USER-SMILE: Ultrafast Stimulus Error Removal and Segmented Model Identification of Linearity Errors for ADC Built-in Self-Test

Tao Chen, *Student Member, IEEE*, Xiankun Jin, *Member, IEEE*, Randall L. Geiger, *Fellow, IEEE*, and Degang Chen, *Fellow, IEEE*

***Abstract*— Linearity testing of analog-to-digital converters (ADCs) is very challenging and expensive due to the stringent linearity requirement on the stimulus and the extremely long test time. This paper introduces a novel method for ADC static linearity testing, allowing the stimulus linearity requirement to be significantly relaxed and the test time to be significantly reduced compared to the state-of-art histogram method. Two nonlinear but functionally related input signals are used as the ADC's excitation and a stimulus error removal technique is used to recover test accuracy. With a segmented non-parametric integral nonlinearity model, this method requires much fewer parameters to accurately represent the nonlinearity. The proposed algorithm has been extensively verified and correlated in simulations. This method not only enables low-cost production testing but can also be used for low-cost on-chip built-in self-test. This method is limited to ADCs with segmented architecture such as SAR ADCs, pipeline ADCs, and cyclic ADCs.**

***Index Terms*— Analog-to-digital converters, integral nonlinearity, differential nonlinearity, USER-SMILE, built-in self-test.**

I. Introduction

THE analog-to-digital converter (ADC) is one of the most important analog and mixed signal (AMS) components. The ADCs have been deeply embedded in the modern system-on-chip (SoC). With ever increasing applications in Internet of things (IoT) and automotive, the ADCs' volume has grown significantly. Testing these ADCs is necessary to guarantee the performance before shipment. However, it is challenging and expensive to fully test the ADC's performance due to various reasons. With such a large volume, reducing the test cost of ADCs becomes significant and necessary. In addition, for some critical applications such as automotive, aerospace and medical areas where reliability and safety requirements are extremely high, one-time test may not be enough since such applications usually last for decades and the degradation of performance or environmental changes may cause a severe influence on the system. There is a strong need to achieve built-in self-test (BIST) capability to not only reduce the test cost, but also guarantee the reliability of the ADCs. The testing of digital circuits has been well addressed with automatic test pattern generation (ATPG) and logic BIST. However, it remains a challenge to test the ADC on chip [1].

The ADC testing involves static linearity tests, such as integral nonlinearity (INL) and differential nonlinearity (DNL) and dynamic linearity tests, such as signal to noise ratio (SNR), total harmonic distortion (THD) and spurious free dynamic range (SFDR) [2]–[4]. Among these tests, the static linearity test is the most time-consuming one. To test the ADC static nonlinearity, the conventional histogram method uses a highly linear ramp or sine wave generated from the precision automated test equipment (ATE) [2], [3]. The signal source is required to be 3 to 4 bits more linear than the device under test (DUT) to accurately test the ADC since any error in the input signal will be treated as part of the ADC's linearity error in the conventional histogram test. To test ADCs beyond 16 bits, the signal generator has to be 19-bit linear or better, which is expensive and difficult to achieve. In addition, the ADC usually takes tens of samples per code to reduce the noise effect in a histogram test. For 16-bit ADC, there are $2^{16} - 1 = 65,535$ transitions to be tested and the number of samples will be close to or even more than millions, which requires seconds or more test time depending on the ADC's sampling rate. Therefore, the cost associated with the test time is very high.

All these requirements pose significant challenges to achieve BIST. First of all, generating a highly linear signal on chip for ADC testing is difficult or even impossible for a high resolution ADC. The cost of building such a signal generator could be much more than the ADC itself. In addition, the test time is not saved if the conventional histogram test is still used. To achieve BIST in an efficient way, these two challenges have to be solved: the test stimulus linearity and long test time. Significant work has been done to overcome these challenges in the past decades. Researchers have achieved some limited BIST features [5]–[18].

Some are addressing the stringent linearity requirement of the input signal. In [14], the delta-sigma modulation technique is applied to generate the highly linear input signal. However, it is not easy to design such a signal generator as the ADC's resolution or speed increases. The design complexity often increases the cost. A low-cost linear ramp generator is proposed in [16]. However, it takes multiple calibrations to achieve better precision and the linearity is still limited.

Manuscript received April 16, 2017; revised August 3, 2017 and October 10, 2017; accepted November 3, 2017. This work was supported by the Semiconductor Research Corporation. This paper was recommended by Associate Editor P. Rombouts. *Corresponding author: Tao Chen.)*

T. Chen, R. L. Geiger, and D. Chen are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: taoc@iastate.edu.)

X. Jin is with NXP Semiconductors, Austin, TX 78735 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSI.2017.2775632

In the contrast, some researchers have focused on algorithms to relax the stimulus linearity requirement. In [15] and [19], the stimulus error identification and removal (SEIR) algorithm is proposed to test precision ADCs using nonlinear stimulus. It has been proven that a 7-bit linear ramp signal can be used to test a high resolution ADC and achieve more than 16 bits accuracy with this method. Rather than using a linear ramp, it uses two nonlinear ramps with a constant voltage shift in between. The nonlinearity in the two ramps will be identified by the algorithm and removed. Therefore, accurate linearity test can be achieved with nonlinear signals. Many practical applications have been presented in [18] and [20]–[22] for SEIR. However, SEIR is based on the histogram method, which means that the data acquisition time is still very long.

In addition to the input signal requirement, the test time reduction is another challenge [23]–[27]. Fast Fourier transform (FFT) test is used to estimate the ADC's INL [24], [25]. A system identification approach is proposed in [28] to evaluate the nonlinearity of a pipeline ADC. Design for test (DfT) methods are introduced in [26] to reduce the test time of calibrating the pipeline ADCs. Model-based testing for ADCs are developed in [29]. Spectral and histogram methods are combined in [30] to reduce the test time. Segmented polynomial fitting method is developed in [31] with low resolution input signal to test the ADC nonlinearity. In [32], a selective code measurement method is introduced to reduce the test time of SAR ADCs. In [17], a ramp generator based on servo-loop method is developed to test the pipeline ADC with reduced test time by the reduced-code linearity test techinuqe. However, these methods reduce the test time by sacrificing other test aspects such as test accuracy or test coverage compared to histogram test. An ultrafast segmented model identification of linearity errors [27] algorithm is proposed to take a system identification approach to capture both linear and nonlinear errors in the ADC. With the segmented non-parametric model, the algorithm can reduce the test time by a factor of over 100 and still achieves a test accuracy superior to the histogram method. However, this method still requires highly linear input signal.

In this paper, the ultrafast stimulus error removal and segmented model identification of linearity errors (USER-SMILE) algorithm is presented. This paper is an expansion based on [15]. The contribution of this paper includes a complete derivation of the algorithm, improvement of algorithm, error analysis and extensive simulations and correlations. The USER-SMILE algorithm uses two nonlinear input signals for the ADC under test. One signal is shifted by a constant voltage with respect to the other nonlinear signal. By subtracting the output codes of the ADC for the two signals, the exact value of the input signal is canceled. The difference between the two output codes represents the nonlinearity difference of these two codes. This nonlinearity of ADC will be represented by a segmented non-parametric INL model. The model parameters will be identified with least square (LS) method. After the identification of the model parameters, the full-code INL/DNL can be constructed. This method is targeted for the built-in self-test which can be used both in production testing to save the test time and the test cost and in the field testing to realize self test in the entire life time of the ADC. The proposed method still has some limitations. Due to the segmented model, this method will work for the ADCs with segmented architectures such as SAR ADCs, Pipeline ADCs and Cyclic ADCs. It will not work for flash ADC or Delta Sigma ADCs.

The remainder of this paper is organized as follows: Section II presents the proposed algorithm with mathematical equations. Section III provides the error analysis. Section IV gives the simulation results and compares them with error analysis. Section V discusses the limitation and the practical implementation of the proposed method. Conclusion is drawn in section VI.

II. User-Smile

In this section, the details of the USER-SMILE algorithm will be presented. The modeling of the ADC nonlinearity errors will be explained first. Then, it will be shown that the input stimulus error will be removed by two functionally related stimuli. At the end, the parameters of the ADC INL model will be identified and used for constructing the INL.

A. Modeling of ADC Linearity Errors

To test the nonlinearity of the ADC, all transition levels need to be identified to obtain the DNL and INL. Industry standard histogram method uses a sine wave or a ramp signal which is sufficiently linear (usually more than 3 bits) than the device under test (DUT). And tens of samples per code are used in order to average the noise. The histogram method shows significant inefficiency. For high resolution ADCs, the number of transitions increases exponentially. And the sampling rate is usually slower for higher resolution ADCs.

