

Design and Implementation of the Arithmetic Circuits testing using Accumulator based Built-in Self Test

R. Anitha¹, V.P. Kolanchinathan²PG Scholar, Dept. of ECE, SVCET, Thiruvallur, India ¹Associate Professor, Dept. of ECE, SVCET, Thiruvallur, India ²

ABSTRACT: Current VLSI circuits, e.g., data path architectures or digital signal Processing chips frequently contain arithmetic modules such as accumulators, Arithmetic Logic Units (ALUs). The basic idea of ABIST is to utilize accumulators for built-in testing which is compression of the CUT responses or generation of test patterns and has been shown to result in low hardware overhead and low impact on the circuit speed. In this work an accumulator-based test pattern generation scheme that compares with the previously proposed schemes and it was proved that the test vectors generated by an accumulator whose inputs are driven by a constant pattern can have acceptable pseudorandom individuality, if the input pattern is appropriately selected. Weighted pseudorandom Built-In Self-Test (BIST) schemes have been utilized in order to drive down the number of vectors to achieve complete fault coverage in BIST applications. Impartial sets comprising three weights, namely 0, 1, and 0.5 can be successfully utilized for test pattern generation, because they result in both low testing time and less power consumption. In this work, the digital circuits such as s27 Bench mark and SISO are, accumulator based BIST. Finally, the synthesis results of these circuits are compared with normal BIST. It was found that the circuit as low delay using accumulator based BIST.

KEYWORDS: BIST, ALU, CUT, ABIST, Combinational Circuits.

I. INTRODUCTION

To test integrated circuits and systems, pseudorandom built-in self-test (BIST) generators have been widely utilized. The arsenal of pseudorandom generators includes accumulators driven by a constant value [3]. A large number of random patterns have to be generated to achieve high fault coverage for circuits with hard to detect faults. Therefore, weighted pseudorandom techniques have been proposed where inputs are biased by changing the probability of a "0" or a "1" on a given input from 0.5 (for pure pseudorandom tests) to some other value [10], [15]. Some faults may require long test sequences to be detected, complete fault coverage usually can't be achieved by the weighted random pattern generation methods that depend on a single weight assignment. Although the weights are computed to be suitable for most faults, by using a reasonable number of test patterns. If their activation and propagation requirements do not match. Different faults require different biases of the input combinations applied to the circuit, to ensure that a relatively small number of patterns can detect all faults [4], multiple weight assignments have been suggested. Approaches to derive weight assignments for given deterministic tests are attractive since they have the potential to allow complete coverage with a significantly smaller number of test patterns [10]. For the minimization of the hardware implementation cost, other schemes were introduced based on multiple weight assignments by utilization of weights 0, 1, and 0.5. By this approach some outputs of the generator remain steady (together 0 or 1) and letting the remaining outputs change values (pseudo-) randomly (weight 0.5). This approach, not only reduces the hardware overhead but also has a beneficial effect on the consumed power, since some of the circuit under test (CUT) inputs (those having weight 0 or 1) remain steady during the specific test session [30]. Pomeranz and Reddy [5] proposed a 3-weight pattern generation scheme relying on weights 0, 1, and 0.5. The choice of weights 0, 1, and 0.5 was done in order to minimize the hardware implementation cost. Wang [8], [13] proposed a 3-weight random pattern generator based on scan chains utilizing weights 0, 1, and 0.5, in a way similar to [5]. Recently, Zhang et al. [9] renovated the interest in the 3-weight pattern generation schemes, proposing an efficient compaction scheme for the 3-weight patterns 0, 1, and 0.5. From the above we can conclude that 3-weight pattern generation based on weights 0, 1, and 0.5 has practical

