-1}$	60	3	5	2	0
$v_{ND}$	60	2	5	3	0

Table 6.2 shows the histogram storage of the test results. For a row in Table 6.1 associated with a DUT output voltage, the corresponding row in Table 6.2 saves the minimum m-ADC output code  $m$ , and the number of hits for  $m$ ,  $m+1$ , ..., up to the maximum output code,

generated by the dithered voltages. The local histogram data can be used to estimate m-ADC transition levels as (6.9) to (6.11). The code bin width represented by the histogram counts is

$$\hat{W}_j^{(k)} = H_j^{(k)} / N_d, \quad (6.16)$$

when the  $j$ -th code bin is completely covered by the dithered voltage of  $v_k$ . And the  $\delta_{dk,j}$ 's in (6.12) can be calculated from the histogram as

$$\delta_{dkj} = \sum_{m < j} H_m^{(k)} / N_d. \quad (6.17)$$

So the testing algorithm carries out as before. However, the number of memory cells is dramatically reduced. Usually each row in Table 6.2 will contain one minimum code and five histogram counts at most. Only 100 K memory cells are needed for testing 16384 points as in the previous example. It is reduction of more than 150 times as compared to Table 6.1.

Furthermore, the time for capturing the data as presented in Table 6.2 can be dramatically reduced by using binary search instead of linearly incremental search. Binary search will identify all the dithering levels at which transitions of the output of the m-ADC happen, associated with a specific DUT output voltage, which gives the  $d_{k,j}$  information. These data can be summarized as in Table 6.3. It's straightforward to determine the information required for m-ADC and DUT identification from Table 6.3. If a 10-bit d-DAC is used in test, the number of samples required for one  $v_k$  is less than 40 when using binary search to determine at most 4 transitions. After the m-ADC is identified in one DAC test, this average number of dithering samples for one  $v_k$  can be even reduced to less than 15 for following tests. It is a reduction of more than 50 times in testing time as compared to 1024-level linear dithering. Of course, binary search is only applicable to quasi-static test.

Table 6.3. Output transitions of measurement ADC's output codes

DUT output	$m$	$d_{k,m}$	$d_{k,m+1}$	$d_{k,m+2}$
$v_1$	2	3	7	
$v_2$	2	2	7	
$v_3$	2	2	6	
...				
$v_k$	40	1	5	9
...				
$v_{ND-1}$	60	3	8	
$v_{ND}$	60	2	7	

Since both the number of the samples, equivalently testing time, and the amount of memory cells required for each DUT output is very small, this approach is cost-effective and can be practically implemented in either production or on-chip testing applications.

#### ***Other Test with the Proposed Algorithm***

The DUT can generate other waveforms than a ramp or triangular signal, while the proposed method is still applicable. We just need to change the first column of the Table 6.1, 6.2 or 6.3 correspondingly to the new waveform, for example, a sine wave. After we recover the waveform in the time domain by using the proposed method, following-up processing, such as FFT, can be taken to determine the DUT performance. The d-DAC's output also does not need to change in a ramp style. Sine wave style dithering can also be used. We just need to modify (6.9) to an appropriate form. The idea is similar to using sine waves in the histogram test of ADCs.

## VI. Simulation Results

14-bit DACs were tested in simulations. The proposed algorithm is not dependent on the architecture of the DUT. We chose the thermometer coded current-steering DAC as the DUT, since it has the largest number of independent errors from each of the current sources. Both static and spectral testing situations are simulated.

### *Quasi-Static Testing*

A 6-bit flash ADC was used in measurement. The  $INL$  of the m-ADC is about 0.3 LSB at the 6-bit level. Its true  $INL_k$  is plotted in black in Figure 6.5. Transition levels of the ADC were first measured as discussed in Section IV. The estimated  $INL_k$  curve is plotted in red in Figure 6.5 as well. The two curves are nearly identical so that we can only see one curve on the plot. The estimation errors are about 0.0002 LSB at the 6-bit level, which is sufficient for 14-bit DAC testing. The DUT has an  $INL$  of 14 LSB, and its true  $INL_k$  is plotted in black in Figure 6.6 (a). The DUT has about 10 bit linearity, which is realistic according to [2]. Another 12-bit DAC of the same structure is used to provide 4096 dithering levels. This d-DAC has about 10-bit linearity as well, with an  $INL$  of 3 LSB at the 12-bit level.

