An Efficient BIST Scheme for Non-Restoring Array Dividers

H. T. Vergos
Computer Engineering and Informatics Dept.,
University of Patras, 26500 Patras, Greece.
vergos@ceid.upatras.gr

Abstract

A new effective built-in self-test (BIST) scheme for non-restoring array dividers (NADs) is proposed, that offers more than 99% cell fault coverage in all NADs of practical use. Moreover, it can be implemented in small hardware and can apply all its test vectors within 128 clock cycles. A version of the proposed scheme for power critical applications is also exploited.

1 Introduction

Several techniques have been developed for enhancing the testability of a design through design for testability (DFT) modifications and for improving the test generation and application process. Among them, built-in self-test (BIST) structures are gaining more interest because they can apply the required test vectors at the same speed as the chip operates, they can cut down the cost of using external testing equipment, and because parts of a BIST structure may be shared among several design modules [1]. For a BIST structure to be effective, it must satisfy the following criteria : (a) avoidance of performance degradation imposed by DFT modifications of the original structure, (b) small area implementation, (c) high fault coverage under a realistic fault model, (d) small test application time and (e) low power dissipation during testing [2, 3]. The latter requirement has recently evolved mainly by the diffusion of battery-powered portable devices, where average power consumption is a critical design concern.

Array dividers are commonly included in general or special purpose microprocessors / microcontrollers, commonly found in today's handheld devices. Three different architectures have been proposed for implementing a divider in hardware, namely the carry - free [4], the restoring and the non-restoring [5] architectures. Testing of these architectures has also been considered in the past. For carry-free dividers C-testable designs were presented in [6, 7], and a BIST structure in [8]. Restoring array dividers can be tested linearly if the method of [9] is followed, no matter what the cell used for their implementation is. Moreover, if a certain cell is used C-testability can be achieved [9]. For achieving the C-testability the authors of [9] were forced to add one control input as well as a linear to the size of the divider number of XOR gates on the critical path. For the common case of a 16 × 16 NAD, this leads to about 3% hardware and delay overhead. However, a BIST for either restoring or non-restoring array dividers has not been presented. A BIST scheme can be of course devised if one takes into account the C-testable design presented in [9] and constructs a test pattern generation circuit (TPG) that produces the required vectors. This solution though can not be regarded efficient considering that the area and delay overheads imposed by the C-testability design requirements remain, the resulting TPG lacks any regularity and that if the divider is used as a design module in a larger IC, the resulting BIST can only be used for the divider; it can not be shared among other modules.

In this paper a new BIST structure for Non-restoring Array Dividers (NADs) with respect to the cell fault model is presented. The proposed BIST scheme fulfils all the requirements set above including that it does not require any modification of the initial divider design.

2 NADs and the Proposed BIST Scheme

In binary non-restoring division, successively right shifted versions of the divisor are subtracted from or added to the dividend or the resulting partial product. The carry out of the partial remainder determines a quotient bit as well as whether the shifted divisor is to be added or subtracted on the next cycle. In order to cut down the number of cycles required for a full division a NAD is usually implemented as a two-dimensional cellular iterative array. The basic cell used in the construction of the array is the controlled adder / subtracter cell (CAS) cell which is presented in Fig. 1. The FA block in Fig. 1 denotes a full adder.

A $n \times n$ NAD array is constructed from n^2 basic cells, interconnected as presented in Fig. 2, with inputs the dividend $N = (N_1, N_2, \ldots, N_{2n-1})$ and the divisor $D = (D_1, D_2, \ldots, D_n)$. The NAD produces the quotient $Q = (Q_1, Q_2, \ldots, Q_n)$ and the remainder $R = (R_n, R_{n+1}, \ldots, R_{2n-1})$ outputs. The line Q_{i-1} which is

10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007)
0-7695-2978-X/07 $25.00 © 2007 IEEE

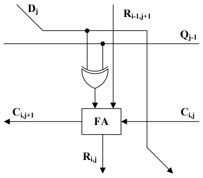

Figure 1. Controlled add / subtract (CAS) Cell

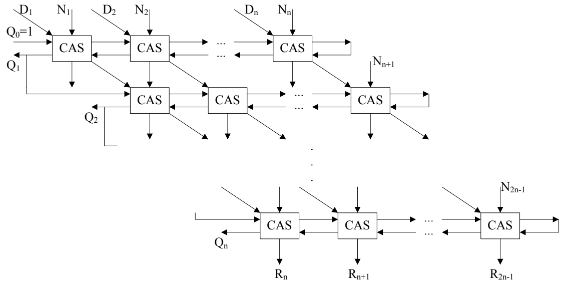

Figure 2. Original Divider Design

the carry or the borrow out of the $i-1$ row of cells, controls the operation that is to be performed by the CAS cells of the ith row. An addition or a subtraction takes place in the ith row if Q_{i-1} is 0 or 1 respectively.