However, for high resolution ADCs, the number of components used to build the ADC is usually small. And these small number of components determine the entire ADC's performance. The "segmented non-parametric" model is proved to be an efficient way of modeling the ADC linearity errors [27] with such characteristics. Instead of modeling the circuits inside the ADC, it models the ADC's INL with a segmented non-parametric model. For an N-bit ADC, it uses a small number of parameters instead of the $2^N - 1$ transition levels to represent the INL.

The top plot of Fig.1 shows a typical INL plot of a binary-weighted SAR ADC. The black vertical lines break the INL into multiple same-width segments, defined as MSB (most-significant bits) segment. For example, if the first 4 bits are used to determine the segments, there will be $2^4 = 16$ MSB segments. Each MSB segment has an average INL value which is defined as $E_{\mathrm{M}}(k)$ for the k-th segment. These MSB segmented errors, E_{M}, are from linear errors or nonlinear errors of the ADC or combination of them. With N_{M} bits used for MSB segments, the ADC is treated like a N_{M}-bit flash ADC.

For each MSB segment, there is a smaller INL curve (Fig.1). Within this segment, the MSB code remains the same and only lower bits are converting. For most ADCs except flash ADC and sigma-delta ADC, the lower-bit code are determined with the same sub-ADC or the same lower-bit circuits. Therefore,

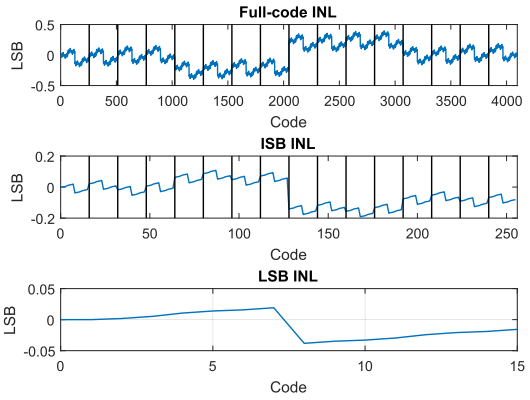


Fig. 1. Segmented Non-parametric INL model (top: full-code INL; middle: ISB-segment INL; bottom: LSB-segment INL).

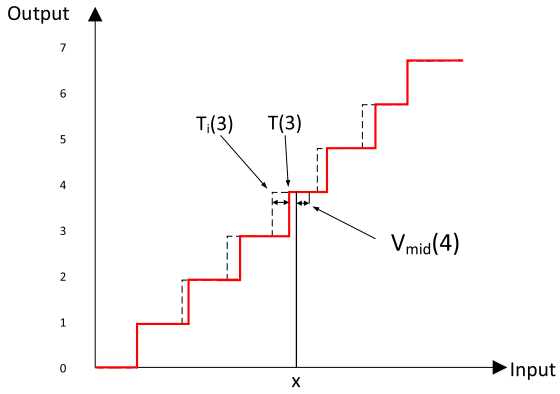


Fig. 2. 3-bit ADC Transfer Function.

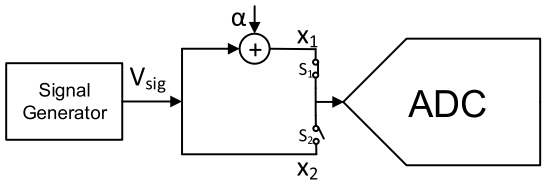


Fig. 3. Algorithm Implementation.

the errors contributed from the lower bits will repeat for each MSB segment. As shown in Fig.1 top, all 16 segments have the same shape. There is an assumption that the errors from lower bits are not affected by the MSB code.

Similarly, the smaller segment, defined as ISB (inter-medium significant bits), can be further broken into smaller LSB (least-significant bits) segments. The ISB segments and LSB segments are shown in the middle and the bottom of Fig.1. The INL for code C can be then defined as

$$INL(C) = E_{\mathrm{M}}(C_{\mathrm{MSB}}) + E_{\mathrm{I}}(C_{\mathrm{ISB}}) + E_{\mathrm{L}}(C_{\mathrm{LSB}}) \qquad (1)$$

where C is the final ADC output code; E_{M}, E_{I} and E_{L} are the MSB, ISB and LSB segmented errors respectively; C_{MSB}, C_{ISB} and C_{LSB} are the MSB, ISB and LSB segment codes. In (1), the segment codes are decimal representations of the corresponding binary codes. In the rest of the paper, the ADC used in simulations and measurements are all binary-weighted ADCs.

This model enables fewer number of parameters to represent the full-code DNL/INL. In most cases, the ADC itself has some nonlinear errors coming from parasitics and secondary effects of transistors or capacitors. If these errors are large enough, it will affect the INL shape. The MSB segmented INL is treating the ADC as a flash. So the MSB INL model can capture these errors.

B. Stimulus Error Removal

In Fig.2, a 3-bit ADC transfer curve is used to explain the notation and the relation among the transition voltages, the input signal and the quantization error. The solid curve is the actual ADC transfer curve and the black dashed line is the ideal transfer curve with same initial and end point. Suppose that there is an input signal x and the ADC output code is $C = 4$. The INL for code 4 is $[T(3) - T_i(3)]/V_{\mathrm{LSB}}$, where T is the actual transition level, T_i is the ideal transition level and V_{LSB} is ideal 1 LSB voltage. For convenience, $T(C)$ stands for transition voltage from code C to code $C+1$. So, $T(0)$ is the transition voltage from code 0 to code 1 and $T(0) = T_i(0)$ in this case. The middle for code C is defined as $V_{\mathrm{mid}}(C)$. And the quantization error is therefore defined as $n_{\mathrm{q}} = V_{\mathrm{mid}}(C) - x$ when the sampled voltage is x and the output code is C. We can then have the relation:

$$\begin{aligned} & x + n_{\mathrm{q}} \\ &= V_{\mathrm{mid}}(C) \\ &= T(C-1) + V_{\mathrm{LSB}} \cdot \frac{1 + DNL(C)}{2} \\ &= T_i(C-1) + V_{\mathrm{LSB}} \cdot \Big\{ INL(C) + \frac{1 + DNL(C)}{2} \Big\} \\ &= T_i(0) + V_{\mathrm{LSB}} \cdot \Big\{ C - 1 + INL(C) + \frac{1 + DNL(C)}{2} \Big\} \\ &= T(0) + V_{\mathrm{LSB}} \cdot \Big\{ C + \frac{INL(C) + INL(C+1) - 1}{2} \Big\} \end{aligned} \qquad (2)$$

where $DNL(C) = INL(C+1) - INL(C)$.

In the precision ADC testing, the input linearity requirement is very high thus making the test cost very high. For built-in self-test purpose, building such highly linear input source is challenging or not practical. Instead, functionally related excitation can be used to relax the linearity requirement.

The implementation of the algorithm is shown in Fig.3. The signal generator generates an output signal V_{sig}. In the first time, the switch s_1 is turned on and switch s_2 is turned off. The signal passes through an adder and the ADC input $x_1 = V_{\mathrm{sig}} + \alpha$. In the second time, the switch s_2 is turned on and the switch s_1 is turned off. The signal generator is directly connected to the ADC input so that $x_2 = V_{\mathrm{sig}}$. In these two samples, x_1 and x_2 are unknown but there is a constant voltage shift α between them.

$$x_1 - x_2 = \alpha. \qquad (3)$$

Considering the additive noise, the two input signals can be expressed as

$$x_1 + n_{a1} + n_{q1} = T(0) + V_{LSB} \cdot \Big\{C_1 + \frac{INL(C_1) + INL(C_1+1) - 1}{2}\Big\}, \tag{4}$$

$$x_2 + n_{a2} + n_{q2} = T(0) + V_{LSB} \cdot \Big\{C_2 + \frac{INL(C_2) + INL(C_2+1) - 1}{2}\Big\} \tag{5}$$

where n_{a1} and n_{a2} are the additive noise for x_1 and x_2. C_1 and C_2 are the corresponding output codes in the two samples.

The amount of the voltage shift is also unknown and will be identified in the proposed method. Subtract (5) from (4), (6) can be obtained:

$$\frac{x_1 - x_2 + n_{a1} - n_{a2} + n_{q1} - n_{q2}}{V_{\mathrm{LSB}}} + C_2 - C_1 = \frac{INL(C_1+1) + INL(C_1) - INL(C_2+1) - INL(C_2)}{2}. \tag{6}$$

In (6), both n_{a1} and n_{a2} are random additive noise. The subtraction of two independent random variables with same variance will also give an random term with doubled variance. n_{q1} and n_{q2} are the quantization error for these two conversions. However, the quantization error is not white. Assuming that the additive noise is large enough (more than 0.3 LSB rms), the entire term $n_{a1}-n_{a2}+n_{q1}-n_{q2}$ can be treated as one random noise because the quantization error can be effectively "whitened" by the additive noise [33]. So, these four terms can be replaced by a single variable n_{all}. Combining (3) and (6) and replacing the noise terms, we can obtain

$$\alpha / V_{\mathrm{LSB}} + n_{\mathrm{all}} + C_2 - C_1 = \frac{INL(C_1+1) + INL(C_1) - INL(C_2+1) - INL(C_2)}{2}. \tag{7}$$

In this equation, we can notice that the input information is no longer needed. The input linearity requirement is completely relaxed. The voltage shift α will be identified; the output codes are already available; and the INL are the unknowns to be solved.