In simulation, the dithering range was chosen to be 3.6 LSB of the m-ADC, and a noise is added to the input of the m-ADC with a standard deviation equal to 0.25 LSB at the 14-bit level. Based on calculation in Section V, a 6-bit ADC, 10-bit linear dithering and the above dithering range can provide 14-bit test accuracy. The estimated  $INL_k$  of the DUT is plotted in red in Figure 6.6 (a). The estimated curve matches the true  $INL_k$  curve very well. The estimation errors for all codes are plotted in Figure 6.6 (b). The  $INL_k$  of the DUT was tested to better than 1 LSB accuracy at the 14-bit level. It is in agreement as what we expected.

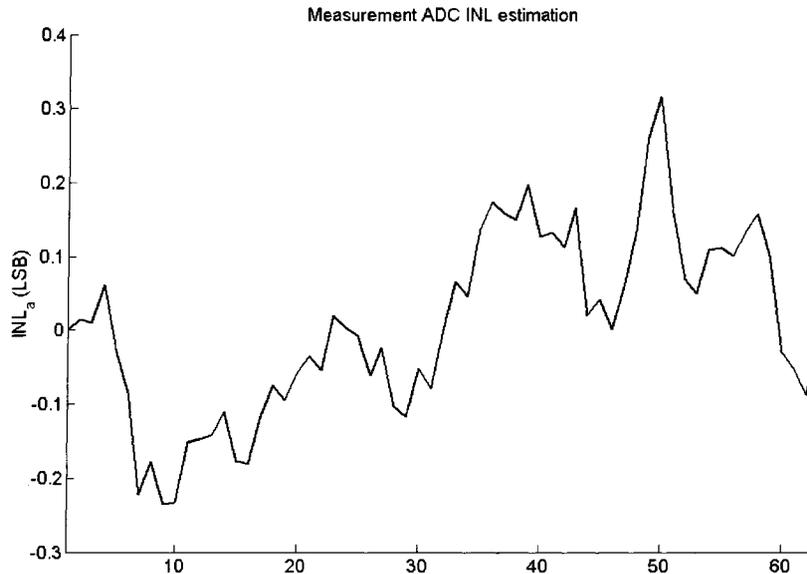


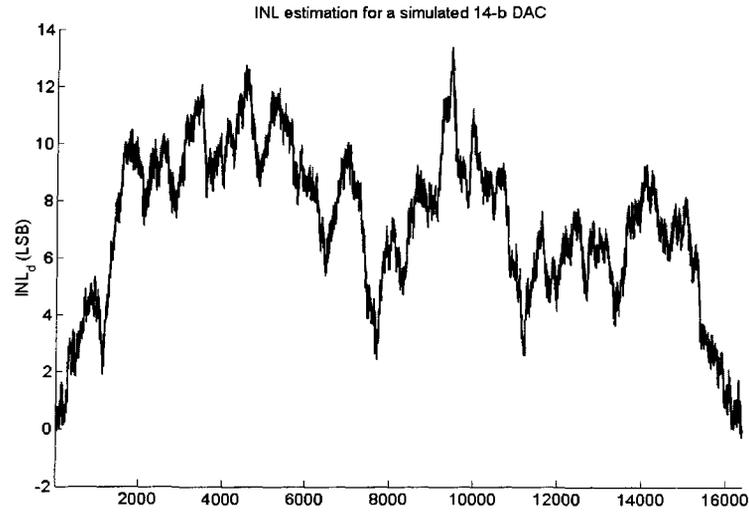
Figure 6.5 True and estimated  $INL_k$  of the m-ADC.

