An ADC-BiST Scheme Using Sequential Code Analysis

Erdem S. ERDOGAN and Sule OZEV
Duke University Department of Electrical & Computer Engineering Durham, NC USA
{ese,sule}@ee.duke.edu

Abstract

This paper presents a built-in self-test (BiST) scheme for analog to digital converters (ADC) based on a linear ramp generator and efficient output analysis. The proposed analysis method is an alternative to histogram based analysis techniques to provide test time improvements, especially when the resources are scarce. In addition to the measurement of DNL and INL, non-monotonic behavior can also be detected with the proposed technique. We present two implementation options based on how much on-chip resources are available. The ramp generator has a high linearity over a full-scale range of $1V$ and the generated ramp signal is capable of testing $13-$ bit ADCs. The circuit implementation of the ramp generator utilizes a feedback configuration to improve the linearity having an area of $0.017mm^2$ in $0.5\mu m$ process.

1 Introduction

Most devices manufactured today contain some analog functionality, leading to the prevalence of mixed signal chips with a large digital and small analog content. An important challenge in the production of mixed signal chips is the need to use more expensive mixed-signal testers even when the analog content in the design is small.

In order to obviate the reliance on mixed-signal testers, researchers have proposed to use BiST techniques that are specifically tailored for the common components in mixed-signal designs such as ADCs and PLLs. Since data converters are essential (and sometimes the only analog) components on the mixed-signal designs, there has been a lot of research activity in BiST techniques for data converters [7, 14, 3, 4, 8] some of which will be discussed in greater detail in the next section. Histogram based approaches have been popular in testing important parameters of ADCs, such as DNL and INL [14, 3, 4, 6, 10]. In histogram based ADC testing, code frequency statistics are collected based on an input signal (ramp, sinusoidal) and analyzed to derive INL, DNL, offset voltage and gain error in terms of the ADC's effective least significant bit (LSB). To collect the data, most histogram techniques require access to an on-chip memory and a DSP. In the absence of such an access either due to the lack of on-chip memory or due to layout constraints, the authors in [14, 3, 4] suggest collecting the histogram of each code in a sequential manner such that the test time is appreciably increased. Another problem with histogram analysis is that it can not detect non-monotonic behavior.

In this paper, we propose an alternative analysis technique for ADC BiST that does not increase the test time appreciably in the absence of a large memory, while also detecting non-monotonic behavior. Our technique uses a counter along with a bit-flip detector to record the code widths. Since code widths are recorded every time a code switch occurs, a full pass of the linear input ramp can be used to calculate DNL, INL, offset voltage, gain and check for non-monotonicity. We present two implementation options depending on the availability of on-chip resources. With our sequential analysis scheme, we pipeline the data collection for each code with the data storage (or shifting) of an earlier code. Thus, our scheme can be used in conjunction with a digital tester or an on-chip memory and DSP.

We also propose a linear ramp generator to be used as an on-chip test stimuli generator for the ADC testing. This ramp generator is implemented in CMOS 0.5μ technology and has a voltage control capability to adjust the slope of the ramp. Based on the post-layout simulations performed in HSpice, the ramp generator has 15-bit linearity on $1V$ full-scale range.

The organization of the remainder of this paper is as follows. In the next section we discuss some examples of ADC BiST techniques including ramp generator circuits. In section 3 we explain our method for ADC test together with the performance analysis and the comparison with existing histogram methods. A high linearity ramp generator circuit is given in Section 4 with post-layout simulation results.

2 Prior Work

Various BiST techniques have been proposed for ADC testing [1, 7, 14, 3, 4, 6, 10]. In [1], the authors reconfigure the device in the test mode to create oscillations such that some measurements are possible without generating a test stimulus. In an another BiST technique [7], the authors propose an on-chip test input generation and a digital output monitoring method for the ADC, based on analysis of the Least Significant Bit (LSB). Histogram-based ADC testing

978-3-9810801-2-4/DATE07 © 2007 EDAA

with sinusoidal input signals has been proposed [11, 12]. Ramp inputs and histogram analysis has been a more popular ADC BiST approach due to its uniform code distribution and reduced storage requirement [14, 3, 4]. However, the BiST scheme still needs a reasonable digital storage to record the frequency of the output codes. The time decomposition technique [14, 3, 4] has been proposed as solution to the need of large memory. This technique reduces the required storage capacity but increases the test time exponentially with the ADC resolution.

