International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 6 (2015) pp. 15305-15312
© Research India Publications
http://www.ripublication.com

Input Vector Monitoring Concurrent BIST Using Compressed Test Pattern

L. Magthelin Therase

Assistant Professor,

Department of ECE, Sathyabama University, Chennai-600119.

ltherase@gmail.com

Abstract

Built In Self Test (BIST) technique constitute an attractive and practical solution to the problem of testing VLSI circuits and system. In this paper propose an Input vector monitoring concurrent BIST using bitmask and Run Length Encoding (RLE) based compression technique. RLE is a method of lossless compression technique. This RLE algorithm uses those runs to compress the original source file while keeping all the non-runs without using for the compression process. This bitmask technique is provides a substantial improvement in the compression efficiency. A larger memory can accommodate more and large applications but increases cost, area, as well as energy requirements. The proposed technique is addressing this problem by reducing the code size of application. It is a major challenge to develop an efficient RLE and bitmask based compression technique that can generate substantial reduction in code size, testing time and memory requirements without affecting the overall performance.

Index terms: Built In Self Test, Compression, Run length encoding

Introduction

A system on chip (SOC) is becoming smaller and denser. Shrinking transistor facilitates the integration of functionality on the chip operating at low supply voltage

although this lowers the silicon chip reliability. Nevertheless it is needs to maintain complete functionality during a lengthy duration, even under converting environments such as temperature instability must be taken as a time varying parameter.

With the current trend to dramatically increase the scale of integration, BIST technique provides an attractive solution to test modules deeply embedded in complex integrated circuits. It is a design for testability (DFT) technique that places the testing function physically with the circuit under test (CUT). It eliminates the necessity of high bandwidth interaction and allows at speed testing. Other benefits of BIST include reduced product development cycle and cost effective system maintenance. The BIST technique is classified based on the operational condition of the circuit under test as online BIST and offline BIST. Online testing occurs during normal functional operating condition. It includes concurrent and non-concurrent. Offline BIST deals with testing a system when it is not carrying out its normal function.

Data compression is the art of representing information in compact form. It reduces the file size which in turn reduces the required storage space and makes the transmission of data faster. Compression techniques try to observe redundant data and eliminate these redundancies. Lossless data compression is a class of data compression algorithms that allows the original data to be perfectly reconstructed from the compressed data.RLE is a very simple form of data compression in which runs of data are stored as a single data count and value, rather than as the real run. This is most important on data that contains many such runs.The compression plays a crucial role, reducing the memory requirements and testing time. It also overcomes Automatic Test Equipment (ATE) bandwidth limitation. Alternatives to outer testing include BIST. However, BIST is not appropriate for logic testing because of its random resistant fault and bus contention during test application, which causes inadequate test coverage.

Existing Method

Input vector monitoring concurrent BIST using SRAM cells illustrates in fig.1. This scheme is based on the idea of monitoring a window of vectors. Every moment, the input vectors belonging to the window are monitored, and if a vector performs a hit, the Response Verifier (RV) is enabled. The bits of the input vector are divided into two distinct sets comprising wand kbits, respectively, such that $w + k = n$.

The k bits of the input vector show whether the input vector belongs to the window under consideration. The remaining bits show the relative location of the incoming vector in the current window. If the incoming vector depend on the current window and has not been received during the examination of the prevalent window, we say that the vector has performed a hit and the RV is clocked to capture the CUT response to the vector.

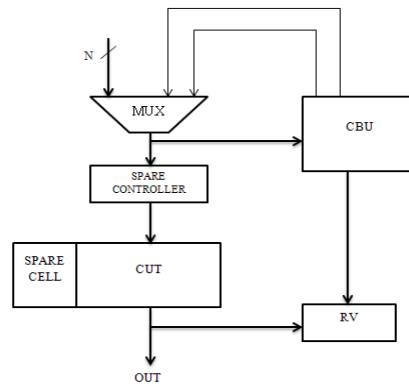

Figure 1: Input vector monitoring concurrent BIST using SRAM cells

In fig.2 illustrates the architecture of input vector monitoring concurrent BIST using SRAM cell. When all the vectors that belong to depend on the current window have given the CUT inputs, it proceeds to examine the next window. It operates in normal and test modes, depending on the value of the signal T/N. When T/N= 0the normal input vector are drive the inputs to the CUT.

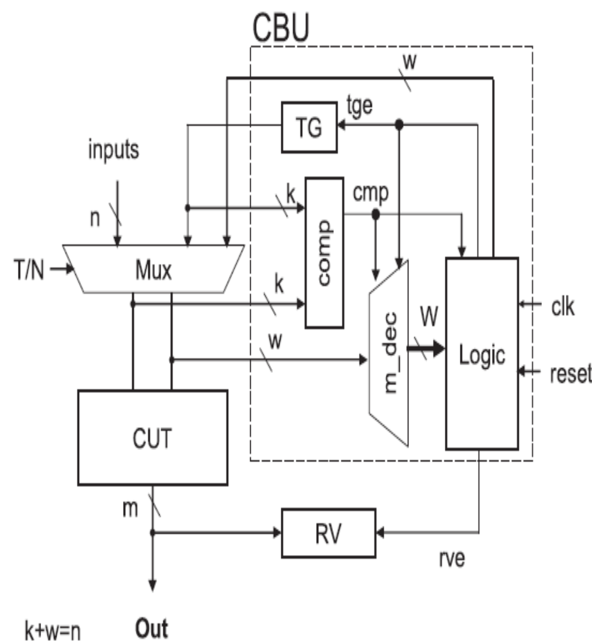

Figure 2: Architecture of Input vector monitoring concurrent BIST using SRAM cell