Several simplifications can be made to this original design. Since Q_0 is always set to 1 in order for the first row to always perform a subtraction, the XOR gates of the CAS cells of the first row can be substituted by inverters that reduce both the delay and the area. The Q input of the first row, Q_0, can also be eliminated. Moreover, an accuracy of n bits can be assumed for the remainder, leading to the leftmost cells of each row excluding the last, to only output a quotient bit and not a remainder bit. Finally, the interconnection of Q_{i-1} and $C_{i,j}$ at the rightmost cell of each row leads to several simplifications that also cut down both the area and the delay of the divider. The above simplifications obviously lead to a harder to test design since both the controllability (a primary input was removed) and the observability (the partial remainder outputs were removed) of the divider are reduced. The resulting 4×4 NAD is shown in Fig. 2 where different shadowing schemes are used for representing cells that have been simplified according to the above.

Although the BIST scheme that will be proposed is effective no matter what the specific implementation chosen for the CAS cell is, in order to get realistic results for stuck-at fault coverage and total power consumption during test, specific gate level implementations of the CAS cell need to be considered. In this work three different implementations for the FA of each CAS cell, depicted respectively in Fig. 4 are considered. The resulting CAS cells will hereafter be denoted as Cell 1, Cell 2 and Cell 3 respectively. The top,

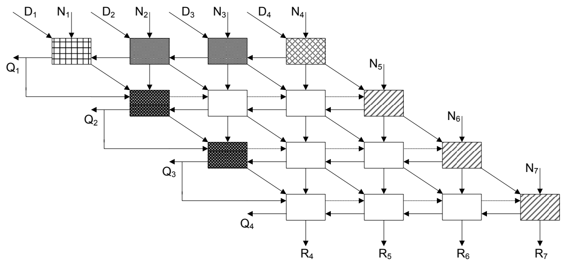

Figure 3. Simplified divider

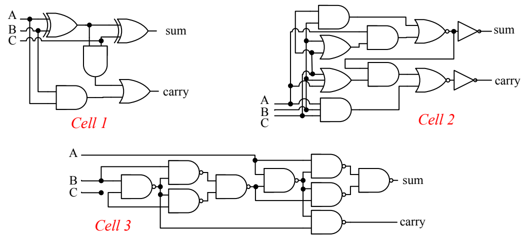

Figure 4. FA implementations of Cell1, Cell2 and Cell3

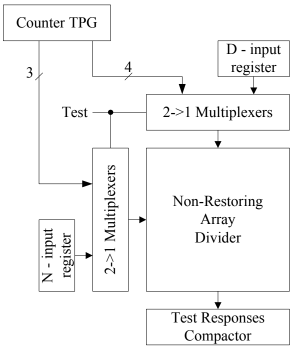

Figure 5. The proposed BIST scheme.

leftmost and rightmost CAS cells are simplified versions of the implementation chosen for the CAS cell.

The proposed BIST scheme is presented in Fig. 5 and is comprised by the input multiplexers, the TPG circuit and the Test Response Compactor circuit. Selection between normal and test mode is accomplished by the input signal Test. When this signal is set to 0 normal division takes place. When this signal is set to 1 the divider inputs receive the TPG outputs and testing mode is selected. The TPG used is a 7-bit counter going through all its 128 possible states. The counter is enabled by signal Test and clocked by the system clock. When in test mode a new value is produced at every cycle. During test mode, the outputs of the counter are applied to the NAD inputs by the following rules : (a) 4 distinct bits of the counter TPG are repetitively used for forming the divisor input D and (b) the rest 3 dis-

0-7695-2978-X/07 $25.00 © 2007 IEEE

Table 1. Proposed BIST Cell Fault Coverage

NAD Size (n)	8	16	32
CFC	99.2%	99.4%	99.1

Table 2. Stuck-at Fault Coverage

NAD Size (n)	Cell 1	Cell 2	Cell 3
8	100%	100%	99.5%
16	100%	100%	99.7%
32	100%	100%	99.8%