On the test input generation front, a common method of generating on-chip ramp signals is to charge a capacitor by a voltage controlled current source. In [2], the authors propose a cascode current mirror based current source architecture. The reported linearity of the ramp is $15 - bit$ for $3V$ full-scale range, which is translated to $1V$ full scale range as $13 - bit$. A modification to this current source circuit, a triangular wave generator is proposed [5]. The authors utilize the same principles for discharging the capacitor and controlling the slope of the voltage ramp. The reported linearity of the triangular wave is about $14 - bit$ for $\pm 1V$ full-scale range.

3 Alternative ADC BiST Scheme

The most critical parameters of ADCs are DNL, INL, offset voltage and gain. Thus, BiST work on ADCs has concentrated on these parameters [14, 3, 4, 6, 10]. Histogram analysis is a powerful tool to measure code widths when input signals do not follow each code sequentially (e.g. sinusoidal inputs). The ramp input is a special case since it passes through each code sequentially. This property of the ramp input can be exploited to reduce the reliance on the on-chip resources.

3.1 The BiST Scheme and Implementation

As an alternative to histogram analysis, we propose a different output analysis scheme. Rather than relying on measuring the *code frequency* as in the histogram technique, our technique uses a counter along with a code change detector to directly measure the *code width*. By sequentially passing through each code and recording the code width, our technique can also detect non-monotonic behavior.

In order to measure the widths of the output codes, a slow ramp signal is needed, the implementation of which we will discuss in the next section.

With a given ramp signal input, the code widths at the output are automatically translated to time durations such that they can be measured by a simple on-chip counter. After the transition from one code to another, the counter value is recorded and then the counter is reset to measure the width of the next code. Since there is at least one bit flip from one code transition to another, a simple bit-flip detector composed of XOR gates and an OR-tree can be used to detect the code transitions.

The sequential analysis also enables the detection of non-monotonic behavior. To test for non-monotonicity, we employ a second counter that is incremented every time there is a code change. A mismatch between the ADC output and the counter value indicates non-monotonic behavior (missing codes).

We will discuss two options to implement the proposed technique based on the availability of on-chip resources. First, we will assume that the chip contains an accessible on-chip memory (a buffer) and an on-chip digital signal processor (DSP) as suggested in of most prior work [14, 3, 4] (option 1). In this case, the width of each code can be recorded in the buffer and analyzed by the DSP as shown in Figure 1. In order to increase the accuracy of measurements and reduce the effect of noise, multiple passes through the codes are preferred. Therefore, the code width values will be updated after each pass by the addition operation of the DSP. Finally, the DSP analyzes the results and makes a pass/fail decision. The additional resources needed for the BiST scheme are two counters, and the code transition detector.

The advantage of this scheme over the histogram analysis is that the memory does not need to be accessed every clock cycle. Memory accesses only occur after each code transition, thus a slower access does not inhibit the application of our technique.

In the second implementation option (option 2), where there is no available on-chip memory or the memory is not accessible due to layout limitations, the values of the counters and the ADC output code are written to the registers as shown in Figure 2. During the measurement of the current code width, the previous code width measurement is scanned out to the digital tester from the registers. Multiple passes are also possible such that the results are updated by the digital tester after each pass.