C. Error Identifications and INL Construction

For an N-bit ADC, there are $2^N - 1$ transition levels and there are $2^N - 3$ INL values (with end-point fitting, the first and last transitions are both 0s by definition). With the "segmented non-parametric" INL model, the number of unknowns to represent the full-code INL can be significantly reduced as introduced in previous sections. Replacing the INL in (7) with the INL model (1), (8) can be obtained:

$$\begin{aligned}
&\alpha / V_{\mathrm{LSB}} + n_{\mathrm{all}} \\
&= C_1 - C_2 \\
&\quad + \frac{INL(C_1+1) + INL(C_1) - INL(C_2+1) - INL(C_2)}{2} \\
&= C_1 - C_2 \\
&\quad + \frac{E_{\mathrm{M}}((C_1+1)_{\mathrm{MSB}}) + E_{\mathrm{I}}((C_1+1)_{\mathrm{ISB}}) + E_{\mathrm{L}}((C_1+1)_{\mathrm{LSB}})}{2} \\
&\quad + \frac{E_{\mathrm{M}}(C_{1\mathrm{MSB}}) + E_{\mathrm{I}}(C_{1\mathrm{ISB}}) + E_{\mathrm{L}}(C_{1\mathrm{LSB}})}{2} \\
&\quad - \frac{E_{\mathrm{M}}((C_2+1)_{\mathrm{MSB}}) + E_{\mathrm{I}}((C_2+1)_{\mathrm{ISB}}) + E_{\mathrm{L}}((C_2+1)_{\mathrm{LSB}})}{2} \\
&\quad - \frac{E_{\mathrm{M}}(C_{2\mathrm{MSB}}) + E_{\mathrm{I}}(C_{2\mathrm{ISB}}) + E_{\mathrm{L}}(C_{2\mathrm{LSB}})}{2}
\end{aligned} \tag{8}$$

where $C_{1\mathrm{MSB}}$ stands for the MSB code of code C_1 and $(C_1+1)_{\mathrm{MSB}}$ stands for the MSB code of code C_1+1. ISB and LSB codes are defined in the same way.

For each pair of input signals, one such equation can be obtained. With M pairs of input signals, M equations will be formed and M is much larger than the number of unknowns to be solved. C_1 and C_2 are then two vectors of output codes. For N_{M}-bit MSB, N_{I}-bit ISB and N_{L}-bit LSB segmentation, there are total $2^{N_{\mathrm{M}}} + 2^{N_{\mathrm{I}}} + 2^{N_{\mathrm{L}}}$ unknowns. Take a 12-bit ADC as an example, 4-4-4 (MSB-ISB-LSB bits) segmentation has only 48 unknowns.

The amount of voltage shift α is still unknown. With a large set of output codes, the value of α can be estimated by the average value of $C_1 - C_2$ for quick estimation. A better estimation is to make the voltage shift as an unknown and identify it in the least square method together with the segmented model parameters. The LS solutions for these unknowns can be expressed as

$$\begin{aligned}
&\big\{\hat{E}_{\mathrm{M}}(0), \hat{E}_{\mathrm{M}}(1), \ldots, \hat{E}_{\mathrm{I}}(0), \hat{E}_{\mathrm{I}}(1), \ldots \hat{E}_{\mathrm{L}}(0), \hat{E}_{\mathrm{L}}(1), \ldots, \hat{\alpha}\big\} \\
&= \arg\min \Bigg\{ \sum_{k=1}^{M} \Big[\frac{\alpha}{V_{\mathrm{LSB}}} + C_2(k) - C_1(k) \\
&\qquad + \frac{INL(C_2(k)+1) + INL(C_2(k))}{2} \\
&\qquad - \frac{INL(C_1(k)+1) + INL(C_1(k))}{2} \Big]^2 \Bigg\}
\end{aligned} \tag{9}$$

with the condition that $INL(0) = INL(2^N - 1) = 0$ for an N-bit ADC.

With all these model parameters identified, the full-code INL can be constructed using (1):

$$\hat{INL}(C) = \hat{E}_{\mathrm{M}}(C_{\mathrm{MSB}}) + \hat{E}_{\mathrm{I}}(C_{\mathrm{ISB}}) + \hat{E}_{\mathrm{L}}(C_{\mathrm{LSB}}). \tag{10}$$

Since the full-code INL has been obtained, the full-code DNL can be derived too.

$$\hat{DNL}(C) = \hat{INL}(C+1) - \hat{INL}(C) \tag{11}$$

With this method, the input doesn't need to hit all the codes. With the hit codes and the ADC segmented architecture, those codes that are not hit can be predicted with the INL segmented models. Therefore, even missing codes can be identified.

III. Error Analysis

There are several factors that affect the effectiveness or the performance of the USER-SMILE method. Although the linearity of the input signal is significantly relaxed, there are certain requirements on the input signals. All segments have to be hit. Otherwise, there will be no information on that segment and least square method will fail to solve the equation. Each segment should get sufficient hits to average the noise effect which will be analyzed in the subsection. In addition to the input signal requirement, four most significant error sources are analyzed including the modeling error, the additive noise, the voltage shift nonconstancy and the quantization error. These error sources will affect the estimation accuracy of the algorithm. The USER-SMILE directly models and identifies the INL of the ADC. Therefore, the maximum DNL estimation error can be twice as the maximum INL estimation error.

A. Effects of Unmodeled Error in the INL Model

In the segmented non-parametric model, the INL curve for the lower bits is assumed to be identical in each MSB segment. However, this may not be true. If the ADC has large nonlinear errors, the lower bits will also be affected. This subsection evaluates the estimation error caused by these unmodeled nonlinear errors.

Define the INL of the ADC as two components: the linear component (INL_{lk}) and the nonlinear component (INL_{nlk}). The full-code INL can be expressed as

$$INL_k = INL_{lk} + INL_{nlk}. \tag{12}$$

For the linear component, it is from the capacitor mismatches, which results in a segmented INL shape. For the nonlinear component, it is mainly from the sampling capacitor voltage coefficient, nonlinear parasitics and other voltage dependent effects, which results in a smooth INL shape. The segmented linear errors are modeled by the proposed method. But the nonlinear error is not directly modeled. Define the nonlinear $f_{\rm nl}(x)$ as a function of the input voltage x and the unit is LSB. When the input voltage $x \in [T_{k-1}\ T_k)$ and the output code is k, the amount of nonlinearity at code k is $INL_{nlk} = f_{\rm nl}(T_{k-1})$. For the j-th MSB segment, define the middle point of this segment as $x_{\rm mid}(j)$. The Taylor series of the nonlinear function at this segment can be expressed as

$$\begin{aligned} f_{\rm nl}(x) &= f_{\rm nl}(x_{\rm mid}(j)) \\ &\quad + f'_{\rm nl}(x_{\rm mid}(j))\big[x - x_{\rm mid}(j)\big] \\ &\quad + \sum_{n=2}^{\infty} \frac{f^{(n)}_{\rm nl}(x_{\rm mid}(j))}{n!}\big[x - x_{\rm mid}(j)\big]^n \\ &\approx f_{\rm nl}(x_{\rm mid}(j)) \\ &\quad + f'_{\rm nl}(x_{\rm mid}(j))\big[x - x_{\rm mid}(j)\big] \end{aligned} \tag{13}$$

where the higher order terms are ignored for approximation purpose. In this equation, the first part of the equation is the constant part which will be captured by the segmented INL model. The value of the second part changes as the slope changes, which is not modeled by the proposed method. Within one MSB segment, the difference between the maximum and minimum nonlinear component can be as large as $f'_{\rm nl}(x_{\rm mid}(j)) \cdot V_{\rm MSB}$, where $V_{\rm MSB}$ is voltage range of one MSB segment. Therefore, the absolute error is as large as $f'_{\rm nl}(x_{\rm mid}(j)) \cdot V_{\rm MSB}/2$ with respect to the middle of this segment. With INL end-point fitting, the maximum unmodeled error can be as large as $f'_{\rm nl}(x_{\rm mid}(j)) \cdot V_{\rm MSB}$. Over the entire input range, the absolute value of first derivative of each segment is bounded by the maximum absolute derivative of the $f_{\rm nl}$ over all j:

$$\text{abs}[f'_{\rm nl}(x_{\rm mid}(j))] \le \max\big\{\text{abs}[f'_{\rm nl}(x)]\big\}. \tag{14}$$

Therefore, the maximum absolute unmodeled error is less than $\max\big\{\text{abs}[f'_{\rm nl}(x)]\big\} \cdot V_{\rm MSB}$ with end-point fitting, which can be used to evaluate how much the nonlinear error contributes to the final INL estimation error.

