

BIST Module for Mixed-Signal Circuits

S. Demidenko¹⁻², V. Piuri³, V. Yarmolik⁴, A. Shmidman⁴

¹EC Department, Singapore Polytechnic, Singapore

²Institute of Engineering Cybernetics, National Academy of Sciences of Belarus, Belarus

³Department of Electronics and Information, Politecnico di Milano, Italy

⁴CS Department, Belarusian State University of Informatics and RadioElectronics, Belarus

Abstract

T-flip-flop implementation of the universal module (signature analyzer and test generator) for built-in self-test of mixed signal circuits is proposed and analyzed.

1. Introduction

Importance of the *mixed-signal testing* and, thus, the interest to the area have grown increasingly during the last years. Intensive research and development have been carried out by many specialists in different countries (e.g., see [1-4]). Among these techniques and approaches, the *built-in self-test (BIST)* solution was proved to be one of the most promising and offering the best features for the practical use. The built-in self-test approach implies that control, test stimuli generation, and measurement (test response analysis) units are incorporated into the *device under test (DUT)*: this allows to perform on-chip testing of the internal circuitry and deliver the test results to the external terminals.

The idea of using "universal" hardware for test stimuli generation and test response analysis both for the digital and the analog circuitry of the DUT was presented in [5], as *Hybrid Built-In Self-Test (HBIST)*. The two main objectives pursued by HBIST are:

- to provide consistency in testing both the analog and the digital parts of the DUT;
- to minimize the test overhead by using the same hardware for analog and digital testing.

By far the most important test tool of the HBIST architecture is the *BILBO*-type [6] register: it is used for test pattern generation and test response evaluation both for the digital and the analogue subsystems (Fig. 1) [5]. The BILBO register is thus a universal module, that is able to perform the following basic functions: parallel register, serial shift register, parallel signature register, and linear feedback shift register. Its architecture is shown in Fig. 2, where x_i (with $i = \overline{1, n}$) are the inputs, y_i (with $i = \overline{1, n}$) are the outputs, $C1, C2, C3$ are the control signals, and $CLOCK$ is the synchronization signal. Traditionally the BILBO register is realized using *D-type flip-flops (D-FF)*.

Since its introduction in the late 70s, BILBO has been predominantly used for pure digital circuit testing [6,7]. Therefore, the following questions remain regarding its application in the mixed-signal circuit environment:

- what kind of universal stimuli have to be generated by the register in order to satisfy the testing requirements both for the analog and the digital parts of DUT?
- what kind of the test response compression technique has to be applied?
- what components have to be used to implement the BILBO register itself in order to maximize its efficiency both in digital and analog test generation and analysis?

2. Fault Models

Faults in analog circuits can be classified into *Catastrophic (Hard)* and *Deviation (Soft) Faults* [3,5]. The catastrophic faults are caused by random defects (dust particles, strong external heat or x-ray exposure, “infant mortality”, etc.) and lead to shorts/opens or to significant changes in the electrical and/or timing parameters of the circuit. Failures of this class manifest themselves in a totally-malfunctioning circuit. They can be detected in a relatively easy way. However, it is difficult to locate their cause and, if possible, to correct it.

The soft faults are caused by relatively less significant deviations from the circuit parameters specified by its data sheet. While propagating through the circuit, a signal may deviate for some error $\pm\Delta$ from its specified value, due to a fault. The soft faults manifest themselves in deterioration in circuit’s ability to work according to its function.

The use of digital tools for the test response analysis in mixed-signal BIST provides substantial benefits mainly in terms of hardware overhead, easiness of implementation, simplicity of control, efficiency in test data processing, and interpretation of the test results [1,5,8]. The choice of the particular digital tool which is the most suitable in the mixed-signal circuit environment depends on the mapping of the analog errors ($\pm\Delta$) onto the set of digital errors ($\pm k$) having multiplicity μ as a result of the analog-to-digital conversion. In this paper the error multiplicity μ ($\mu \in \{1, 2, \dots, N\}$) is defined as the number of erroneous bits in the N -bit digital word representing an analog-signal sample in the presence of an analog error, where N is the width (i.e., the number of bits) of the *analog-to-digital converter (ADC)*.