### ***Spectral Performance Testing***

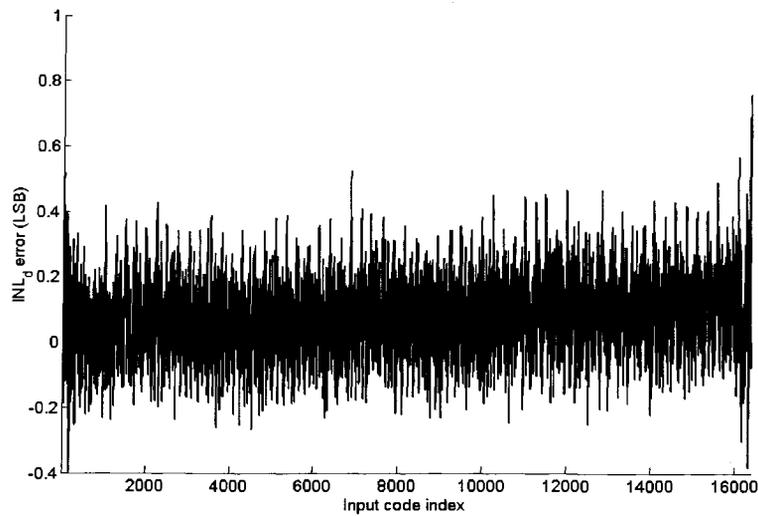
A single tone test was done on a 14-bit DAC for testing its SFDR. The m-ADC was still a 6-bit flash ADC. 512 dithering levels were used. A waveform length of 8196 samples, containing 111 periods of a sinusoidal signal, was used in simulation. A Gaussian noise with a standard deviation of 1 LSB at the 14-bit level was added to the sine wave output of the DUT. The typical SFDR of simulated DACs was set at 85 dB.

The spectrum of the true output sine wave of a DUT is plotted on the top of Figure 6.7, for which the SFDR was read as 86.85 dB. By using the proposed testing algorithm, the spectrum was estimated and plotted on the bottom of Figure 6.7, where the SFDR was estimated as 87.19 dB. The two spectrums match very well at the significant frequency

components and the true and tested SFDR readings are within a 1-dB accuracy window, when a 6-bit ADC was used in measurement.



(a)



(b)

Figure 6.6  $INL_k$  estimation of a 14-bit DAC. The true and estimated  $INL_k$  are plotted in black and red in (a), respectively. Estimation errors are in (b).

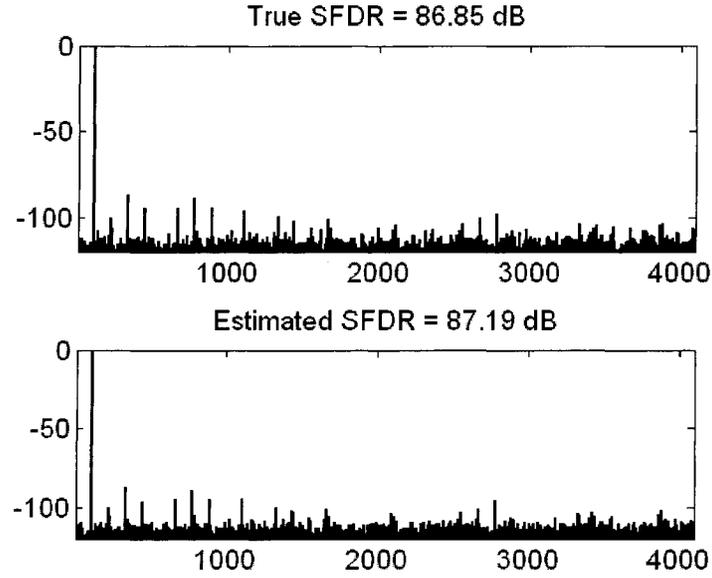


Figure 6.7 SFDR test of 14-bit DAC with 6-bit ADC and dithering.

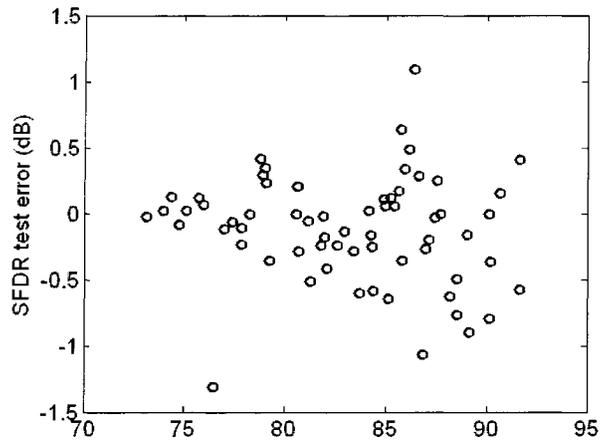


Figure 6.8 SFDR test error with 6-bit ADC.

To further validate the performance of the proposed testing strategy, same simulation was repeated on 64 different 14-bit DUTs. The SFDR estimation errors are plotted in Figure 6.8, with the true SFDR as the horizontal axis. Most of the SFDR testing errors are less than 1 dB

and all of the errors are within 1.5 dB for SFDR ranging from less than 75 dB to more than 90 dB.