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

interests since it combines low implementation cost with low test time. Current VLSI circuits, e.g., data path architectures, or digital signal processing chips commonly contain arithmetic modules [accumulators or arithmetic logic units (ALUs)]. Thus the idea of arithmetic BIST (ABIST) [6] arises. The basic idea of ABIST is to utilize accumulators for built-in testing (compression of the CUT responses, or generation of test patterns) and has been shown to result in low hardware overhead and low impact on the circuit normal operating speed [22]–[27]. In [22], Manich et al. presented an accumulator-based test pattern generation scheme that compares favourably to previously proposed schemes. In [7], it was proved that the test vectors generated by an accumulator whose inputs are driven by a constant pattern can have acceptable pseudorandom characteristics, if the input pattern is properly selected. However, modules containing hard-to-detect faults still require extra test hardware either by inserting test points into the mission logic or by storing additional deterministic test patterns. In order to overcome this problem, accumulator weighted pattern generation scheme was proposed in [11]. The scheme generates test patterns having one of three weights, namely 0, 1, and 0.5 therefore it can be utilized to drastically reduce the test application time in accumulator-based test pattern generation. However, the scheme proposed in [11] possesses three major drawbacks: 1) it can be utilized only in the case that the adder of the accumulator is a ripple carry adder; 2) it requires redesigning the accumulator; this modification, apart from being costly, requires redesign of the core of the datapath, a practice that is generally discouraged in current BIST schemes; and 3) it increases delay, since it affects the normal operating speed of the adder. A scheme for accumulator-based 3-weight generation is presented in [33]. This scheme copes with the inherent drawbacks of the scheme proposed in [11]. More precisely: 1) It does not impose any requirements about the design of the adder (i.e., it can be implemented using any adder design); 2) it does not require any modification of the adder; and hence, 3) does not affect the operating speed of the adder. Furthermore, the proposed scheme compares favourably to the scheme proposed in [11] and [22] in terms of the required hardware overhead. But it has its drawbacks too. By using the adder in this scheme the complexity of the hardware used increases. In this paper a novel scheme for weighted pattern generation of an accumulator is presented. Replacing the adder by XOR gate we can cope with that previous scheme [33], thus reducing the hardware, complexity for the test pattern generation.

II. WEIGHTED PATTERN GENERATION OF AN ACCUMULATOR

The idea of a weighted pattern generation of an accumulator was illustrated by means of an example. Let us consider the test set for the c17 ISCAS benchmark [12], [31] given in Table I. In order to apply the 3-weighted pattern generation scheme, one of the schemes proposed in [5], [8], and [9] can be utilized. According to these schemes, a typical weight assignment procedure would involve separating the test set into two subsets, S_1 and S_2 as follows: $S_1 = \{T_1, T_4\}$ and $S_2 = \{T_2, T_3\}$. The weight assignments for these subsets is $W(S_1) = \{-, -, 1, -, 1\}$, and $W(S_2) = \{-, -, 0, 1, 0\}$, where a “_” denotes a weight assignment of 0.5, a “1” indicates that the input is constantly driven by the logic “1” value, and “0” indicates that the input is driven by the logic “0” value. In the first assignment, inputs $A[2]$ and $A[0]$ are constantly driven by “1”, while inputs $A[4]$, $A[3]$, $A[1]$ are pseudo randomly generated (i.e., have weights 0.5). Similarly, in the second weight assignment (subset S_2), inputs $A[2]$ and $A[0]$ are constantly driven by “0”, input $A[1]$ is driven by “1” and inputs $A[4]$ and $A[3]$ are pseudo randomly generated.

These are the reasoning calls for a configuration of the accumulator, where the following conditions are met: 1) an accumulator output can be constantly driven by “1” or “0” and 2) an accumulator cell with its output constantly driven to “1” or “0” allows the carry input of the stage to transfer to its carry output unchanged. This latter condition is required in order to effectively generate pseudorandom patterns in the accumulator outputs.

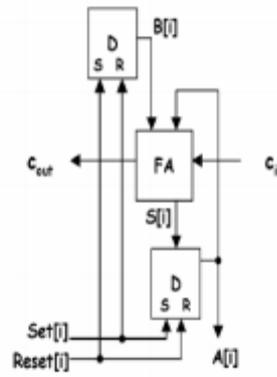
III. DESIGN METHODOLOGY

The implementation of the weighted pattern generation scheme is based on the accumulator cell presented in Fig. 1, which consists of a Full Adder (FA) cell and a D-type flip-flop with asynchronous set and reset inputs whose output is also driven to one of the full adder inputs. In Fig. 1, we assume that the set and reset are active high signals. For this accumulator cell, one out of three configurations can be utilized, as shown in Fig. 2.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