In this paper the additive error model is adopted. A digitized signal is considered as the sum of an error-free component (corresponding to the analog-to-digital conversion of the error-free analog signal samples) and a random error. Thus, the value of the digitized sample of the distorted signal (i.e., signal affected by an error) depends on the digital representation of the error itself (Fig. 3). For example, if an error-free component is *001101* and the digitized value of the error is $k=+5$ (101 in the binary notation), then the bits 1, 2, 3, 4, and 5 of the resulting code are affected by the error (as shown in bold face in Fig. 3a). If the error value is $k=+13$ (1101 in the binary notation), then the bits 1, 2, 3, and 5 of the resulting code differ from the value of the fault-free signal (Fig. 3b). Fig. 3c shows an example of the code distortion corresponding to the errors of $k=+10$.

Zero-valued least-significant bits (LSBs) of the error code do not contribute to the distortion of the digital code (e.g., see the zero LSB for $k=+10$ in the above example). The other zeros do not directly affect the resulting value of the digitized samples as well: even if the corresponding bits can be affected due to carry propagation from the less significant bits (e.g., bit 5 for $k=+10$). The maximal error multiplicity μ_{MAX} thus depends both on the width of the ADC and on number of zero-valued LSBs in the error code. It is $\mu_{MAX} = N - j$, where N is the width of the ADC and j is the number of zero-valued LSBs.

In general, the digital error can be represented as $k = B \cdot 2^j$, where 2^j (with j equal to a suitable non-negative integer number) represents the set of zero-valued LSBs and B is the odd integer number representing the remaining most-significant part of the error code. For example, the error code $k=+10$ can be presented as $101 \cdot 2^1$.

Small deviations $\pm\Delta$ of the analog signal lead to small digital errors $\pm k$ and, in turn, to small values of B and j . Since small erroneous deviations are far more probable than the big ones [8], we focus on the error distributions for small k ’s. By using the above expressions for k and μ_{MAX} and by grouping the errors having the same value of B , the total number of errors having a given multiplicity can be computed. For example, for $k=1 \cdot 2^j$, $3 \cdot 2^j$, and $5 \cdot 2^j$, the total number C of possible errors of multiplicity μ is, respectively

$$C = \begin{cases} 2^{\log_2(k)+2}, \mu = N - \log_2(k) \\ 2^{N-\mu+1}, \mu = 1..N - \log_2(k) - 1 \end{cases} \quad (1)$$

$$C = \begin{cases} 2^{\log_2(k/3)+2}, \mu = N - \log_2(k/3) \\ 3 \cdot 2^{\log_2(k/3)+2}, \mu = N - \log_2(k/3) - 1 \\ 2^{N-\mu+2}, \mu = 2..N - \log_2(k/3) - 2 \end{cases} \quad (2)$$

$$C = \begin{cases} 2^{N-1}, \mu = 2 \\ 3 \cdot 2^{N-\mu+1}, \mu = 3..N - \log_2(k/5) - 2 \\ 5 \cdot 2^{\log_2(k/5)+2}, \mu = N - \log_2(k/5) - 1 \\ 2^{\log_2(k/5)+2}, \mu = N - \log_2(k/5) \end{cases} \quad (3)$$

Simulation of 9-bit analog-to-digital conversion for $k = \overline{1,6}$ was done to practically evaluate the error distribution: results are given in Fig. 4. These results confirm that small-multiplicity errors constitute the vast majority of the total number of errors caused by soft faults in the analog circuitry of the mixed-signal DUT. Since signature analysis is particularly efficient in detecting small-multiplicity errors, the above result further supports the idea of using BILBO-type registers for the test response analysis.

3. Test stimuli generation

Digital BIST techniques are traditionally based on the use of *Linear Feedback Shift Registers (LFSR)* to realize *Pseudo-Random Test Pattern Generation (PRTPG)*. The universal BILBO register in one of its operation modes is an 8-bit LFSR.

Generation of the test stimuli by using LFSR can be adopted to test the analog blocks of the mixed-signal circuits [1,5,8]. The main requirement for the test signal to be generated is that of having “white noise” characteristics [9]. Ideally, the generated test signal must have an unlimited frequency band and a uniform frequency spectrum (i.e., the signal auto-correlation function has to be the δ -function).

The direct use of PRTPG realized by the traditional LFSR is not the best choice for implementation in the analog sub-circuitry testing. This is due to the imperfect characteristics of the generated sequences [10]. In order to satisfy the “white noise” criteria, the following LFSR-based algorithm was proposed for the test stimuli generation [5]

$$U_p(t+1) = 2^{N-1}a_i(t) + 2^{-1}[U_p(t) - a_N(t)] \quad (4)$$

where $U_p(t+1)$ is the signal generated at the $t+1$ -th time step, and $a_i(t) \in \{-1, +1\}$ (with $i=1,2,3,...,N$) are the LFSR-generated bits in which 0's and 1's have been substituted by -1's and +1's, respectively.