Application of histogram techniques when there is no

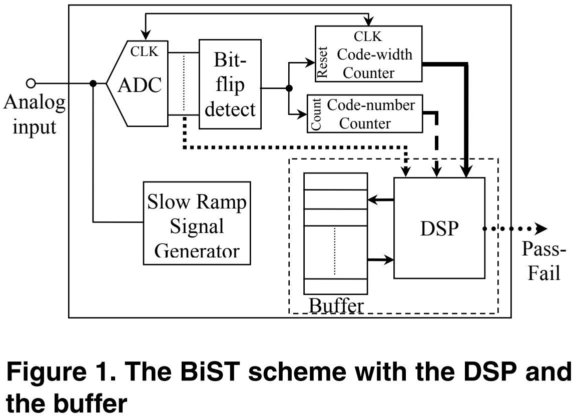

Figure 1. The BiST scheme with the DSP and the buffer

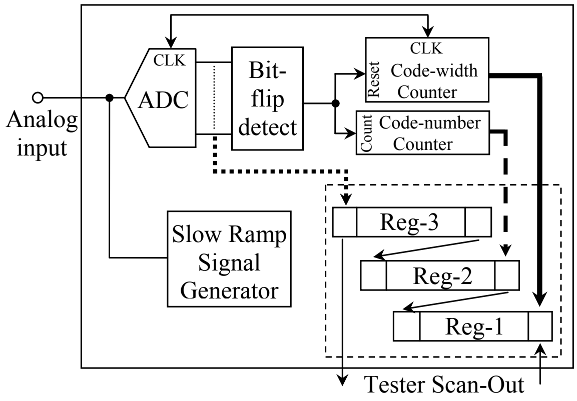

Figure 2. The BiST scheme with the tester

available on-chip memory requires collecting each code's data in a sequential manner [3, 4], increasing the test time by 2^n where n is the ADC resolution. With our technique, the test application does not change.

3.2 Implementation Details

The proposed BiST scheme uses 2 counters, a bit-flip detector, and up to 3 registers (depending on whether option 1 or option 2 is used).

The Bit-flip Detector

The bit-flip detector is composed of n, $2-bit$ shift registers to compare the current digital code with the previous one. With each output from the ADC, shift registers will shift and the difference between the two output codes will be detected by an XOR-OR network.

The Counters

The desired accuracy of the test scheme determines the size of the code-width counter. For example, an ADC test with 16 hits-per-code (HPC: total number of conversions per ADC code) and $0.5LSB$ maximum DNL requires a final counter value of 25 $(16 \times 1.5 + 1)$ assuming that the maximum acceptable code width is 24. Therefore, a $5-bit$ counter is sufficient for this test setup for a $1/16LSB$ accuracy. The code-number counter should be the same size as the ADC since there are 2^n digital output codes to be passed.

Additional Components Needed for Option 1 and Option 2

According to the two previously mentioned implementation options, the BiST scheme will utilize different on-chip resources. In option 1, where a buffer and a DSP are available, the analysis is performed by the DSP using the recorded values in the buffer. The size of the storage needed in the buffer is determined by the desired accuracy and by the number of repeated measurements of the BiST scheme. In the second case, three required scan-out registers should be able to keep the numbers generated by the counters and by the ADC. Therefore, their sizes depend on the ADC resolution and the size of the code width counter.

3.3 Accuracy, Test Time and Area Overhead Analysis

Clearly, the test time and the implementation area mainly determined by the resolution of the ADC as well as the desired accuracy of the test scheme. The accuracy of the test scheme is impacted by three factors: HPC, noise, and the linearity of the ramp. The error due to the quantization of the code width can be calculated in terms of LSB by:

$$\varepsilon_{HPC} = \frac{1}{HPC_{ramp} \times N} = \frac{1}{HPC} \qquad (1)$$

where N is the number of passes through the ramp.

The measurement error due to the nonlinearity of the ramp signal can be determined in a similar manner:

$$\varepsilon_{NL} = \frac{1}{2^{(N_{ramp} - N_{ADC})}} \qquad (2)$$

where N_{ramp} represents the linearity of the ramp signal.