For a 16-bit binary-weighted ADC with 6-5-5 segmentation, there are $2^6 = 64$ segments for MSB codes and each MSB segment corresponds to the voltage range of $1/64$ (input range is normalized to 1). Suppose that the nonlinear function $f_{\rm nl}(x) = 10x^2(x-1)$, and the maximum absolute INL_k caused by the nonlinear component is about 1.5 LSB. The maximum slope of this function is around 10. Therefore, the unmodeled error is less than $\max\big\{\text{abs}[f'_{\rm nl}(x)]\big\} \cdot V_{\rm MSB}/2 = 10 \cdot 1/64/2 = 0.078$ LSB with best fitting and 0.156 LSB with end-point fitting. This unmodeled error is very small and can be further reduced by increasing the number of MSB segments.

B. Additive Noise in the Input Signals

To analyze the additive noise effect, rewrite (8) into a complete matrix form:

$$\alpha / V_{\rm LSB} + n_{\rm all} - C_1 + C_2 = H \begin{bmatrix} E_{\rm M} \\ E_{\rm I} \\ E_{\rm L} \end{bmatrix} \tag{15}$$

where H is a $M \times K$ matrix. M is the total number of samples in each ramp. K is the total number of unknown for $E_{\rm M}$, $E_{\rm I}$ and $E_{\rm L}$. In the H matrix, each row has three locations of $+1/2$ corresponding to the code C_1+1's MSB, ISB and LSB codes, three locations of $+1/2$ corresponding to the code C_1's MSB, ISB and LSB codes, three locations of $-1/2$ corresponding to the code $C_2 + 1$'s MSB, ISB and LSB codes, and three locations of $-1/2$ corresponding to the code C_2's MSB, ISB and LSB codes. Then, (15) is the matrix expression for (8) with M sets of data. With the least square method, the segmented model parameters are estimated as:

$$\begin{bmatrix} \hat{E}_{\rm M} \\ \hat{E}_{\rm I} \\ \hat{E}_{\rm L} \end{bmatrix} = (H^T H)^{-1} H^T (\alpha / V_{\rm LSB} - C_1 + C_2). \tag{16}$$

In this equation, the noise is effectively averaged. In order to evaluate the exact effect of noise on the final estimation, the noise term n_{all} needs to be included. Define the estimation error vector to be e_M, e_I and e_L for MSB, ISB and LSB

segmented INL respectively.

$$\begin{aligned}\begin{bmatrix}\hat{E}_{\mathrm{M}}\\ \hat{E}_{\mathrm{I}}\\ \hat{E}_{\mathrm{L}}\end{bmatrix} &= \begin{bmatrix}E_{\mathrm{M}}\\ E_{\mathrm{I}}\\ E_{\mathrm{L}}\end{bmatrix} - \begin{bmatrix}e_{\mathrm{M}}\\ e_{\mathrm{I}}\\ e_{\mathrm{L}}\end{bmatrix}\\ &= (H^TH)^{-1}H^T(\alpha/V_{\mathrm{LSB}} + n_{\mathrm{all}} - C_1 + C_2)\\ &\quad -(H^TH)^{-1}H^Tn_{\mathrm{all}}. \end{aligned} \tag{17}$$

Therefore, the estimation error caused by the noise is expressed as:

$$\begin{bmatrix}e_{\mathrm{M}}\\ e_{\mathrm{I}}\\ e_{\mathrm{L}}\end{bmatrix} = (H^TH)^{-1}H^Tn_{\mathrm{all}}. \tag{18}$$

To evaluate the relation between the estimation error with the noise variance, some matrix transformations are performed. Multiply both sides by their transpose:

$$\begin{bmatrix}e_{\mathrm{M}}\\ e_{\mathrm{I}}\\ e_{\mathrm{L}}\end{bmatrix}\begin{bmatrix}e_{\mathrm{M}}^T & e_{\mathrm{I}}^T & e_{\mathrm{L}}^T\end{bmatrix} = (H^TH)^{-1}H^Tn_{\mathrm{all}}n_{\mathrm{all}}^TH\left((H^TH)^{-1}\right)^T. \tag{19}$$

In the left side of (19), the diagonal terms are the square of each estimation error.

$$\begin{aligned}&\begin{bmatrix}e_{\mathrm{M}}\\ e_{\mathrm{I}}\\ e_{\mathrm{L}}\end{bmatrix}\begin{bmatrix}e_{\mathrm{M}}^Te_{\mathrm{I}}^Te_{\mathrm{L}}^T\end{bmatrix}\\ &= \begin{bmatrix} e_{\mathrm{M}}(0)^2 & \cdots & e_{\mathrm{M}}(0)\cdot e_{\mathrm{L}}(2^{N_{\mathrm{L}}}-1)\\ \vdots & \ddots & \vdots\\ e_{\mathrm{M}}(0)\cdot e_{\mathrm{L}}(2^{N_{\mathrm{L}}}-1) & \cdots & e_{\mathrm{L}}(2^{N_{\mathrm{L}}}-1)^2\end{bmatrix}.\end{aligned} \tag{20}$$

Similarly, the noise vector multiplied by its transpose is

$$n_{\mathrm{all}}n_{\mathrm{all}}^T = \begin{bmatrix} n_{\mathrm{all}}(1)^2 & \cdots & n_{\mathrm{all}}(1)\cdot n_{\mathrm{all}}(M)\\ \vdots & \ddots & \vdots\\ n_{\mathrm{all}}(1)\cdot n_{\mathrm{all}}(M) & \cdots & n_{\mathrm{all}}(M)^2\end{bmatrix}. \tag{21}$$

There are M sets of noise terms and each noise term has a variance σ_n^2. Assume the noise is random with 0 mean and each noise term is independent of the other noise terms. The expected value of $n_{\mathrm{all}}n_{\mathrm{all}}^T$ matrix is

$$\begin{aligned}E\left[n_{\mathrm{all}}n_{\mathrm{all}}^T\right] &= \begin{bmatrix} n_{\mathrm{all}}(1)^2 & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & n_{\mathrm{all}}(M)^2\end{bmatrix}\\ &= \sigma_n^2 I_M. \end{aligned} \tag{22}$$

where I_M is the $M \times M$ identity matrix. The diagonal terms of $\sigma_n^2 I_M$ are σ_n^2 while other terms are all zeros.

In the right side of (19), H matrix is a constant matrix for a given input signal. The expected values of both sides become:

$$\begin{aligned}E\left\{\begin{bmatrix}e_M\\ e_I\\ e_L\end{bmatrix}\begin{bmatrix}e_M^Te_I^Te_L^T\end{bmatrix}\right\} &= \sigma_n^2(H^TH)^{-1}H^TH\left((H^TH)^{-1}\right)^T\\ &= \sigma_n^2\left((H^TH)^{-1}\right)^T\\ &= \sigma_n^2(H^TH)^{-1}. \end{aligned} \tag{23}$$

Therefore, the expected value of the estimation error's squares are the product of noise variance and the diagonal elements of the $(H^TH)^{-1}$ matrix.

$$E\left\{\begin{bmatrix}e_{\mathrm{M}}^2(0)\\ e_{\mathrm{M}}^2(1)\\ \vdots\end{bmatrix}\right\} = \sigma_n^2\cdot\mathrm{diag}\{(H^TH)^{-1}\}. \tag{24}$$

The H matrix actually depends on the input signal waveform and the voltage shift added to the input, as well as the segmentation in the INL model. To evaluate the sensitivity from the noise to the estimation error, the ADC and the input signal are assumed to be approximately linear for simplicity. Nonlinear ADC or nonlinear input will slightly change the sensitivity but the effect is very small. For a 12-bit ADC with 4-4-4 segmentation (4-bit MSB, 4-bit ISB and 4-bit LSB) and two 1 hit per code ramps, the maximum value for the diagonal elements in the $(H^TH)^{-1}$ matrix is around 0.02. So, the variance of INL estimation is less than $0.06\sigma_n^2$ (worst case is when MSB, ISB and LSB have the same maximum variance). Therefore, the 3 sigma of the estimation error due to noise is less than $0.75\sigma_n$. For 16-bit ADC with 6-5-5 segmentation and two 1 hit per code ramps as inputs, the 3 sigma of the INL estimation error due to noise is less than $0.4\sigma_n$. For comparison, the histogram ramp test with h hits per code has a estimation uncertainty variance being σ_n^2/h. The 3 sigma of 20 hits per code histogram ramp test is around $0.67\sigma_n$. So, the USER-SMILE algorithm produces a similar estimation error due to noise but with 10 times less data.