The auto-correlation function $K(\tau)$ of the sequence generated by (4) is far from being ideal. It is known that, for any long LFSR, $K(\tau)$ can be roughly approximated by the exponential function $2^{-\tau}$ (with $\tau=1,2,...,N$) [7,9], leading to a lower fault coverage. To generate better quality test stimuli, the BILBO implementation can be enhanced by introducing the *decimation* of the LFSR-generated sequence [9]: the PRTPG will thus produce non-correlated patterns at every operation step. The use of T-type flip-flops, as proposed in this paper, allows to realize this approach in the most economical way.

4. Module Implementation

Flip-flops of T-type (T-FF) (Fig. 5) are proposed to implement the BILBO-like universal module for the mixed-signal BIST. The choice of T-FF is supported by the following theorem.

Theorem. *A sequence generated by T-FF LFSR coincides with a decimated pseudo-random sequence generated by the conventional LFSR. The decimation coefficient is equal to or greater than N , where N is the length of the shift register.*

The proof of this theorem can be derived from the results presented in [11]. In that paper, the authors proved that the PRTPG realized with T-type flip-flops follows the states of the sequential generation based on D-type flip-flops by sampling such a sequence with a suitable sampling period.

Obviously, the LFSR which completely changes its state at each clock pulse generates a “white noise” sequence with better quality than that of a traditional realization based on D-FFs. With D-FFs, the generated patterns are correlated every N consecutive clock cycles.

In addition to improving the test pattern generation properties, the T-FF implementation enhances the fault detection capability of the module. The BILBO-type register is in fact used both for the test stimuli generation and the test response analysis in the mixed-signal BIST environment. It was shown [12] that a BILBO-based Multi-Input Signature Register realized with D-flip-flops provides very limited error coverage due to the presence of error masking in the register channels. Implementation of the MISR by using T-FFs leads to a highly better fault coverage due to the substantial elimination of the masking effect.

5. Conclusion

The paper proposes the T-FF implementation of the BILBO-type BIST module for mixed-signal testing. Such an implementation leads to generate quasi “white noise” test signals having better quality with respect to conventional approaches. Besides, the module provides an enhanced fault coverage for the multi-input signature compression in the case of low-multiplicity errors (which are the dominating ones in the mixed-signal environment).

References

- [1] Cheng, K.-T. (1996), Built-In Self Test for Analog and Mixed Signal Designs: The Opportunities and Challenges - Keynote Presentation, *Proceedings ATS'96*, Nov., 28 pp.
- [2] Roberts, G. (1996), Mixed-Signal Testing - Tutorial 5, *ITC'96 Conference*, Oct.
- [3] Soma, M. (1996), Mixed-Signal Fault Models and Design for Test (I) - Tutorial 2, *ATS'96 Symposium*, Nov.
- [4] Wey, C.-L. (1996), Mixed-Signal Fault Models and Design for Test (II) - Tutorial 4, *ATS'96 Symposium*, Nov., 1996.
- [5] Ohletz, M.J. (1991), Hybrid Built-In Self-Test (HBIST) for Mixed Analogue/Digital Integrated Circuits, *Proceedings ETC'91*, Apr., 307-316
- [6] Koenemann, B., Mucha, J. and Zwiehoff, G. (1979), Built-In Logic Block Observation Techniques, *Proceedings ITC '79*, Oct., 37-41.
- [7] Bardell, P.H., McAnney, W.N. and Savir, J. (1987), *Built-In Self-Test for VLSI*, John Wiley.
- [8] Pan, C.-Y. and Cheng, K.-T. (1995), Pseudo-Random Testing and Signature Analysis for Mixed-Signal Circuit, *Proceedings ICCAD-95*, Nov., 102-107
- [9] Korn G.A., Korn T.M. (1964), *Electronic Analog and Hybrid Computers*, Mc Graw Hill
- [10] Yarmolik, V.N. and Demidenko, S.N. (1988), *Generation and Application of Pseudo-random Sequences for Random Testing*, John Wiley
- [11] Yarmolik, V.N. and Murashko, I.A. (1995), A new test pattern generator design approach for VLSI built-in self-testing, *Avtomatika i telemekhanika*, N5, 25-35
- [12] Kachan I.V. (1990), Digital Circuits of Self-Testing VLSI, PhD Dissertation, Belarusian State University of Informatics and RadioElectronics, Minsk, 180 pp.