The accuracy also depends on the thermal noise in the system, where the random noise spikes may result in incorrect data conversions. The impact of thermal noise on the accuracy can be evaluated assuming Gaussian distribution for the amplitude of the noise spikes having an average of 0 and standard deviation of σ_n. If we call the fractional part of the analog input signal as v_f, which has a uniform distribution, $p_v(v_f)$ from 0 to $1LSB$, the amplitude of the noise spike should be either smaller than $-v_f$ or greater than $LSB - v_f$ to change the digital output code. The probability of incorrect output code due to the noise spike can be calculated as:

$$P_n = 2 \times \int_0^{0.5LSB} p_v(v_f) \times [\int_{\frac{0.5LSB - v_f}{\sigma_n}}^{\infty} p_n(u)du]dv_f \qquad (3)$$

where $p_n(u)$ is obtained by a change of variables:

$$p_n(u) = \frac{1}{\sqrt{2\pi}} \exp{(-u^2/2)} \qquad (4)$$

If the distribution of v_f is discretized and the integration inside the square brackets is referred as the Q function, the probability can be recalculated as

$$P_n = 2 \times \frac{1}{K} \sum_{i=1}^{K} Q(\frac{0.5LSB - v_f(i)}{\sigma_n}) \qquad (5)$$

where $v_f(i) = i \cdot 0.5LSB/K$

Using the probability obtained for incorrect conversion, the error in the DNL measurement due to the thermal noise effects can be calculated as

$$\varepsilon_n = \sum_{i=1}^{HPC} C(HPC,i)P_n^{(i)}(1-P_n)^{(HPC-i)}\frac{i}{HPC} \quad (6)$$

Since all error components are uncorrelated, the overall error will be:

$$\varepsilon_{overall} = \varepsilon_{HPC} + \varepsilon_{NL} + \varepsilon_n \quad (7)$$

It is also clear from Equation 7 that increasing HPC beyond a certain point will not necessarily increase the accuracy. As an example, for an $LSB = 61\mu V$, a $\sigma_n = 0.01LSB$ (corresponding to $-191dBm/Hz$ noise PSD for a $100MHz$ bandwidth), and $(N_{ramp} - N_{ADC}) = 3$ the error due to ramp linearity, (ε_{NL}) will become the dominating factor (i.e. $\varepsilon_{NL} > 10\varepsilon_n$, $\varepsilon_{NL} > 10\varepsilon_{HPC}$), when $HPC = 80$. Therefore, increasing HPC beyond 80 will not provide any benefit for the measurement error.

The test time is determined by the ADC properties, such as the resolution and the sampling frequency $F_{sampling}$ as well as HPC. The total test time can be calculated by

$$TestTime_{Total} = \frac{HPC \times 2^n}{F_{sampling}} sec \quad (8)$$

The required slope of the ramp signal for a given full-scale-range FS of the ADC can also be calculated with the test parameters as:

$$RampSlope = \frac{F_{sampling} \times FS}{HPC_{ramp} \times 2^n} Volts/sec \quad (9)$$

The area overhead introduced by the digital part of the BiST scheme will be negligible for both of the options of available on-chip resources since the added blocks are the counters, the bit-flip detector and a maximum of three registers.

3.4 Comparison with Histogram Techniques

Since the accuracy is mainly determined by HPC, the accuracy of our test scheme is the same as the histogram methods with equal test time.

If an available on-chip memory is assumed (option 1) our scheme also has the same test time as the histogram technique. The advantage of our scheme for option 1 stems from the fact that memory does not have to be accessed every clock cycle. For the histogram methods without an on-chip memory [14, 3, 4], our method will utilize similar implementation area but a very short test time since the time decomposition technique increases test time exponentially as the resolution of the ADC increases. As an example, with no on-chip memory, a $14-bit$ ADC with a sampling frequency of $1MHz$ can be tested in 0.5 seconds with an accuracy of $0.03LSB$. The same test would require more than 5 hours with the histogram method proposed in [4].

Another advantage of our scheme is that it can detect the non-monotonic behavior of the ADC.

4 The Ramp Generator

A common way to generate a voltage ramp is to charge a capacitor by a constant current source as shown in Figure 3. The capacitor voltage will be the ramp voltage having the following expression:

$$V_{ramp}(t) = \frac{I_c}{C} \cdot t \quad (10)$$

where I_c is the current that charges the capacitor C over the time period t.