C. Effects of the Voltage Shift Between Two Signals

The constancy of the voltage shift is critical in the USER-SMILE algorithm. Recent researches have proposed various low-cost highly-constant shift generators [20], [21].

Define the voltage shift as $\alpha = \bar{\alpha} + \alpha_e$, where $\bar{\alpha}$ is the mean value of α and the α_e is the error part. Similar to (15), the estimation error can be obtained with only the error of the voltage shift considered

$$\begin{aligned}&\alpha/V_{\mathrm{LSB}} - C_1 + C_2\\ &\quad= \bar{\alpha}/V_{\mathrm{LSB}} + \alpha_e/V_{\mathrm{LSB}} - C_1 + C_2\\ &\quad= H\begin{bmatrix}E_{\mathrm{M}}\\ E_{\mathrm{I}}\\ E_{\mathrm{L}}\end{bmatrix} = H\begin{bmatrix}\hat{E}_{\mathrm{M}}\\ \hat{E}_{\mathrm{I}}\\ \hat{E}_{\mathrm{L}}\end{bmatrix} + H\begin{bmatrix}e_{\mathrm{M}}^{(\alpha)}\\ e_{\mathrm{I}}^{(\alpha)}\\ e_{\mathrm{L}}^{(\alpha)}\end{bmatrix}, \end{aligned} \tag{25}$$

and

$$\alpha_e/V_{\mathrm{LSB}} = H\begin{bmatrix}e_{\mathrm{M}}^{(\alpha)}\\ e_{\mathrm{I}}^{(\alpha)}\\ e_{\mathrm{L}}^{(\alpha)}\end{bmatrix} \tag{26}$$

where $e_{\mathrm{M}}^{(\alpha)}$, $e_{\mathrm{I}}^{(\alpha)}$ and $e_{\mathrm{L}}^{(\alpha)}$ are the estimation errors for E_{M}, E_{I} and E_{L} respectively due to the nonconstancy part of the voltage shift.

As the input signal changes (such as sine wave or ramp), the LSB and ISB segments will change faster and the MSB segments will change slower. If there is an error in the voltage shift within a small input voltage range, it is likely to hit different ISB and LSB segments so that the error is

evenly distributed in different segments. Therefore, the shift nonconstancy effect on ISB and LSB is small. To analyze the effect of shift nonconstancy on the MSB segments, the ISB and LSB estimation errors are assumed to 0 and (26) can be approximated as:

$$\alpha_e / V_{\mathrm{LSB}} \approx H_{\mathrm{M}} \left[e_{\mathrm{M}}^{(\alpha)} \right] \tag{27}$$

where H_{M} is the first $2^{N_{\mathrm{M}}}$ columns of the H matrix. In most cases, output code C_1 and $C_1 + 1$ are in the same MSB segment. C_2 and $C_2 + 1$ are also in the same MSB segment. So, in the H_M matrix, each row has one "-1" and one "+1" in the $C_{1\mathrm{MSB}}$ and $C_{2\mathrm{MSB}}$ locations respectively. For different types of input signals, the MSB segments change differently. For a ramp input, $C_{1\mathrm{MSB}}$ and $C_{2\mathrm{MSB}}$ increase from 0 to the maximum. For a sine wave input, $C_{1\mathrm{MSB}}$ and $C_{2\mathrm{MSB}}$ go up and down periodically. Regardless of the input signal type, the H_{M} matrix can be sorted in ascending order according to the ADC's output codes. The corresponding α_e vector will be rearranged according to H_{M}'s sorting sequence. An example of H_{M} matrix after sorting is shown below.

$$\begin{bmatrix} -1 & 1 & 0 & 0 & \cdots & 0 \\ -1 & 0 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{28}$$

Each row has one "−1" and one "+1". The column index of the "+1" location is the value of $C_{1\mathrm{MSB}}$. Group all rows with the same $C_{1\mathrm{MSB}}$ and take the average value. For example, the 2nd row and the 3rd row are summed and averaged since the locations of "+1" or the value of $C_{1\mathrm{MSB}}$ are the same. The average value for corresponding shift error with $C_{1\mathrm{MSB}} = k$ is defined as $\bar{\alpha}_e(k)$. Repeat this averaging process for all the MSB codes. The smallest $C_{1\mathrm{MSB}}$ location is the second column since it will be canceled by the "−1" in the first column if $C_{1\mathrm{MSB}}$ is also in the first column. Since all the error differences are relative, $e_{\mathrm{M}}(0)$ can be defined to be 0. Therefore, the first column of the new matrix is removed. If $e_{\mathrm{M}}(0)$ is not 0, all the errors will add a constant value, which doesn't change the INL. Then, a new lower triangle matrix can be formed with all ones in the diagonal.

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -l_{2,1} & 1 & 0 & \cdots & 0 \\ -l_{3,1} & -l_{3,2} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ -l_{k,1} & -l_{k,2} & -l_{k,3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \end{bmatrix} \begin{bmatrix} e_{\mathrm{M}}(1) \\ e_{\mathrm{M}}(2) \\ \vdots \\ e_{\mathrm{M}}(k) \\ \vdots \end{bmatrix} \approx \begin{bmatrix} \bar{\alpha}_e(1) \\ \bar{\alpha}_e(2) \\ \vdots \\ \bar{\alpha}_e(k) \\ \vdots \end{bmatrix} / V_{\mathrm{LSB}}. \tag{29}$$

All the elements above the diagonal are zeros. And for all the elements below the diagonal, they satisfy the following conditions

$$0 \le l_{k,i} \le 1 \tag{30}$$

for all $i < k$ and $1 < k < 2^{N_{\mathrm{M}}}$.

$$0 \le \sum_{i=1}^{k-1} l_{k,i} \le 1 \tag{31}$$

for all $1 < k < 2^{N_{\mathrm{M}}}$.

For a lower triangle matrix, forward substitution can be used to solve for all the unknowns.

$$\begin{aligned} e_{\mathrm{M}}(1) &\approx \bar{\alpha}_e(1), \\ e_{\mathrm{M}}(2) &\approx \bar{\alpha}_e(2) + l_{2,1} e_{\mathrm{M}}(1), \\ &\vdots \\ e_{\mathrm{M}}(k) &\approx \bar{\alpha}_e(k) + \sum_{i=1}^{k-1} \left[l_{k,i} e_{\mathrm{M}}(i) \right]. \end{aligned} \tag{32}$$

The maximum absolute value for α_e is $\max\{|\alpha_e|\}$. Taking the absolute value for both sides, the inequation can be obtained:

$$|e_{\mathrm{M}}(k)| \le \max\{|\alpha_e|\} + \sum_{i=1}^{k-1} \left[l_{k,i} |e_{\mathrm{M}}(i)| \right], \tag{33}$$

with the initial condition

$$|e_{\mathrm{M}}(1)| \le \max\{|\alpha_e|\}. \tag{34}$$

Therefore, the bound for the k-th MSB's estimation error is

$$|e_{\mathrm{M}}(k)| \le k \cdot \max\{|\alpha_e|\}. \tag{35}$$

Define that the constancy of the voltage shift as the ratio of the voltage shift error over the average shift value. Take a 12-bit ADC as an example. With 4-bit as the MSB segment and the voltage shift is around 1 MSB, the maximum error is less than 16 $\times \max\{|\alpha_e|\}$. To achieve 0.2 LSB error from the USER-SMILE algorithm, the voltage shift error should be less than 0.0125 LSB in 12-bit level, which is 50ppm for the shift constancy. For a 16-bit ADC with 6-bit as the MSB segment, the constancy requirement is 3 ppm. The previous work has demonstrated that voltage shift constancy can achieve below 1 ppm [20].

D. Effects of Quantization Error

The USER-SMILE algorithm enables less data than histogram test. In all previous assumptions and derivations, we assume that the total noise is random with 0 mean. However, with 1 hit per code or less than 1 hit per code input signal, the quantization error is not random. With more hits per code like histogram, the average quantization error is less.

If the standard deviation of the additive noise is comparable to 1 LSB or fraction of LSB, the effect of the quantization error will be similar to random noise, which means it is effectively whitened. If the test environment is ultra low-noise and the ADC is also designed to have a very low noise, additional dithering is needed to whiten the quantization error [33], [34].