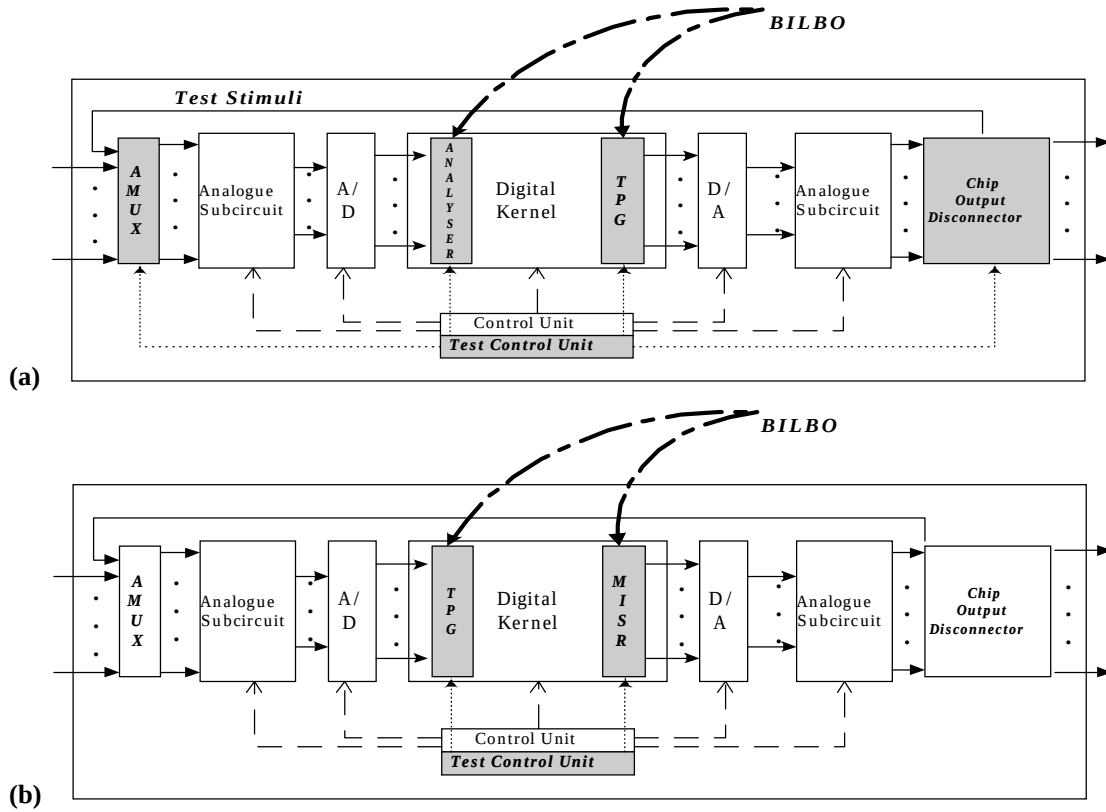


Figure 1: HBIST: (a) analog sub-circuit testing, (b) digital sub-circuit testing.

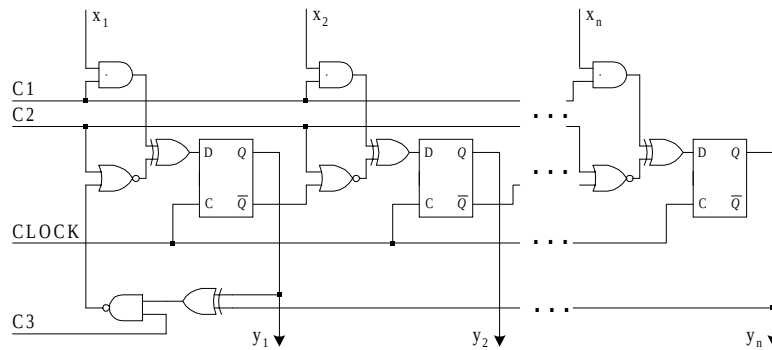


Figure 2: BILBO.