Since the current source is implemented using an MOS transistor, its finite output impedance, as shown in Figure 3, deviates the ramp voltage from its ideal value and decreases the linearity of the ramp signal.

4.1 Ramp Generator with Feedback

A carefully designed circuit should fix the voltage on the current source such that the charging current will be really *constant*. To obtain an almost constant voltage on the current source, a differential amplifier can be used in a feedback configuration [13] as shown in Figure 4.

If the differential amplifier is ideal i.e. with infinite gain, the circuit will behave as the circuit in Figure 3 except for the direction of the ramp signal (a negative ramp). If the differential amplifier has a finite gain i.e. A, the voltage on the current source will vary but the variation is limited to only a fraction of the ramp voltage, which is:

$$\Delta V_{cs}(t) = \frac{V_{ramp}(t)}{A} \ . \quad (11)$$

Therefore, the output impedance degradation effect can be lowered by increasing the gain of the differential amplifier. The direction of the ramp signal can be easily inverted through an inverting amplifier, which also allows us to increase the effective overall gain. In order to adjust the final value of the ramp, a voltage controlled current source (VCCS) is needed to control the slope of the ramp.

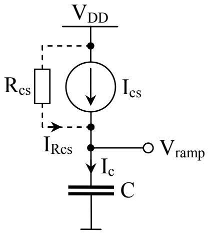

Figure 3. Ramp generation concept and current source non-ideality

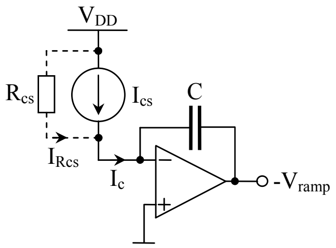

Figure 4. Ramp generation with differential amplifier feedback

4.2 Offset Cancelation

We propose a simple offset cancelation feedback mechanism to initialize the beginning of each ramp to $\pm 10\mu V$ maximum. The feedback loop is composed of a switch, a capacitor and a differential amplifier such that the loop will be closed by a digital signal at the beginning of each ramp signal for a short period of time. The complete ramp generator circuit with offset cancelation feedback can be seen in Figure 5.

The overall circuit operation is controlled by two digital signals to initialize the ramp signal and an analog signal to control the slope of the ramp. This analog signal can also be used in a feedback loop as in [2, 5, 13] to automatically control the ramp slope in the presence of process variations.

4.3 Circuit Implementations

The VCCS and the differential amplifiers are implemented in $0.5\mu m$ CMOS process. The circuits are supplied with $\pm 1.65V$ supply voltages. The output stage transistors are adjusted to have minimum offset voltage at the outputs of the circuits.

The high output impedance VCCS is realized by a cascode current mirror and a high power supply rejection ratio (PSRR) self biasing network, which is controlled by the slope control voltage. The bias generated by the applied slope control voltage in the self biasing network ($M1$ through $M5$), is turned into the capacitor charging current by the cascode current source of $M6$ and $M7$. The CMOS implementation of the circuit can be seen in Figure 6.

The amplifier block is a regular, two stage, differential input, single-ended output circuit [9].

The analog part of the BiST scheme which consists of the ramp generator covers a chip area of $0.017mm^2$ for an off-chip charging capacitor. With an on-chip capacitor the implementation area will be $0.9mm^2$. The chip layout of the ramp generator can be seen in Figure 7.

4.4 Post Layout Simulations

Post layout simulations are performed for each block in HSpice individually to determine the parameters of the VCCS and to adjust the compensation network for the stability of the differential amplifier.

In HSpice simulations, the ramp generator circuit is configured by the control signals to produce consecutive voltage ramps of $1ms$ duration with a full-scale range from $0V$ to $1V$. The final value of the negative ramp signal (the output of the first differential amplifier) is adjusted to $-0.5V$ by the slope control voltage. The inverting amplifier is set to have a gain of -2 to obtain the $1V$ final ramp value. At the beginning of each voltage ramp, the ramp capacitor is discharged by the switches and the output DC offset is canceled by the offset cancelation feedback loop. This initialization phase takes about 50μ seconds.