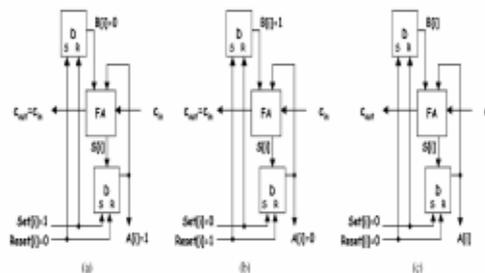
Vol. 5, Issue 3, March 2016



Accumulator Cell

we present the configuration that drives the CUT inputs when $A[i]=1$ is required. $Set[i] = 1$ and $Reset[i] = 0$ and hence $A[i] = 1$ and $B[i] = 0$. Then the output is equal to 1, and C_{in} is transferred to C_{out} .

we present the configuration that drives the CUT inputs when $A[i]=0$ is required. $Set[i] = 0$ and $Reset[i] = 0$. The D input of the flip-flop of register B is driven by either 1 or 0, depending on the value that will be added to the accumulator inputs in order to generate satisfactorily random patterns to the inputs of the CUT. Replacement of adder by a Xor gate is done. Generally A full adder can be implemented in many different ways such as with a custom transistor-level circuit or composed of other gates. In this implementation, the final OR gate before the carry-out output may be replaced by an XOR gate without altering the resulting logic. Using only two types of gates is convenient if the circuit is being implemented using simple IC chips which contain only one gate type per chip. The general configuration of the proposed scheme is presented. The Logic module provides the $Set[n-1:0]$ and $Reset[n-1:0]$ signals that drive the S and R inputs of the Register A and Register B inputs. Note that the signals that drive the S inputs of the flip-flops of Register A, also drive the R inputs of the flip-flops of Register B and vice versa.



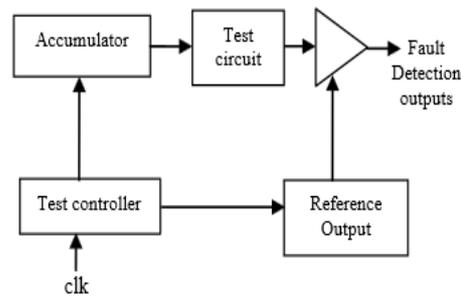
Configurations of Accumulator Cell

In Section IV-B, we shall compare the proposed scheme with the 3-weight scan schemes that have been proposed in [5] and [8]. In Section IV-C, we shall compare the proposed scheme with accumulator based 3-weight pattern generation that has been proposed. The test application algorithms that have been invented and applied by previous researchers, e.g., [5], [8], [9] can be equally well applied with both implementations based on the number of test patterns applied by [11]. Therefore, the comparison will be performed with respect to: 1) the hardware overhead and 2) the impact on the timing characteristics of the adder of the accumulator. For the proposed scheme, no modification is imposed on the adder of the accumulator. Therefore, there is no impact on the datapath timing characteristics.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016



Proposed ABIST

In the 3-weight pattern generation scheme proposed by Pomeranz and Reddy in [5] the scan chain is driven by the output of a linear feedback shift register (LFSR). Logic is inserted between the scan chain and the CUT inputs to fix the outputs to the required weight (0, 0.5, or 1). In order to implement the scheme [5], a scan-structure is assumed. Furthermore, an LFSR required to feed the pseudorandom input to the scan inputs is implemented (the number of LFSR stages is n , where n is the number of scan cells), as well as a scan counter, common to all scan schemes. A number of 3-gate modules is required for every required weighted input Wang [8] proposed a low-overhead 3-weight random BIST scheme, again based on scan chains. He proposed two schemes, namely serial fixing BIST and parallel fixing BIST. Serial fixing scheme is shown to be more costly [8]; therefore we shall concentrate our comparisons to the parallel fixing BIST scheme. The hardware overhead of the decoding logic for some of the ISCAS benchmarks is calculated in [8, Table I]. All schemes require the application of the session counter, required to alter among the different weight sessions. Schemes proposed in [5] and [8] are test per scan schemes, and, of course, assume the existence of scan capability of the latches of the design.

IV. CONCLUSION

We have tested the weighted pattern generation of an accumulator which can be utilized to efficiently generate weighted patterns without altering the structure of the adder. Comparisons with a previously proposed accumulator-based 3-weight pattern generation technique [11] indicate that the hardware overhead of the proposed scheme is lower, while at the same time no redesign of the accumulator is imposed, thus resulting in reduction of 20%–95% in test application time. Comparisons with scan based schemes [5], [8] show that the proposed schemes result in lower hardware overhead. Finally, comparisons with the accumulator-based scheme proposed in [22] reveal that the proposed scheme results in significant decrease in hardware overhead. The main goal for designing it is to maintain speed and area which are successfully achieved. Further verify this Project on Static Timing analysis (STA) and System on chip verification (SOC) for Practical Debugging and layout design.

