Code-Width Testing-Based Compact ADC BIST Circuit

Dongmyung Lee, Kwisung Yoo, Kicheol Kim, Gunhee Han, *Member, IEEE*, and Sungho Kang, *Member, IEEE*

***Abstract*—This paper proposes a new analog-to-digital converter (ADC) built-in self-test (BIST) scheme based on code-width and sample-difference testing that does not require a slope-calibrated ramp signal. The proposed BIST scheme can be implemented by a simple digital circuit whose gate count is only approximately 550. The proposed BIST scheme is verified by simulation with 138 test circuits of 6-b pipeline ADC with arbitrary faults. Simulation results show that it effectively detects not only the catastrophic faults but also some parametric faults. The simulated fault coverage is approximately 99%.**

***Index Terms*—Analog-to-digital converter (ADC), built-in self-test (BIST), code width.**

I. Introduction

THE test cost of an integrated circuit often overwhelms the design and the fabrication cost due to the complexity of the implemented system [1], [2]. Various mixed-signal BIST techniques have been introduced to test an analog block embedded in a complex system [3], [4].

The ADC is one of the most frequently used mixed-signal blocks in a system. Several ADC testing schemes were reported in [5]–[9] which can be categorized into the dynamic testing and the static testing. The dynamic testing measures the spectral response of ADC with single tone or multitone sinusoidal input signal to obtain signal-to-noise ratio (SNR), spurious-free dynamic range (SFDR), signal-to-noise and distortion (SINAD), and intermodulation distortion (IMD). This technique requires fast Fourier transformation whose overhead is unacceptably large for built-in self-test (BIST) [10]. The static testing such as histogram, integral nonlinearlity (INL), and differential nonlinearlity (DNL) testing measures the difference between the ADC output and the ideal output. The histogram-based technique is one of the most widely used testing methods with relatively low hardware complexity [11], [12]. However, its overhead is still the primary concern. INL and DNL testing require a slope-calibrated ramp signal generator [13] that requires complex circuits.

This paper proposes a new BIST scheme based on code-width testing that does not require a slope-calibrated ramp signal generator. Section II describes the proposed ADC BIST scheme. Section III presents the BIST implementation and Section IV shows the simulation results. Section V provides the conclusion.

Manuscript received October 6, 2003; revised April 9, 2004. This work was supported in part by the Ministry of Information and Communications, Korea, under the Information Technology Research Center Support Program. This paper was recommended by Associate Editor P. Carbone.

The authors are with the Electrical and Electronic Engineering Department, University of Yonsei, Seoul, Korea (e-mail: dmlee@cad.yonsei.ac.kr; ykwisung@cad.yonsei.ac.kr; kkei99@soc.yonsei.ac.kr; gunhee@yonsei.ac.kr; shkang@yonsei.ac.kr).

Digital Object Identifier 10.1109/TCSII.2004.836034

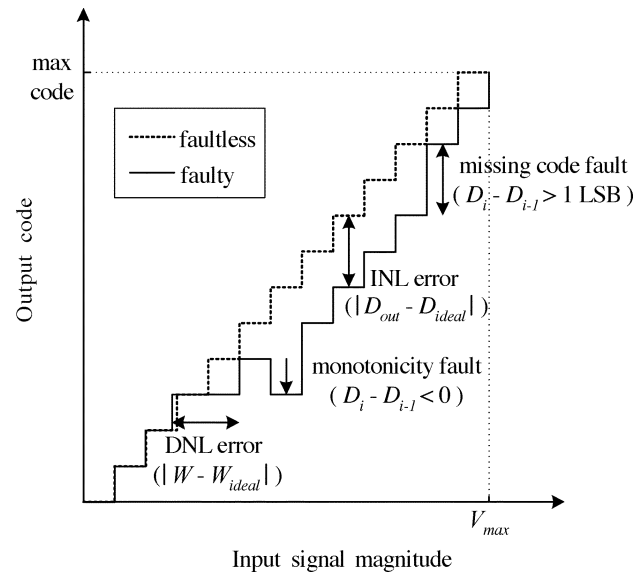

Fig. 1. Definition of ADC faults and errors.

II. ADC BIST Scheme

Various types of static ADC errors and faults are defined in Fig. 1. The missing code fault is defined when the difference of the current ADC output (D_i) and the previous output (D_{i-1}) exceeds 1 LSB when the slope of the ramp input is sufficiently low. The monotonocity fault is defined when the ADC output decreases though the test input signal is monotonically increasing. The INL error is defined by the difference between the ADC output (D_{out}) and the ideal expected output (D_{ideal}). The code width (W) is defined by the width of the analog input that corresponds to 1-LSB digital output. Then the DNL error is defined by the difference between the measured code width and the ideal code width (W_{ideal}).