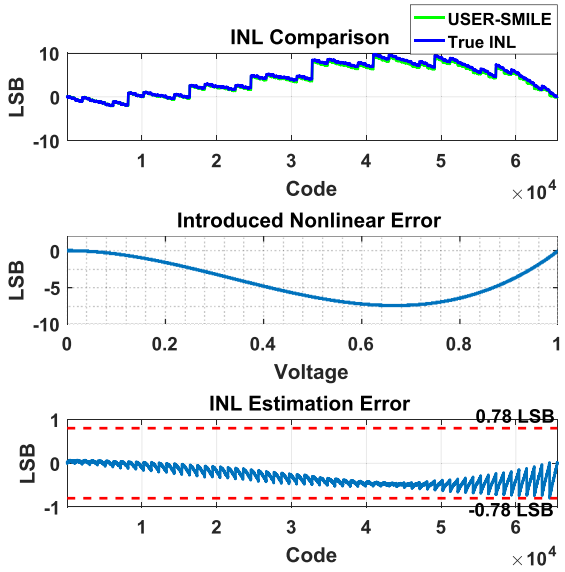


Fig. 4. INL Modeling Error.

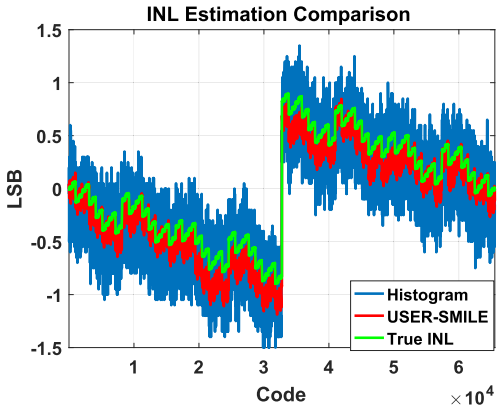


Fig. 5. 16-bit ADC INL Estimation (1 LSB noise).

IV. Simulation Results

In this section, extensive simulations have been done to verify the effectiveness and accuracy of the proposed algorithm. The estimation errors are compared with the analysis in section III. The SAR ADC is particularly modeled due to its wide usage, high resolution and low power features. The behavior model simulation has the advantage of statistical analysis as well as the control over ADC performance. Since everything is mathematical model, the ADC INL/DNL can be theoretically derived. In all the following simulations, 16-bit ADC with 6-5-5 segmentation is modeled unless specified. The input signals are two 1 hit/code nonlinear ramp with around 8-bit linearity performance.

In the SAR ADC, multiple non-idealities are modeled including capacitor random mismatches, voltage dependent coefficient in switch and capacitors, input-referred noise and voltage shift nonconstancy. When analyzing one error source, the other error sources will be removed or minimized. In addition, the capability of identifying missing codes is explained at the end of this section.

A. Modeling Error

In the USER-SMILE algorithm, it is assumed that the ISB or LSB error terms are identical across the entire full

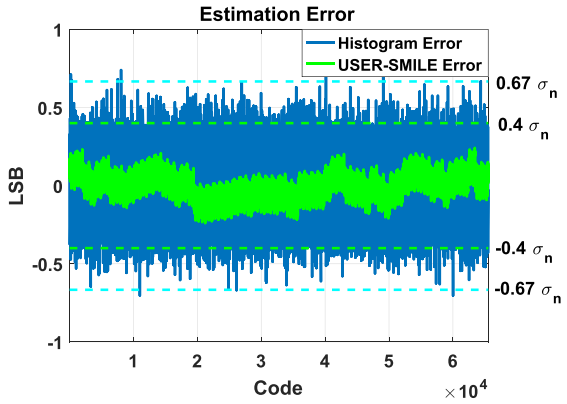


Fig. 6. 16-bit ADC INL Estimation Error(1 LSB noise).

codes for the same ISB or LSB code. However, this may not be the case if the ADC nonlinearity has voltage dependency such as capacitor voltage coefficient or voltage dependent parasitics. A high order polynomial function models the voltage dependency effect. In this test, the noise is set to 0 and the voltage shift is ideal.

In the previous section, the function of $f_{\rm nl}(x) = 10x^2(x-1)$ is used. In order to exaggerate the modeling error effect, a 5 time larger nonlinear function (more than 7.5 LSB) is used to represent the smooth nonlinearity of the ADC. Fig.4 shows the INL comparison, the introduced nonlinear error and the estimation error. Based on the analysis, the maximum unmodeled error from USER-SMILE is 0.78 LSB with end-point fitting. In the simulation result, a boundary of ± 0.78 LSB is shown and the estimation error is within the error boundary.

B. Noise Effect

Random noise has a direct impact on the USER-SMILE estimation accuracy. In the previous analysis, the INL estimation error has a 3-sigma of $0.4\sigma_n$. For comparison, the ADC is also tested with a 20 hits/code linear ramp by histogram test. The 3 sigma estimation error for histogram is 0.67 σ_n. Random noise with 1 LSB sigma (σ_n) is added to the input. The voltage dependency and voltage shift constancy are set to 0. Both USER-SMILE and histogram INL estimations are shown in Fig. 5 together with the theoretical true INL. Both histogram test and USER-SMILE align well with the theoretical INL. But the USER-SMILE result shows better noise averaging.

The INL estimation errors for this test case are plotted in Fig.6. The USER-SMILE INL estimation errors are all within the 3-sigma estimation boundary ($\pm 0.4\sigma_n$). For the histogram test, there are a few codes beyond the 3-sigma boundary ($\pm 0.67\sigma_n$), which is reasonable considering around 65k codes for a 16-bit ADC.

SAR ADCs are randomly generated to further verify the noise effect. For each ADC, the worst INL estimation across all 65,536 codes is selected. Therefore, 1000 worst INL estimation errors (absolute value) can be plotted in Fig.7 for both USER-SMILE and histogram test. In this plot, the 6-sigma lines are drawn ($0.8\sigma_n$ for USER-SMILE, $1.34\sigma_n$

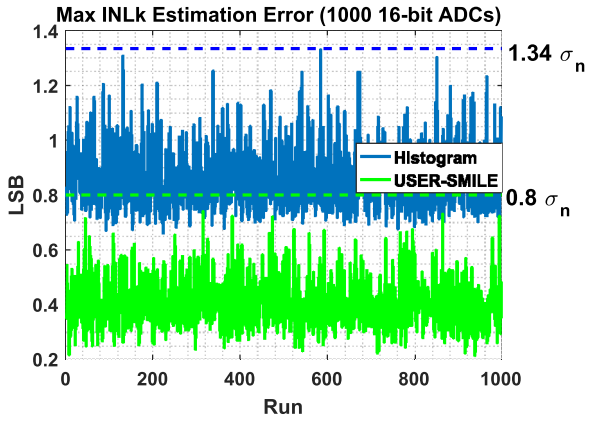


Fig. 7. 1000-time 16-bit ADC INL Estimation Error(1 LSB noise).

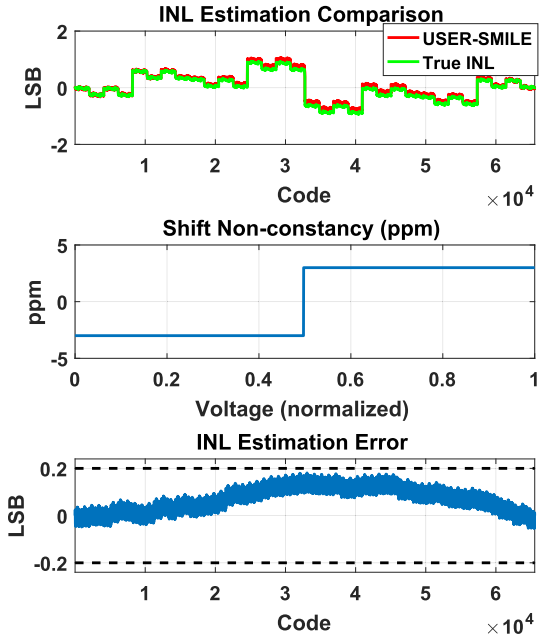


Fig. 8. INL Estimation Error with 3ppm nonconstancy.

for histogram). All the 1000 simulations are below the 6-sigma lines for both USER-SMILE and histogram.

C. Voltage Shift Effect

The constancy of the voltage shift is critical in the USER-SMILE algorithm. As analyzed in the earlier section, to achieve 0.2 LSB estimation error, the shift constancy needs to be less than 3ppm for a 16-bit ADC. In this example, we generate a 3ppm step function as the nonconstancy of the voltage shift while other error sources are all set to be 0. This voltage shift error results in a "bell"-shaped estimation error as shown in Fig.8. From the simulation, the estimation errors are all within the ±0.2 LSB error limits.

D. Missing Codes Identifications

The USER-SMILE algorithm can estimate the full-code INL with even less than 1 hit/code input signal, which means some codes may not be hit in the test. Thanks to the segmented

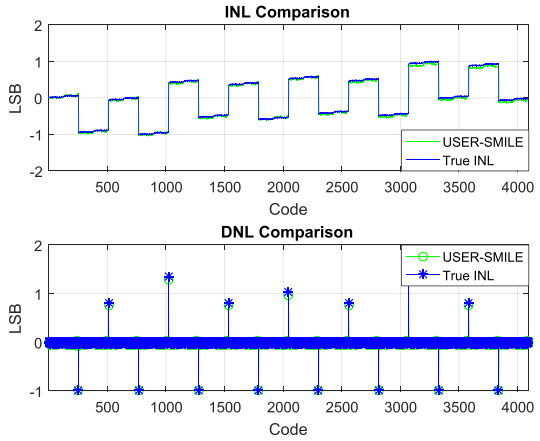


Fig. 9. Missing codes at the boundary.