## VII. Experimental Results

Some experiments were done to validate the performance of the proposed DAC testing algorithm with low-resolution ADCs. We used a Conejo baseboard by Innovative Integration in our experiments. This board has four 16-bit DACs, four 14-bit ADCs, and a TI DSP on chip. As the total number of samples for the dithered measurement is limited by the data storage capability of the board, testing of very high-resolution DACs were not carried out, but we can show that the concept of the proposed method is working by the following results.

### *Spectral Test*

A sine wave signal with a synthesized  $-60$  dB second harmonic component was tested. 2048 samples were taken on a waveform containing 11 periods. The signal was first measured by using a 14-bit ADC. The FFT spectrum is plotted in Figure 6.9. The measured SFDR was 59.91 dB. It will be used as a reference to evaluate the performance of the proposed method.

The signal was then tested by using a low-resolution ADC with dithering. Since there was no 6-bit ADC in our test setup, we used the high-resolution ADC on the Conejo baseboard and truncated the least significant bits of the output to get 6-bit digitizing results. Although the original ADC had a very high performance, its quantization effects after truncation could easily mask the true spectral errors in the signal as shown in Figure 6.10. There are many spurious components have larger than  $-60$  dB magnitudes.

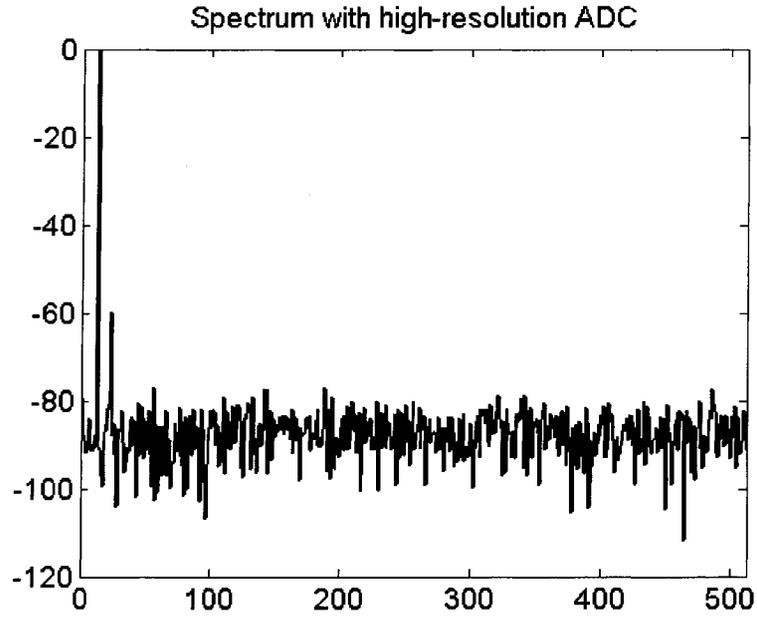


Figure 6.9 Estimated spectrum using a 14-bit ADC.

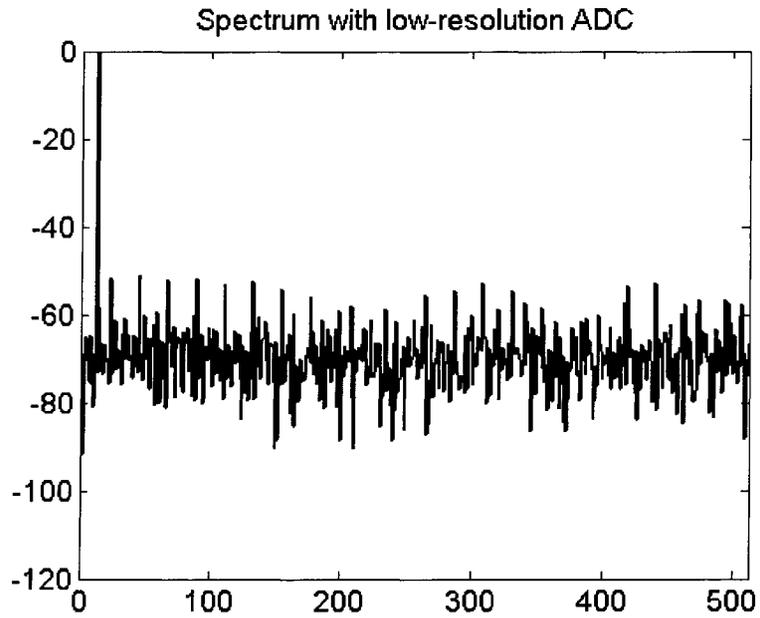


Figure 6.10 Estimated spectrum using 6-bit ADC w/o dithering.