The overall circuit simulations show that the generated ramp voltage signal has $15-bit$ linearity over $1V$ full-scale range with $\pm 10\mu V$ maximum DC offset error around $0V$. Figure 8 and Figure 9 show the V_{ramp} signal and the corresponding INL error for a single ramp portion respectively.

5 Conclusion

In this paper, we propose a complete ADC BiST scheme based on sequential code analysis at the output rather than

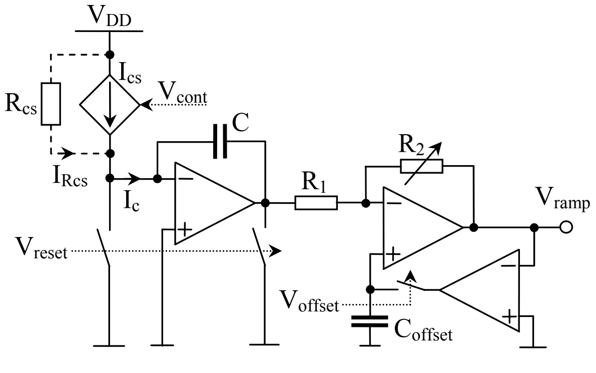

Figure 5. Complete ramp generator circuit with offset cancelation

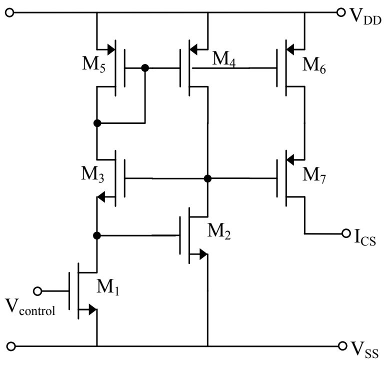

Figure 6. The VCCS circuit

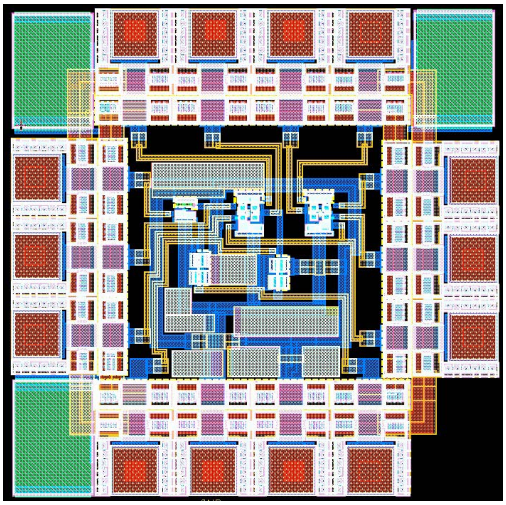

Figure 7. The ramp generator layout

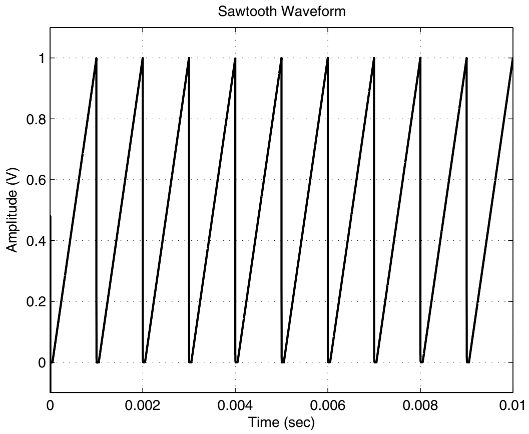

Figure 8. The generated V_{ramp} signal

code frequency analysis, as in histogram based testing. We also propose a ramp generator with $15-bit$ linearity over $1V$ full-scale range. Our analysis scheme has several advantages over the traditional histogram based analysis. First, it is capable of detecting non-monotonicity. Second, when an on-chip memory is available, our scheme does not need fast access to the memory, which is practically hard to achieve. Third, when no on-chip memory is available, histogram based techniques require too long of a test time to be practically applicable, whereas our scheme does not increase the test time. We believe the proposed sequential code analysis is preferable to histogram based techniques when ramp inputs are used.