Table 3. Stuck-at Fault Coverage with Aliasing

NAD Size (n)	Cell 1	Cell 2	Cell 3
8	99.5%	99.9%	99.4%
16	99.8%	99.9%	99.6%
32	99.8%	99.9%	99.7%

tinct bits of the counter TPG are repetitively used for forming the dividend input N. The proposed test of the divider requires only 128 clock cycles. The $2n$ outputs of the circuit are sampled at the end of each cycle of the test. It is assumed that the circuit responses are captured and compacted by an adder-accumulator implementing the single rotate carry compaction scheme. After all the 128 patterns have been applied the accumulator contains a value (signature) that is compared against a pre-computed correct value. Different values imply a fault in the divider. Identical values imply that either the divider is fault free or an aliasing has occurred during the compaction of the test responses.

3 Effectiveness of the Proposed BIST Scheme

The cell fault coverage and the stuck-at fault coverage for the three distinct implementations of the CAS cell of Fig. 4 are examined in this Section. The aliasing probability is also given when examining stuck-at faults. The power dissipated during test is also examined and a modified TPG is also suggested targeting low power dissipation during test. The area overhead of the proposed BIST scheme is finally discussed. The results that are presented excluding the area measurements were gathered by developing an HDL generator for arithmetic circuits, a test pattern generator, a cell fault simulator, a stuck-at fault simulator implementing a variety of output compactor schemes and a gate level power simulator. The Synopsys suite of tools and a 0.18 μm implementation technology were used for the pre-layout area estimations of the NADs and the required BIST circuits.

When the counter TPG goes through all its 128 cycles almost all CAS cells of the examined NADs receive all possible input combinations. The set of cells that do not receive all possible input combinations is mainly composed of the leftmost cells of all but the first row which do not receive the all 0's input combination and few ($< 15\%$) other cells that follow a random pattern in all examined NAD sizes. Let S_i denote the number of possible input combinations of cell i and R_i the number of distinct input combinations that cell i receives when the proposed counter TPG goes through all its 128 states. The Cell Fault Coverage (CFC) of the NAD denotes the ratio $\frac{\sum_i R_i}{\sum_i S_i}$. Table 1 presents the CFC obtained for the examined NAD sizes. In all cases the CFC is over 99%. Depending on the particular gate level implementation used for the FA, the missing cell input combination may or may not impact the stuck-at fault coverage of the circuit, since there are gate level implementations of the full adder that do not require all the possible combinations for detection of all single stuck-at faults. This implies that CFC values always correspond to at least the same single stuck-at fault value, regardless of the gate level implementation chosen for the FA of the cells. The above analysis along with the results of Table 1 indicate that a stuck-at fault coverage of over 99% is expected, if the aliasing probability is not taken into account. For the three considered implementations of the FA, Table 2 lists the single stuck-at fault coverage that can be obtained by the proposed BIST scheme when aliasing is not taken into account. The percentage of undetected stuck-at faults is extremely small and in all cases less than 0.5%.

The aliasing probability is examined next. This depends heavily on the circuit used for the compaction of the responses. In order to keep the area overhead as low as possible, in the proposed BIST scheme a simple rotate carry adder-accumulator is used as the test response compactor. Table 3 presents the attained stuck - at fault coverage when aliasing is also taken into account. As can be observed in all but one cases the single stuck-at fault coverage is at least 99.5%. In [1] it has been shown that the power consumption during test is higher than that during normal circuit operations. In battery-powered devices this means that battery life depends on the power consumed during the application of power-up or periodic tests (often implemented by BIST approaches). So there is an emerging need for BIST structures which offer low power consumption during test. Three different techniques have been proposed [11] for a counter-based BIST scheme that aim to reduce the power consumed during test, namely, the proper assignment of the TPG outputs to the Circuit Under Test (CUT) inputs, the use of Gray instead of binary counters for reducing the number of transitions at the inputs of the CUT and the reduction of the test set. The application of the first two techniques were investigated for the proposed BIST scheme. Table 4 presents the reduction that can be achieved in the total power consumption when only the first or both the first and the second techniques are used. The power consumption reduction per-

Table 4. Power Consumption Reduction

NAD size	$n = 8$			$n = 16$			$n = 32$		
	Cell 1	Cell 2	Cell 3	Cell 1	Cell 2	Cell 3	Cell 1	Cell 2	Cell 3
Proper Assignment	15.0%	12.8%	9.5%	6.9%	5.6%	4.7%	5.8%	5.3%	5.0%
Proper Assignment + Gray	21.9%	18.6%	16.0%	12.7%	10.8%	9.6%	10.8%	9.6%	9.0%