The proposed BIST scheme is based on code-width measurement as defined in Fig. 2. As the analog test ramp signal, the N–bit ADC under test samples the input signal 2^{N+k} times until the ramp signal reaches maximum input $V_{\max}$. Here, the ramp signal generator should have sufficiently low slope and higher resolution than the ADC but not necessarily be slope-calibrated. Therefore, k is unknown. The ADC's code width for a specific

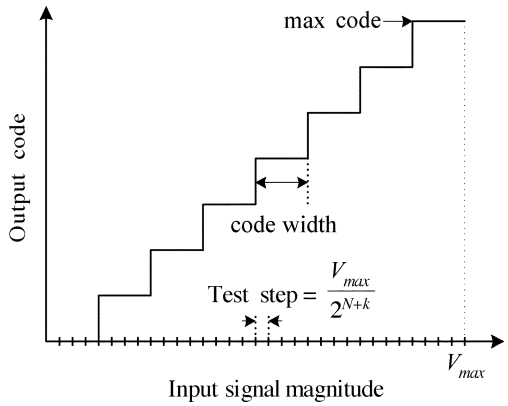

Fig. 2. Definition of code width.

code is measured by counting the number of samples between the ADC output code transition.

The maximum and minimum measured code width can be expressed as follows:

$$\begin{aligned} W_{\max} &= W_{\text{ideal}} + \Delta_{\max} \\ W_{\min} &= W_{\text{ideal}} - \Delta_{\min}, \end{aligned} \quad \Delta_{\max,\min} > 0 \tag{1}$$

where $\Delta_{\max}$ and $\Delta_{\min}$ is the maximum absolute code width deviation from the ideal one. If W_{ideal} is known, then the ADC is categorized as a faulty circuit with DNL error when the deviation Δ exceeds certain boundaries as follows:

$$\Delta = \Delta_{\max} + \Delta_{\min} > \alpha W_{\text{ideal}} \tag{2}$$

where α is a constant that represents the fault decision boundary. $\alpha = 1$ is chosen for most of the applications and the W_{ideal} is often referred to as 1 LSB. Since the proposed BIST scheme does not know W_{ideal}, it checks the code width fault that is defined as follows:

$$W_{\max} > 2W_{\min}. \tag{3}$$

The equation (3) can be manipulated as follows:

$$W_{\text{ideal}} + \Delta_{\max} > 2(W_{\text{ideal}} - \Delta_{\min}). \tag{4}$$

Assuming that $\Delta_{\min} = \Delta_{\max} = (\Delta/2)$, then (4) can be simplified as

$$\Delta > \frac{2}{3} W_{\text{ideal}}. \tag{5}$$

Therefore, the code width test based on (3) corresponds to DNL testing with the decision boundary of 2/3 LSB DNL error.

Since the INL error is caused by the accumulated code width deviation, the INL error at the Jth code can be defined by accumulation of $(W_i - W_{\text{ideal}})$ up to the Jth transition. Then the ADC is considered as a faulty circuit when the magnitude of maximum INL error exceeds certain boundary (typically, $\pm W_{\text{ideal}}$) as follows:

$$\text{INL error} = \max \left| \sum_{i}^{J} (W_{\text{ideal}} - W_i) \right| > \frac{W_{\text{ideal}}}{2}, \quad J = 1 \cdots 2^N. \tag{6}$$