The sine wave was then repeated 256 times and dithered by a sine wave generated by the d-DAC, using the single-ended version of the resistor summing circuitry in Figure 6.4. The range of the dithering signal was 5% of the output of the DUT. The dithering signal contained 257 periods during the total of  $2048 \times 256$  samples on the DUT output, so that each DUT output experienced a complete period of sine wave dithering. The dithered output was quantized by the pseudo 6-bit ADC. The proposed algorithm was used to draw the FFT spectrum, which is plotted in Figure 6.11. The estimated SFDR was 59.23 dB. This number is very close to the true value. The -60 dB second harmonic was identified. Other spurious terms were at least 20 dB smaller in the test result, which is natural considering that the original resolution of the DAC is 16-bit.

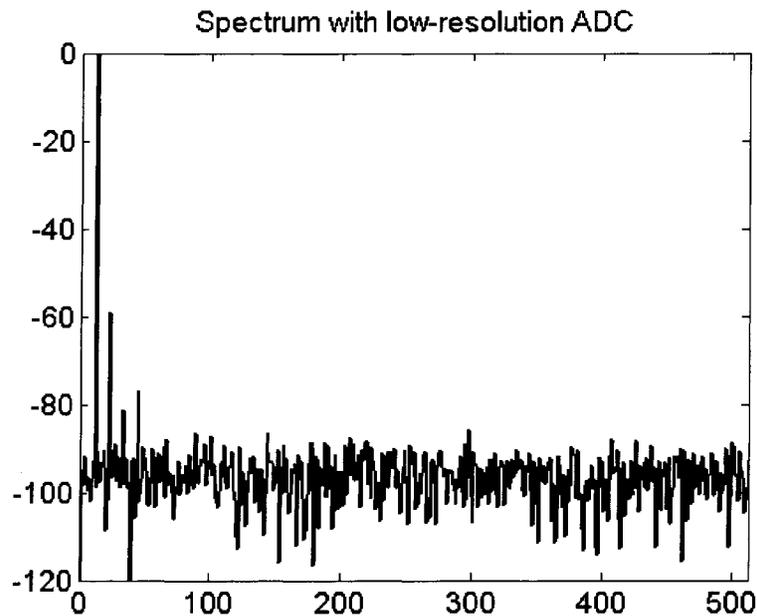


Figure 6.11 Estimated spectrum using 6-bit ADC w. dithering.

It is obvious that the noise level in the test results with the 6-bit ADC is about 10 dB lower than that with the high-resolution ADC. This benefit comes from the averaging effect of the large number of dithering.

### *Quasi-Static Test*

A pseudo 10-bit DAC was generated by using the 16-bit DAC on the Conejo baseboard for  $INL_k$  testing. An extra sinusoidal shape  $INL_k$  was purposely introduced.  $INL_k$  of the 10-bit DUT was measured using a 14-bit ADC many times. The mean value of  $INL_k$  from different measurements, when the noise effect is averaged out, would be used as a reference for evaluating the performance of the proposed method. It is plotted in Figure 6.12. The 10-bit DAC was then tested by using the proposed algorithm with a 6-bit ADC, from truncation, and 512 level dithering. The dithering range is set to be about 5% of the m-ADC input range. The measured  $INL_k$  is plotted in Figure 6.13.