References

[1] K. Arabi and B. Kaminska. Oscillation Built-In Self-Test (OBIST) Scheme for Functional and Structural Testing of Analog and MixedSignal Integrated Circuits. In *IEEE ITC*, pages 786–795, Nov 1997.

[2] F. Azaïs, S. Bernard, Y. Bertrand, X. Michel, and M. Renovell. A low-cost adaptive ramp generator for analog BIST applications. In *IEEE VTS*, pages 266–271, Apr 2001.

[3] F. Azaïs, S. Bernard, Y. Bertrand, and M. Renovell. A Low-Cost BIST Architecture for Linear Histogram Testting of ADCs. *Springer JET*, 17(2):139–147, Apr 2001.

[4] F. Azaïs, S. Bernard, Y. Bertrand, and M. Renovell. Implementation of a Linear Histogram BiST for ADCs. In *IEEE DATE*, 2001.

[5] S. Bernard, F. Azaïs, Y. Bertrand, and M. Renovell. A High Accuracy Triangle-Wave Signal Generator for On-Chip ADC Testing. In *IEEE ETW*, pages 89–94, May 2002.

[6] J. Blair. Histogram Measurement of ADC Nonlinearities Using Sine Waves. *IEEE TIM*, 43(3):373–383, Jun 1994.

[7] R. de Vries, T. Zwemstra, E. Bruls, and P. Regtien. BuiltIn SelfTest Methodology for A/D Converters. In *IEEE ED&TC*, pages 353–358, Mar 1997.

[8] A. Frisch and T. Almy. HABIST: Histogram-Based Analog Built-In Self-Test. In *IEEE ITC*, page 760767, Nov 1997.

[9] Johns and Martin. *Analog Integrated Circuit Design*. John Wiley and Sons, Canada, 1997.

[10] J. Larrabee, F. Irons, and D. Hummels. Using Sine Wave Histograms to Estimate ADC Dynamic Error Functions. *IEEE TIM*, 47(6):1448–1456, Dec 1998.

[11] A. K. Lu and G. W. Roberts. An Analog Multi-Tone Signal Generator for Built-In Self-Test Applications. In *IEEE ITC*, pages 650–659, Oct 1994.

[12] A. K. Lu, G. W. Roberts, and D. A. Johns. A High-Quality Analog Oscillator Using Oversampling D/A Conversion Techniques. *IEEE TCAS-II*, 41(7), Jul 1994.

[13] B. Provost and E. Sanchez-Sinencio. On-Chip Ramp Generators for Mixed-Signal BIST and ADC Self-Test. *IEEE JSSC*, 38(2):263–273, Feb 2003.

[14] M. Renovell, F. Azaïs., S. Bernard, and Y. Bertrand. Hardware Resource Minimization for a Histogram-based ADC BIST. In *IEEE VTS*, pages 247–252, Apr 2000.

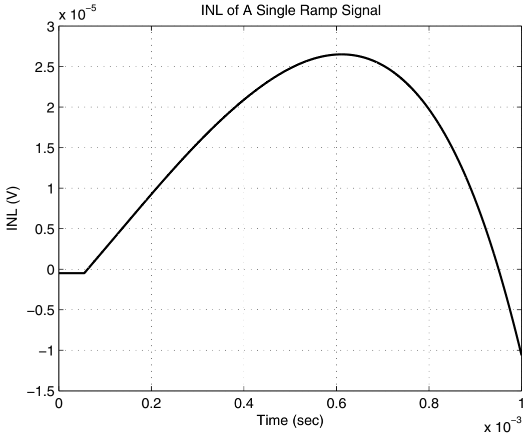

Figure 9. The INL error of a single ramp signal