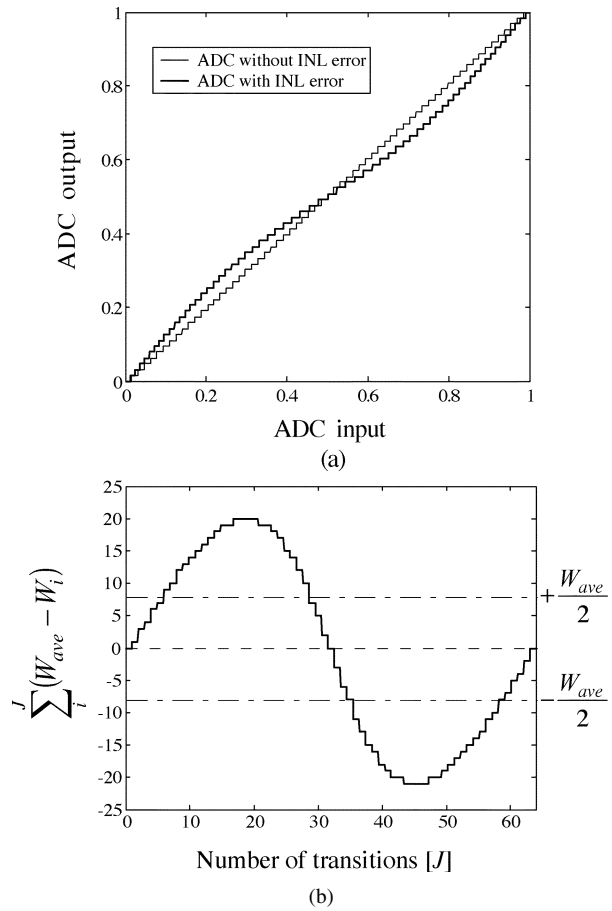

Fig. 3. Example of INL error measurement with a 6-b ADC. (a) ADC transfer curve. (b) INL error obtained from the cumulative code width deviation.

As the W_{ideal} is unknown, the average code width W_{ave} is used instead. If the ADC passes the code width test, then the value of W_{ave} is sufficiently close to the value of W_{ideal}. Fig. 3 shows the INL error that is obtained from the cumulative code width deviation from W_{ave} as an example.

III. BIST Implementation

The proposed BIST circuit is depicted in Fig. 4. The ramp signal generator, the register, and the counters are reset at the beginning of the test. The register holds the ADC output for one sample period. Then the subtractor calculates the difference between the current ADC output and the previous output that is stored in the register. ORing of all bits in the subtractor output detects the *transition* of the ADC output. If the subtractor output is negative, then this represents that the monotonocity fault occurs.

If the subtractor output is larger than 1, then this means that the code transition is larger than 1 LSB and it corresponds to missing code fault. The ORing of all bits except for the LSB in the subtractor output detects the missing code fault.

The code width counter counts the number of clocks until the *transition* occurs. If a transition is detected, then the content of the counter represents the code width for the specific code. Once the code width is checked, then the counter is reset for the next code width measurement. The hold min and hold max block detect the minimum and the maximum code width, respectively. Then the difference between twice the minimum code width and the maximum code width is calculated. If the subtractor output is

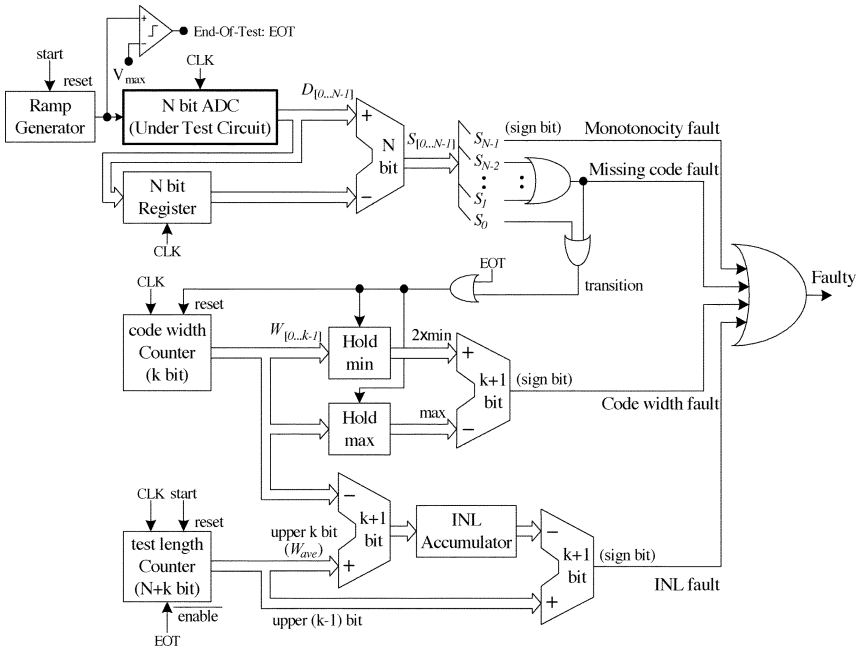

Fig. 4. Conceptual block diagram of the proposed BIST error detector.