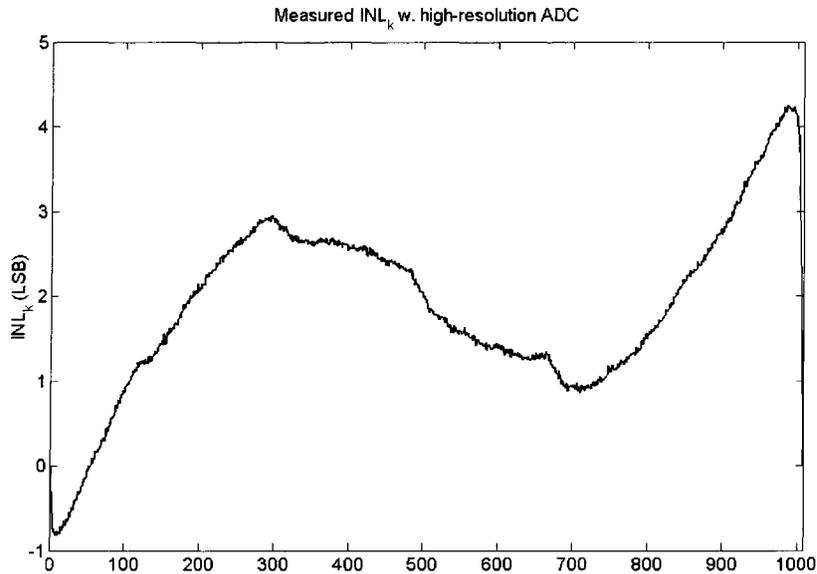


Figure 6.12  $INL_k$  measurement with a high-resolution ADC.

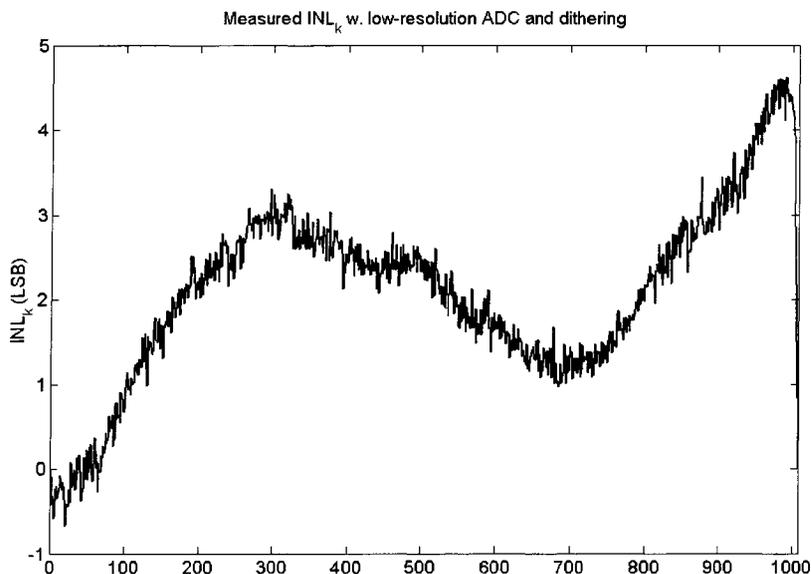


Figure 6.13  $INL_k$  measurement with 6-bit ADC and 9-bit dithering.

It can be calculated from (6.13) that a 6-bit ADC plus 9-bit dithering and the dithering range we used can provide about 13-bit test accuracy. From Figure 6.12 and 13, we can observe that  $INL_k$  measured by the 14-bit ADC and the proposed method are very close to each other. Therefore the algorithm works and achieves the performance we predicted.

### VIII. Conclusions

An effective DAC testing approach is presented in this paper. This approach uses high-speed flash ADCs and dithering to test high-resolution DACs. Simulation results show that  $INL_k$  of 14-bit DACs can be tested to 1-LSB accuracy by using a 6-bit ADC and 12-bit dithering, and spectrums of signals with more than 85-dB SFDR can be measured to 1 dB accuracy. Experimental results also supported the effectiveness of the algorithm in DAC testing using low-resolution ADCs. Because the proposed algorithm doesn't require high-precision test

instruments, it provides a potential practical solution to the problem of production and on-chip testing of high-speed high-resolution DACs.

### Acknowledgements

This work is supported by an SRC project through a National Semiconductor's customer funding. The author would like to thank Dr. Turker Kuyel with Texas Instruments for the comments of current DAC testing paradigm in our communications. The authors also would like to thank Mr. Hanjun Jiang and Mr. Deepak Sahoo for their help in the experimental results generation.