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my Supervisor as well as our principal who gave me the golden opportunity to do this wonderful project on the topic, which also helped me in doing a lot of Research and I came to know about so many new things I am really thankful to them.

REFERENCES

- [1] R. S. Patti, "3-D integrated circuits and the future of system-on-chip designs," Proc. IEEE, vol. 94, no. 6, pp. 1214–1224, Jun. 2006.
- [2] M. Motoyoshi, "Through-silicon via (TSV)," Proc. IEEE, vol. 97, no. 1, pp. 43–48, Jan. 2009.
- [3] Y. Chen, D. Niu, Y. Xie, and K. Chakrabarty, "Cost-effective integration of three-dimensional (3D) ICs emphasizing testing cost analysis," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, 2010, pp. 471–476.
- [4] E. J. Marinissen, "Testing TSV-based three-dimensional stacked ICs," in Proc. IEEE Design, Autom., Test Eur. Conf. Exhib., 2010, pp. 1689–1694.
- [5] H. H. S. Lee and K. Chakrabarty, "Test challenges for 3D integrated circuits," IEEE Design Test Comput., vol. 26, no. 5, pp. 26–35, Sep.–Oct. 2009.
- [6] R. Weerasekera, L.-R. Zheng, D. Pamunuwa, and H. Tenhunen, "Extending systems-on-chip to the third dimension: Performance, cost and technological tradeoffs," in Proc. Int. Conf. Comput.-Aided Design, 2007, pp. 212–219.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 3, March 2016

- [7] P. Y. Chen, C. W. Wu, and D. M. Kwai, "On-chip testing of blind and open-sleeve TSVs for 3D IC before bonding," in Proc. IEEE VLSI Test Symp., 2010, pp. 263–268.
- [8] Y. Lou, Z. Yan, F. Zhang, and P. D. Franzon, "Comparing Through-Silicon-Via (TSV) void/pinhole defect self-test methods," J. Electron. Testing, vol. 28, no. 1, pp. 27–38, 2011.
- [9] M. Cho, C. Liu, D. H. Kim, S. K. Lim, and S. Mukhopadhyay, "Pre-bond and post-bond test and signal recovery structure to characterize and repair TSV defect induced signal degradation in 3D system," IEEE Trans. Compon., Packag., Manuf. Technol., vol. 1, no. 11, pp. 1718–1727, Nov. 2011.
- [10] Y. Fkih, P. Vivet, B. Rouzeyre, M.-L. Flottes, and G. Di Natale, "A 3D IC BIST for pre-bond test of TSVs using ring oscillators," in Proc. IEEE Int. New Circuits Syst. Conf., 2013, pp. 1–4.
- [11] S. Deutsch and K. Chakrabarty, "Non-invasive pre-bond TSV testing using ring oscillators and multiple voltage levels," in Proc. Design, Autom., Test Europe Conf. Exhib., 2013, pp. 1065–1070.
- [12] S. Sunter, "Essential principles for practical analog BIST," EDN Netw. Nov. 2010 [Online]. Available: <http://edn.com/design/analog/4363796/Essential-principles-for-practical-analog-BIST-tem-2>
- [13] C. Wang, J. Zhou, B. Zhao, X. Liu, R. Philippe, and M. Je, "Self-test methodology and structures for pre-bond TSV testing in 3D-IC system," in Proc. Asian Solid State Circuit Conf., Nov. 2012, pp. 393–396.
- [14] G. Katti, A. Mercha, M. Stucchi, Z. Tokei, D. Velenis, J. Van Olmen, C. Huyghebaert, A. Jourdain, M. Rakowski, I. Debusschere, P. Soussan, H. Oprins, W. Dehaene, K. DeMeyer, Y. Travaly, E. Beyne, S. Biesemans, and B. Swinnen, "Temperature dependent electrical characteristics of Through-Si-Via (TSV) interconnections," in Proc. Int. Interconnect Tech. Conf., 2010, pp. 1–3.
- [15] M. Maymandi-Nejad and M. Sachdev, "A monotonic digitally controlled delay element," IEEE J. Solid-State Circuits, vol. 40, no. 11, pp. 2212–2219, Nov. 2005.