TABLE I
FAULT DEFINITIONS AND THEIR DETECTION METHODS

Detected parameters and faults	Definition of faults	Detection method
Transition	–	OR ($S_0, \ldots, S_{N-2}$)
W (code width)	–	Clock counting until transition
Missing code fault	$D_i - D_{i-1} > 1$	OR ($S_1, \ldots, S_{N-2}$)
Monotonocity fault	$D_i - D_{i-1} < 0$	Sign (S)
Code width fault (high gain fault, offset)	$W_{max} > 2W_{min}$	Sign $(W_{max} - \text{shift left}(W_{min}))$
INL fault	$\text{Max}\left\lvert \sum_i^J (W_{ave} - W_i) \right\rvert > \frac{W_{ave}}{2}$	Sign $\left(\text{shift right}(W_{ave}) - \left\lvert \sum_i^J (W_{ave} - W_i) \right\rvert\right)$ @ J-th transition

negative, then it means that $W_{\max} > 2W_{\min}$ which corresponds to the code width fault defined in (3). Multiplication by 2 is implemented by simple interconnection mapping that connects d_i of the $W_{\min}$ to d_{i+1} of the subtractor input and assigns d_0 of the subtractor to be zero. This is equivalent to the shift-left operation to multiply by 2 in binary code.

The offset cancelled comparator generates an end-of-test (EOT) signal when the input ramp signal reaches $V_{\max}$. The test length counter counts the number of clocks until the EOT during the code width testing phase that is explained above. If the ADC has passed the code width testing, then the test length counter has a total summation of code width. Division of this test length by 2^N (where N is the resolution of ADC) provides the average code width W_{ave}. This division is realized by getting rid of the lower N bits of the test length counter. Once the EOT is generated, then the BIST starts the second phase of testing for the INL testing.

The ramp signal generator, the register, and the code width counter are reset. The code width of the ADC is measured in the same way as the code width testing. Then the difference between the measured code width and W_{ave} is accumulated. If the difference between the magnitude of these accumulated code width deviations and $W_{\text{ave}}/2$ is negative, then it means that the INL error exceeds $\pm 1/2$ LSB. Table I summarizes the fault definitions and their detection methods.

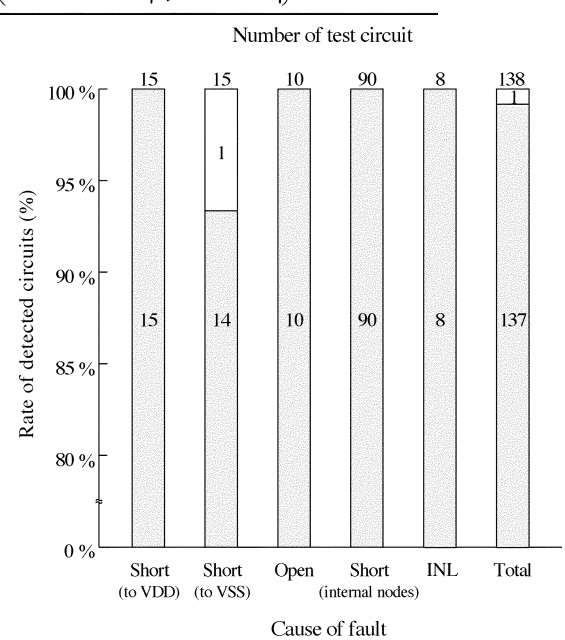

Fig. 5. Simulated fault coverage for 138 test ADCs with arbitrary faults.

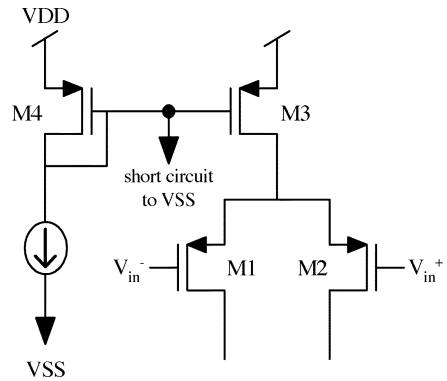

Fig. 6. Example of a faulty circuit that is not detected by the proposed BIST.

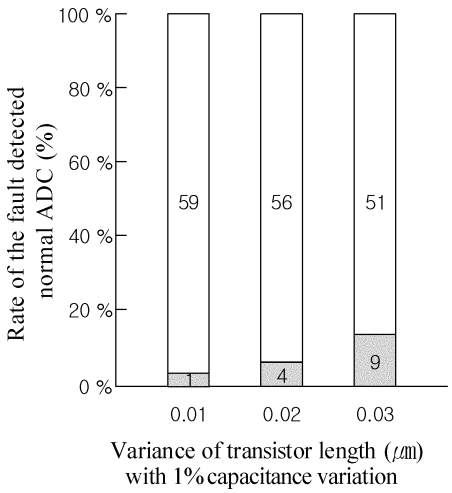

Fig. 7. Rate of the fault-detected normal ADC with process variation.