Table 5. Area Overhead

NAD Size (n)	8	16	32
Proposed BIST	16.7%	6.7%	3.0%
Proposed LP-BIST	18.1%	7.1%	3.2%

centages are computed in Table 4 over a BIST scheme with the worst assignment that does not make use of Gray code. The power consumption caused by the BIST circuitry itself has also been taken into account. Since the employment of a Gray instead of a binary counter requires the addition of some XOR gates the last row of Table 4 corresponds to a TPG circuit that has a larger implementation area than the TPG of the initially introduced BIST. The BIST scheme with a Gray counter TPG will be denoted as the LP-BIST scheme. The area overhead imposed by the proposed or the LP-BIST BIST scheme on the initial divider design is presented in Table 5. The area overhead includes the binary or the Gray counter TPG as well as the input multiplexers. Note that, although pre-layout estimations, the results provided by the Design Analyzer tool of Synopsys also take into account net interconnection area. In the area overhead computations the test response compactor circuit was not taken into account. This assumption was based on the fact that a single rotate carry adder-accumulator was adopted for output compaction. Since a divider is always accompanied in any datapath by an accumulator and an adder, these can be used efficiently during the test of the divider.

The results of Table 5 indicate that the proposed BIST circuit as well as the LP-BIST increase the area of the divider by at most 7% when n=16 or 32. The area overhead percentage would be a lot smaller in a whole chip, since the TPG circuit of either BIST schemes can be used efficiently for testing other structures (for example adders or multipliers). As stated in the introduction, the C-testable design of NAD has an area overhead of 3% when n=16. This percentage would however increase significantly when we add a TPG for producing the vectors extracted in [9] and the required input multiplexers. Moreover the resulting BIST would be a dedicated circuitry for testing the divider which can not be used for testing other parts of a whole chip. Finally, but most important, the BIST structure proposed in this paper does not require the addition of any DFT hardware in the original critical path of the divider. On the other hand, a BIST devised according to the vectors presented in [9] requires the addition of n XOR gates on the critical path of the divider.

4 Conclusions

A new effective BIST scheme for non-restoring array dividers has been proposed in this paper. The proposed scheme has all the desired characteristics of a BIST scheme since it requires small implementation area and achieves more than 99% cell fault coverage by applying only 128 test vectors. Moreover, another version of the proposed scheme was exploited that reduces the total power dissipated during test by 13% on the average and can be used more efficiently in a power critical application.

References

[1] Y. Zorian. A Distributed BIST Control Scheme for Complex VLSI Devices. *IEEE VLSI Test Symposium*, pp. 4–9, 1993.

[2] P. Girard et al. A Test Vector Inhibiting Technique for Low Energy BIST Design. *IEEE VLSI Test Symposium*, pp. 407–412, 1999.

[3] F. Corno and M. S. Reorda. Optimal Vector Selection for Low Power BIST. *IEEE Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 219–226, 1999.

[4] A. Vandemeulebroecke et al. A New Carry-Free Division Algorithm and its Application to a Single-Chip 1024-b RSA Processor *IEEE J. of Solid - State Circuits*, pp. 748–756, 1990.

[5] K. Hwang. Computer Arithmetic : Principles, Architecture and Design. *John Wiley & Sons*, 1979.

[6] H. R. Srinivas et al. A C-testable Carry-Free Divider. *IEEE Conference on Computer Design*, pp. 206–213, 1993.

[7] C. L. Wey. Design and Test Generation of C-testable High-Speed Carry-Free Divider. *IEE Proceedings - Computers and Digital Techniques* pp. 193–200, 1995.

[8] C. L. Wey. Built-In Self-Test (BIST) Design of High-Speed Carry-Free Dividers. *IEEE Transactions on VLSI Systems*, pp.141–145, 1996.

[9] N. K. Jha and A. Ahuja. Easily Testable Nonrestoring and Restoring Gate-Level Cellular Array Dividers. *IEEE Transactions on Computer - Aided Design of Integrated Circuits and Systems*, pp. 114–123, 1993.

[10] J. Rajski and J. Tyszer. Test Response Compaction in Accumulators with Rotate Carry Adders. *IEEE Transactions on Computer - Aided Design of Integrated Circuits and Systems*, pp. 531–539, 1993.

[11] D. Bakalis et al. Low Power Dissipation in BIST Schemes for Modified Booth Multipliers. *IEEE Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 121–129, 1999.