### References

- [1] *International Technology Roadmap for Semiconductors*, 2001 and 2003 edition, available online: <http://public.itrs.net>.
- [2] K. Arabi, B. Kaminska, and M. Sawan, "On Chip Testing Data Converters Using Static Parameters," *IEEE Trans. VLSI System*, vol.6, no.3, pp. 409-418, September 1998.
- [3] B. Vargha, J. Schoukens, and Y. Rolain, "Static Nonlinearity Testing of Digital-to-Analog Converters," *IEEE Trans. Instru. Meas.*, vol. 50, no. 5, pp. 1283-1288, October 2001.
- [4] S. Rafeeqe K.P. and V. Vasudevan, "A built-in-self-test scheme for digital to analog converters," in Proc. 17th Intl. Conference VLSI Design, pp. 1027-1032, 2004.
- [5] D. Groeneveld, H. Schouwenaars, H. Termeer, and C. Bastiaansen, "A Self-Calibration Technique for Monolithic High-Resolution D/A Converters," *IEEE J. Solid-State Circuits*, vol. 24, pp. 1517-1522, December 1989.

- [6] A. Bugeja and B. Song, "A Self-Trimming 14-b 100-MS/s CMOS DAC," *IEEE J. Solid-State Circuits*, vol. 35, pp. 1841-1852, December 2000.
- [7] Y. Cong and R. Geiger, "A 1.5-V 14-bit 100-MS/s Self-Calibrated DAC," *IEEE J. Solid-State Circuits*, vol. 38, pp. 2051 -2060, December 2003.
- [8] D. Chen and R. Geiger, SRC annual review presentation, Atlanta, GA, in May 2005, <http://www.SRC.org>.
- [9] Personal communication with Turker Kuyel at Texas Instruments, Dallas, TX, in December 2004.
- [10] Personal communication with engineers at Analog Devices, Dublin, Ireland, in August 2004.
- [11] L. Jin, H. Haggag, R. Geiger, and D. Chen, "Precision DAC Test Using Low-Resolution ADCs and Dithering," in *Proc. 2005 SRC TECHCON*, Portland, OR, Oct. 2005.

## Chapter 7

### Conclusions

My colleagues and I have developed new methodologies for testing data converters using low-accuracy instruments with system identification and data processing algorithms.

A stimulus error identification and removal (SEIR) algorithm is developed for testing ADCs that uses two unknown nonlinear signals with a constant offset. A strategy, called center-symmetric interleaving (CSI), that minimizes the time-varying effect of environment nonstationarity on the test is proposed. The combination of the SEIR algorithm and the CSI strategy provides a solution to the challenging problem of testing high-performance ADCs, utilizing low-accuracy instruments in a realistic time-varying environment. This approach dramatically reduces the test time and cost, as the stimulus signal can be easily generated at very high speeds, and enables the built-in self-test of high-resolution ADCs, as a simple stimulus generator can be integrated on chip.

We developed another approach using a single nonlinear stimulus signal for testing ADCs with some widely adopted architectures, including cyclic and pipeline, based on the ADC's structure characteristics. The single signal test algorithm further simplifies the ADC testing problem, as compared to the SEIR algorithm, by making use of some readily available information. Since pipelined and cyclic ADCs are very popular in nowadays IC systems, this algorithm can have a significant impact on the ADC design practice.

We proposed a modified code-density algorithm that uses the Kalman filter to reduce the effect of errors on the histogram data. This algorithm can achieve the same level of accuracy as that of the conventional code-density algorithm but using a significantly smaller number

of samples, which means shorter test time and lower test cost. The new method can be used to enable testing of high-resolution ADCs with better coverage and reduce the time and cost of testing medium-resolution ADCs.

We have come up with and investigated a novel method of using low-accuracy instruments to test high-performance DACs. It is shown that high-resolution DACs can be accurately tested by using low-resolution ADCs with appropriate voltage dithering. Because of the availability of very high-speed low-resolution ADCs, this approach provides a potential solution to the testing problem for high-speed high-resolution DACs.

As the results included in the dissertation have shown, we can provide practical and accurate test solutions for high-performance data converters. Meanwhile, the proposed testing methodologies are very cost-effective, because of the low price and high speed of the utilized low-accuracy test instruments. These algorithms have the potential of serving as built-in self-test solutions integrated on chip and can be generalized to other mixed-signal circuitries. When incorporated with self-calibration, these algorithms can enable new design techniques for mixed-signal integrated circuits.

## **Acknowledgements**

There have been many people and organizations helping me during my Ph.D. work. I would like to sincerely thank all of them for their help and support, in particular to the following people and agencies:

- Dr. Degang Chen and Dr. Randall Geiger for their instructions everywhere on my research works.
- Dr. Robert Weber, Dr. Chris Chu, and Dr. Fritz Keinert for their help and suggestions on my work.
- All my colleague students for their collaborations to complete my Ph.D.
- 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.