IV. Simulation Result

The proposed BIST circuit is verified by simulations with 138 test circuits of a 6-b pipeline ADC with arbitrary faults such as open or short or some parametric variation. The error detector is designed with HDL and synthesized with a standard cell. The total gate count is about 550 (270 for the INL testing and 280 for others).

Fig. 5 shows the summary of the simulation results. All faulty test circuits except one test circuit are detected by the proposed BIST circuit, yielding fault coverage of 99%. The test circuit whose fault is not detected by the proposed BIST performs normally for the ramp signal testing. The fault is located at the tail current source in an operational transconductance amplifier (OTA), as shown in Fig. 6. If the resolution or sampling rate of the ADC is high, then this fault can be detected by the proposed BIST because this fault may cause the performance degradation that exceeds the decision boundary. In practice, the fault coverage may be lower than this simulation result because the test circuits do not cover all possible cases.

Monte Carlo analysis has been performed as well to estimate the effect of the process variation. The simulations are performed with different variances of transistor gate length and capacitor mismatch. Fig. 7 shows that increased variance results in an increased number of faulty circuits.

V. Conclusion

This paper proposed a new ADC BIST scheme based on code width testing without a slope-calibrated ramp signal. The proposed BIST can be implemented with a small overhead of 550 gate count, excluding the ramp signal generator and the comparator. Though the required gate count is linearly dependent on the ADC resolution, an optimized design may result in a much smaller gate count than 550. The simulated fault coverage was 99% with 138 test circuits.

The simulation results show that the proposed ADC BIST scheme can detect most catastrophic faults and some parametric faults as well. Due to noise or any uncertainty in a real circuit, the test should be performed several times, and the BIST controller will finally decide that the ADC is faulty when the fault detection is repeated.

References

[1] M. F. Toner and G. W. Roberts, "On the practical implementation of mixed analog-digital BIST," in *Proc. IEEE Custom Integrated Circuits Conf.*, May 1995, pp. 525–528.

[2] M. Lu and C. W. Wu, "Cost and benefit models for logic and memory BIST," in *Proc. DATE*, Mar. 2000, pp. 710–714.

[3] M. M. Hafed and G. W. Roberts, "A stand-alone integrated excitation/extraction system for analog BIST applications," in *Proc. IEEE Custom Integrated Circuits Conf.*, May 2000, pp. 83–86.

[4] M. B. Santos, F. M. Goncalves, M. Ohletz, and J. P. Teixeira, "Defect-oriented testing of analogue and mixed signal ICs," in *Proc. IEEE Int. Conf. Electronics, Circuits and Systems*, vol. 2, Sept. 1998, pp. 419–424.

[5] M. Ehsanian, B. Kaminska, and K. Arabi, "A new digital test approach for analog-to-digital converter testing," in *Proc. 14th VLSI Test Symp.*, Apr./May 1996, pp. 60–65.

[6] K. Arabi and B. Kaminska, "Efficient and accurate testing of analog-to-digital converters using oscillation-test method," in *Proc. Eur. Design and Test Conf. (ED&TC)*, Mar. 1997, pp. 348–352.

[7] R. de Vries, T. Zwemstra, E. M. J. G. Bruls, and P. P. L. Regtien, "Built-in self-test methodology for A/D converters," in *Proc. Eur. Design and Test Conf. (ED&TC)*, Mar. 1997, pp. 353–358.

[8] J. L. Huang, C. K. Ong, and K. T. Cheng, "A BIST scheme for on-chip ADC and DAC testing," in *Proc. DATE 2000*, Mar. 2000, pp. 216–220.

[9] Y. C. Wen and K. J. Lee, "An on chip ADC test structure," in *Proc. DATE 2000*, Mar. 2000, pp. 221–225.

[10] E. Teraoka, T. Kengaku, I. Yasui, K. Ishikawa, T. Matsuo, H. Wakada, N. Sakashita, Y. Shimazu, and T. Tokuda, "A built-in self-test for ADC and DAC in a single-chip speech CODEC," in *Proc. Int. Test Conf.*, Oct. 1993, pp. 791–796.

[11] F. Azais, S. Bernard, Y. Bertrand, and M. Renovell, "Implementation of a linear histogram BIST for ADCs," in *Proc. DATE 2001*, Mar. 2001, pp. 590–595.

[12] K. Parthasarathy, L. Jin, T. Kuyel, D. Chen, and R. Geiger, "A histogram based AM-BIST algorithm for ADC characterization using imprecise stimulus," in *Proc. 14th Midwest Symp.*, vol. 2, Aug. 2002, pp. 274–277.

[13] B. Provost and E. Sánchez-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.