INL model, even some codes are not hit, their linearity can still be accurately predicated from other codes. Missing codes in the ADC is usually a very important specification. Some special cases are generated to illustrate the missing codes identification in the USER-SMILE. In this example, 12-bit ADCs with 4-4-4 segmentation are used for simplicity. For SAR ADCs, define the most significant capacitor as C_1 and the least significant capacitor as C_{12}.

In the first case, suppose the 4-th MSB capacitor (C_4) is smaller than expected, causing missing codes at major transitions (Fig.9). Therefore, code 255 will be missing. As a result, all the transition voltages for the second MSB segment (from code 256 to code 511) will be lower, causing the segmented error of this MSB segment ($e_M(1)$) to be lower. From the Fig.9, the INL from code 256 to 511 is around 1 LSB lower than the INL from code 0 to 255. The missing codes occur every time when C_4 is selected (every 512 codes). Therefore, the next missing code is 767 (255+512). For the USER-SMILE algorithm, although code 255 is never hit, the INL(255) can be estimated by:

$$\hat{INL}(255) = e_M(0) + e_I(15) + e_L(15) \tag{36}$$

where $e_M(0)$, $e_I(15)$ and $e_L(15)$ can be estimated from other codes. When constructing the INL, if the beginning of the MSB segment is more than 1 LSB lower than the end of the previous MSB segment, a missing code is identified. The DNL for that code will be set to -1 and the INL will be updated accordingly.

In the second case, the missing codes occur in the middle of a MSB segment rather than the MSB major transitions. In Fig.10, the 5-th capacitor (C_5) is smaller than normal and causes a missing code at code 127. It will be identified in the ISB segmented errors. This missing code will repeat in every MSB segment regardless of the MSB errors. The next appearance will be 383 (127+256).

V. Discussion

The segmented INL model can accurately represent the INL for ADCs with segmented architecture. However, for low-resolution high-speed flash ADCs, the INL is not segmented.

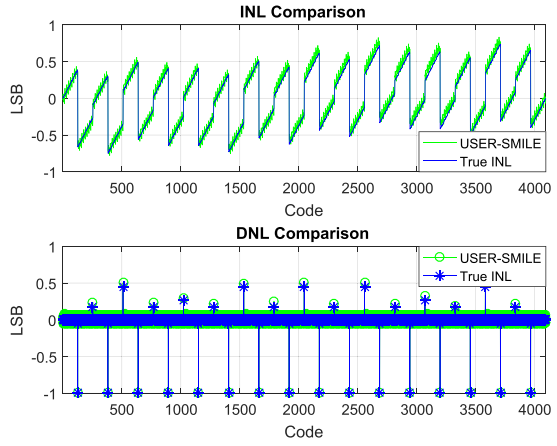


Fig. 10. Missing codes at the middle.

Therefore, the segmented INL model cannot be used to estimate the linearity for flash ADCs. The INL for delta sigma ADCs is also not segmented. But high resolution delta sigma ADCs are never tested for full-code INL/DNL [27].

In this paper, the segmented model is targeted for binary-weighted ADCs. For other segmented ADCs such as Pipeline ADCs, Cyclic ADCs and subradix-2 SAR ADCs, some modifications are needed to the segmented model. For these ADCs, the segmentation of the INL is not determined by the final output codes due to calibration or redundancy. Instead, the raw code directly from the comparators or shift registers should be used to determine the segmentation. In (8), C_1 and C_2 are the final codes but the MSB/ISB/LSB codes should be obtained from the raw code.

The proposed method can be implemented as on-chip BIST in a low-cost way. The linearity requirement for the signal generator has been significantly relaxed. Therefore, the design of such signal generators is simplified. DACs such as R2R DACs or simple ramp generators can be chosen as the signal generators [22]. For embedded ADCs, there is usually a DAC in the same SoC, which can be reused as the signal generator. The key of the implementation is the voltage shift generation. It has been shown in many literature that generation of highly constant voltage shift can be achieved on-chip[18], [20]–[22].

VI. Conclusion

A fast and cost-effective method for ADC linearity test is presented in this paper. The USER-SMILE algorithm allows the stimulus signal's linearity requirement to be significantly relaxed and the test time to be reduced by orders of magnitude compared to the state-of-art histogram method, thus greatly reducing the test cost. The simulation results demonstrate that the USER-SMILE can achieve superior test coverage and accuracy. With the USER-SMILE algorithm, a new BIST solution can be practical, which doesn't require highly accurate and expensive ATE as the signal generator. Furthermore, it saves the test time and simplifies the test board and interface design.

To actually implement this method as on-chip hardware, the signal generator design is simplified. The voltage shift generator can be implemented in a cost-effective way. The algorithm part can be processed in the ATE for production testing. Or it can be implemented as software in the CPU of a system-on-chip (SoC) or designed as a hardware block which doesn't consume CPU resource. The extra cost introduced is well compensated by the test cost reduction.

With on-chip signal generator and on-chip computation in SoC, it not only saves the test cost, but also enables the field testing. For example, the BIST can be performed every time the chip is powered on. For critical applications such as automotive, this self-test feature is very important to guarantee the functionality and performance of the electronic system over the product's life time. In addition, the self-test results can be further used for calibration purposes which can repair and improve the ADC performance.

References

[1] L. Wilson, "International technology roadmap for semiconductors (ITRS)," Semicond. Ind. Assoc., Washington, DC, USA, Tech. Rep. 2013 edition, 2013.

[2] *IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters*, IEEE Standard 1241-2010 (Revision of IEEE Standard 1241-2000), Jan. 2011, pp. 1–139.

[3] M. Burns G. W. Roberts, and F. Taenzler, *An Introduction to Mixed-Signal IC Test and Measurement*. New York, NY, USA: Oxford Univ. Press, 2011.

[4] T. Kuyel, "Linearity testing issues of analog to digital converters," in *Proc. Int. Test Conf.*, Sep. 1999, pp. 747–756.

[5] T. Chen and D. Chen, "Ultrafast stimulus error removal algorithm for ADC linearity test," in *Proc. IEEE 33rd VLSI Test Symp. (VTS)*, Apr. 2015, pp. 1–5.

[6] J. Duan, L. Jin, and D. Chen, "INL based dynamic performance estimation for ADC BIST," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2010, pp. 3028–3031.

[7] J. Duan, D. Chen, and R. Geiger, "A low cost method for testing offset and gain error for ADC BIST," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2012, pp. 2023–2026.

[8] F. Azais, S. Bernard, Y. Bertrand, X. Michel, and M. Renovell, "A low-cost adaptive ramp generator for analog BIST applications," in *Proc. 19th IEEE VLSI Test Symp. (VTS)*, Apr./May 2001, pp. 266–271.

[9] W. Jiang and V. D. Agrawal, "Built-in self-calibration of on-chip DAC and ADC," in *Proc. IEEE Int. Test Conf.*, Oct. 2008, pp. 1–10.

[10] Z. Czaja, "Using a square-wave signal for fault diagnosis of analog parts of mixed-signal electronic embedded systems," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 8, pp. 1589–1595, Aug. 2008.

[11] A. J. G. Arteaga, E. J. Peralias, and A. Rueda, "Blind adaptive estimation of integral nonlinear errors in ADCs using arbitrary input stimulus," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 2, pp. 452–461, Feb. 2011.

[12] S.-F. Hung and H.-C. Hong, "A fully integrated BIST $\Delta\Sigma$ ADC using the in-phase and quadrature waves fitting procedure," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 12, pp. 2750–2760, Dec. 2014.

[13] M. F. Toner and G. W. Roberts, "A BIST scheme for a SNR, gain tracking, and frequency response test of a sigma-delta ADC," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 42, no. 1, pp. 1–15, Jan. 1995.

[14] J.-L. Huang, C.-K. Ong, and K.-T. Cheng, "A BIST scheme for on-chip ADC and DAC testing," in *Proc. Design, Autom. Test Eur. Conf. Exhibit.*, 2000, pp. 216–220.

[15] L. Jin, K. Parthasarathy, T. Kuyel, D. Chen, and R. L. Geiger, "Accurate testing of analog-to-digital converters using low linearity signals with stimulus error identification and removal," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 3, pp. 1188–1199, Jun. 2005.

[16] 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, no. 2, pp. 263–273, Feb. 2003.