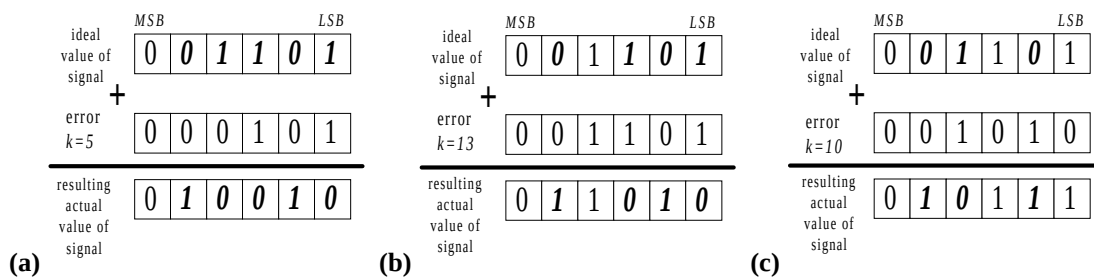
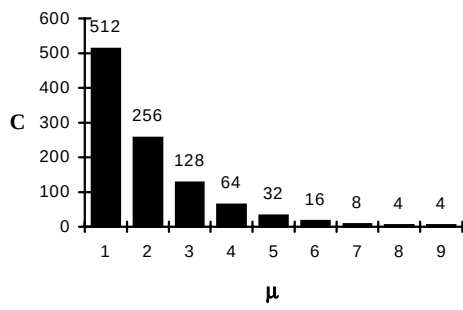
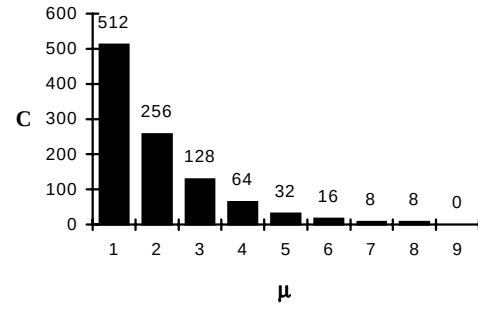


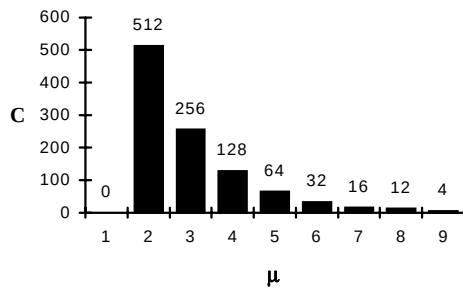
Figure 3: Digital errors of multiplicity: (a) $\mu=5$, (b) $\mu=4$, and (c) $\mu=3$.



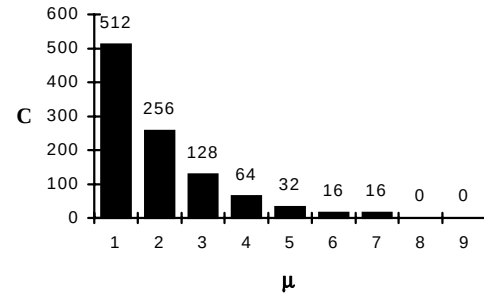
a) $k=1, N=9$



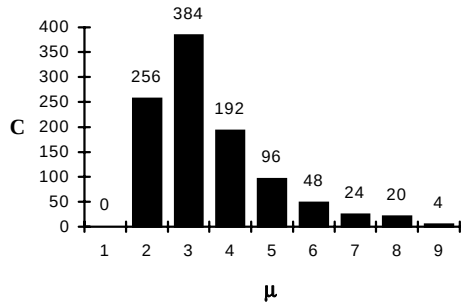
b) $k=2, N=9$



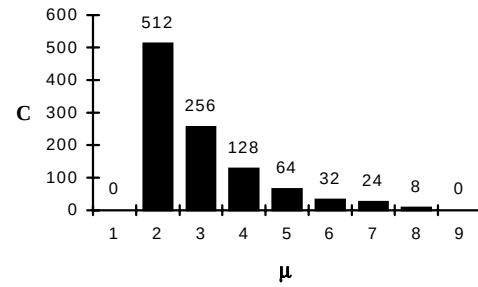
c) $k=3, N=9$



d) $k=4, N=9$

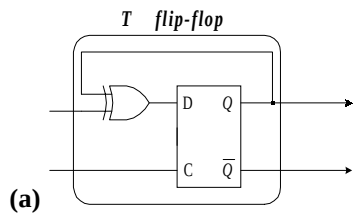


e) $k=5, N=9$

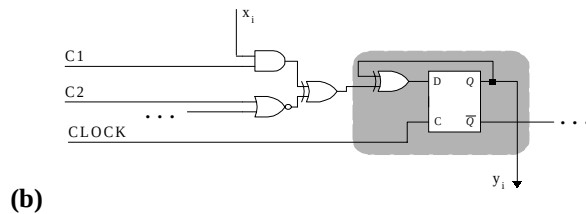


f) $k=6, N=9$

Figure 4: The number C of errors having multiplicity μ .



(a)



(b)

Figure 5: The BILBO register based on the T-type flip-flop: (a) the T-FF and (b) the building block of the BILBO register.