[17] G. Renaud *et al.*, "A 65 nm CMOS ramp generator design and its application towards a BIST implementation of the reduced-code static linearity test technique for pipeline ADCs," *J. Electron. Test.*, vol. 32, no. 4, pp. 407–421, 2016.

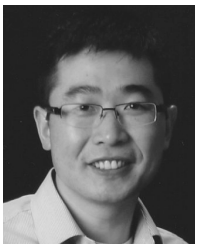
[18] A. J. Gines *et al.*, "Linearity test of high-speed high-performance adcs using a self-testable on-chip generator," in *Proc. 21th IEEE Eur. Test Symp. (ETS)*, May 2016, pp. 1–6.

[19] L. Jin, D. Chen, and R. L. Geiger, "SEIR linearity testing of precision A/D converters in nonstationary environments with center-symmetric interleaving," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 5, pp. 1776–1785, Oct. 2007.
[20] Y. Duan, T. Chen, Z. Liu, X. Zhang, and D. Chen, "High-constancy offset generator robust to cdac nonlinearity for SEIR-based ADC BIST," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 3016–3019.
[21] X. Jin and N. Sun, "Low-cost high-quality constant offset injection for SEIR-based ADC built-in-self-test," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2014, pp. 285–288.
[22] J. Duan, B. Vasan, C. Zhao, D. Chen, and R. Geiger, "Stimulus generator for SEIR method based ADC BIST," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, Jul. 2009, pp. 251–255.
[23] A. Laraba, H.-G. Stratigopoulos, S. Mir, and H. Naudet, "Exploiting pipeline ADC properties for a reduced-code linearity test technique," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 10, pp. 2391–2400, Oct. 2015.
[24] F. Adamo, F. Attivissimo, N. Giaquinto, and M. Savino, "FFT test of A/D converters to determine the integral nonlinearity," *IEEE Trans. Instrum. Meas.*, vol. 51, no. 5, pp. 1050–1054, Oct. 2002.
[25] F. Attivissimo, N. Giaquinto, and I. Kale, "INL reconstruction of A/D converters via parametric spectral estimation," *IEEE Trans. Instrum. Meas.*, vol. 53, no. 4, pp. 940–946, Aug. 2004.
[26] H.-M. Chang, C.-H. Chen, K.-Y. Lin, and K.-T. Cheng, "Calibration and testing time reduction techniques for a digitally-calibrated pipelined ADC," in *Proc. 27th IEEE VLSI Test Symp.*, May 2009, pp. 291–296.
[27] Z. Yu and D. Chen, "Algorithm for dramatically improved efficiency in ADC linearity test," in *Proc. IEEE Int. Test Conf.*, Nov. 2012, pp. 1–10.
[28] Z. Yu, D. Chen, R. Geiger, and Y. Papantonopoulos, "Pipeline ADC linearity testing with dramatically reduced data capture time," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 1. May 2005, pp. 792–795.
[29] C. Wegener and M. P. Kennedy, "Linear model-based testing of ADC nonlinearities," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 1, pp. 213–217, Jan. 2004.
[30] A. C. Serra, M. F. da Silva, P. M. Ramos, R. C. Martins, L. Michaeli, and J. Saliga, "Combined spectral and histogram analysis for fast ADC testing," *IEEE Trans. Instrum. Meas.*, vol. 54, no. 4, pp. 1617–1623, Aug. 2005.
[31] S. Kook, H. W. Choi, and A. Chatterjee, "Low-resolution DAC-driven linearity testing of higher resolution ADCs using polynomial fitting measurements," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 3, pp. 454–464, Mar. 2013.
[32] S. Goyal *et al.*, "Test time reduction of successive approximation register A/D converter by selective code measurement," in *Proc. IEEE Int. Conf. Test*, Nov. 2005, pp. 1–8.
[33] L. Schuchman, "Dither signals and their effect on quantization noise," *IEEE Trans. Commun. Technol.*, vol. COM-12, no. 4, pp. 162–165, Dec. 1964.
[34] Y. Duan, T. Chen, and D. Chen, "Low-cost dithering generator for accurate ADC linearity test," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2016, pp. 1474–1477.

Tao Chen (S'12) received the B.S. degree in electrical engineering from Iowa State University, Ames, USA, in 2013, where he is currently pursuing the Ph.D. degree in electrical engineering.

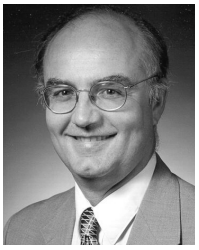
In 2014, he joined Freescale Semiconductor as an Intern focusing on ADC built-in self-test. In 2015, he joined Freescale Semicondcutor as an Analog Design Intern focusing on Cyclic ADC design. In 2016, he joined NXP Semiconductors as a Design Intern focusing on precision amplifier design.

His current research interests include data converter design, modeling, built-in self-test, and calibration.



Xiankun Jin (M'14) received the M.S. degree in electrical engineering from the University of Texas at Austin in 2013. He is a Senior Principle Engineer with the Automotive Microcontroller and Processors Group, Mixed-Signal IP Team, NXP Semiconductors, Austin, TX, USA.

He led the implementation of on-chip parametric ADC BIST solution supported by NXP/Freescale Discovery Labs starting 2014. He has been an Analog and Mixed-Signal Test Engineer, a DFT Team Leader, and a Design Engineer with the Microcontroller Group, Freescale/Motorola, since 2002. His main topic of interest includes on-chip analog BIST techniques to reduce test cost, BIST-based calibration to improve performance of analog modules, and in-field test to enhance functional safety.



Randall L. Geiger (S'75–M'77–SM'82–F'90) received the B.S. degree in electrical engineering and the M.S. degree in mathematics from the University of Nebraska, Lincoln, NE, USA, in 1972 and 1973, respectively, and the Ph.D. degree in electrical engineering from Colorado State University, Fort Collins, CO, USA, in 1977.

He was a Faculty Member with the Electrical Engineering Department, Texas A&M University, College Station, TX, USA, from 1977 to 1990. Since 1991, he has been a Faculty Member with the Electrical and Computer Engineering Department, Iowa State University, Ames, IA, USA, where he is currently the Doluca Professor. His teaching and research interests are in the fields of analog and mixed-signal VLSI design, specifically in the areas of amplifier design, test and built-in self test of mixed-signal circuits, data converter design, device modeling, and design for yield.

Dr. Geiger was a recipient of the Meritorious Service Award of the IEEE Circuits and Systems Society in 1996, the Golden Jubilee Medal of the IEEE Circuits and Systems Society in 2000, and the IEEE Millennium Medal in 2000. He was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART II and the Circuits and Systems Society Editor of the *IEEE Circuits and Devices Magazine*. He was a member of the IEEE Publications Board and the IEEE Periodicals Council. He has served in various capacities on the Technical Program Committees and the Organizing Committees for the IEEE International Symposium on Circuits and Systems and the IEEE Midwest Symposium on Circuits and Systems. He is a past member of the Board of Governors, a past Vice President of Publications, and a past President of the IEEE Circuits and Systems Society. He is a past Chair of the Transactions Committee of the Periodicals Council.

Degang Chen (S'90–M'92–SM'02–F'16) received the B.S. degree in instrumentation and automation from Tsinghua University, Beijing, China, in 1984, and the Ph.D. degree in electrical and computer engineering from the University of California at Santa Barbara, Santa Barbara, CA, USA, in 1992.

He was the John R. Pierce Instructor of electrical engineering with the California Institute of Technology, Pasadena, CA, USA, in 1992. He was a Faculty Fellow with Boeing, Chicago, IL, USA, in 1999; Maxim Integrated, San Jose, CA, USA, in 2001; and Texas Instruments (TI), Dallas, TX, USA, in 2011, 2012, and 2014. He has delivered technical seminars at Carnegie Mellon University, Pittsburgh, PA, USA; Columbia University, New York City, NY, USA; South Methodist University, Dallas, University of Minnesota, Minneapolis, MN, USA, The University of Texas at Dallas, Dallas; Broadcom, CA, USA; Cypress Semiconductor, WA/MN, USA; Global Foundries, VT, USA; IBM Watson, NY, USA; Infineon, Germany; Intel, CA/OR, USA; NXP, TX, USA; TI Dallas/India/Santa Clara; and Xilinx, CA, USA. Since 1992, he has been with Iowa State University, Ames, IA, USA, where he is currently a Professor and the Jerry Junkins Chair of electrical and computer engineering. He has authored over 230 refereed journal and conference publications. His current research interests include analog and mixed-signal integrated circuit design and testing, integrated circuit sensor design, and high-accuracy test without requiring high-accuracy instrumentation.

Dr. Chen received 13 best paper awards and other honors, including the IEEE Ned Kornfield Best Paper Award in 2013 and 2014.