The inputs of the CUT are also driven to the Concurrent BIST Unit (CBU) as follows: the high order bits are driven to the inputs of a k-stage comparator; the outputs of a k stage test generator are driven the other inputs of the comparator. When test generator (TG) enable is enabled, all outputs of the decoder are equal to one. When comparator is disabled all outputs are disabled. When test generator is disabled and comparator is enabled, the module operates as an ordinary decoding structure.It comprises Wcells, a sense amplifier, two *D* flip flops, and a w-stage counter. The overflow signal of the counter drives the test generator signal through a unit flip-flop delay. In the sequel, we have assumed a clock that is active during the second half of the period. In the sequel, we describe the operation of the logic module, presenting the following cases: 1) reset of the module 2) hit of a vector 3) a vector that belongs in the current window reaches the CUT inputs but not for the first time; and 4) test generation operation.

Proposed Method

The proposed technique is input vector monitoring using bitmask and RLE based compression. A larger memory can accommodate more and large application but increases cost, area as well as energy requirements. The bit mask and RLE based compression technique are used to reduce the test pattern, testing time and memory requirements. In fig.3 Input vector monitoring concurrent BIST using bitmasks and RLE based compression method.

Bitmask based compression of test data may seem attractive, but it presents different challenges.RLE is a method of lossless compression technique.This RLE algorithm uses those runs to compress the original source file while keeping all the non runs without using for the compression process. It has to determine not only the effective bitmasks, butalso a bankable dictionary that produces optimalresults.These compression algorithms are first given the test data. Hereinafter it is divide the test data based on scan chains. Then select the bitmasks and RLE. Next perform dictionary selection. Finally compress using selected bitmasks, RLE and dictionary.

BIST techniques are kindly classified into offline and online. In the proposed method operates in one out of two modes, normal, and test, depending on the value of the signal T/N (Test or normal). When T/N = 0 (normal mode) the inputs to the CUT are driven by the normal input vector. The inputs of the CUT are also driven to the compressionunit.

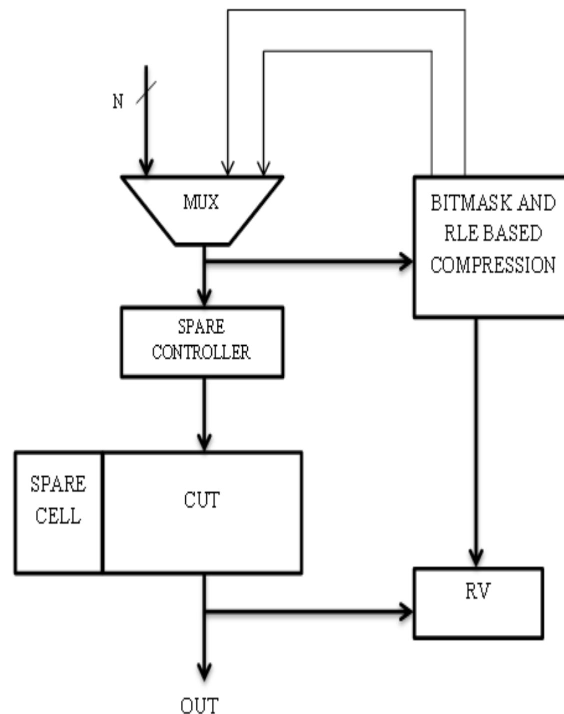

Figure 3: Input vector monitoring concurrent BIST using bitmasks and RLE based compression method

This architecture test the CUT simultaneously with its normal operation by using input vectors appearing to the inputs of the CUT; if the incoming vector belongs to a set called active test set; the Response Verifier is enabled to capture the CUT response. The CUT has n inputs and moutputs and is tested exhaustively; hence, the test set size is N = 2n. The technique can operate in either normal or test mode, belongs to the value of the signal labeled T/N.

Compression Techniques

The bitmasks and RLE based compression techniques are provide efficient code reduction method. In fig.4 illustrates the bitmask and RLE based compression. The first step divides the uncompressed data set to equal length slices for compression. Next, it tries to determine the profitable bitmasks, RLE and dictionary using the procedures. The dictionary selection algorithm generates the dictionary entries while the bitmask selection algorithm provides the number and type of profitable bitmasks. Finally, the test compression is performed using the generated dictionary and bitmasks.

Bitmask based compression

Bitmask based compression is an enhancement on the dictionary based compression method that helps us to get more matching patterns. In dictionary based compression, each vector is compressed only if it totally matches with a dictionary entry. The compressed data is denoted as follows. Those vectors that match directly are compressed with three bits. The first bit denotes whether it is compressed (using 0) or not (using 1). The second bit represents whether it is compressed using bitmask (using 0) or not (using 1). Finally the last bit represents the dictionary index. Data that are compressed using bitmask needs seven bits. The first two bits, as before, indicate if the data is compressed, and whether the data is compressed using bitmasks.

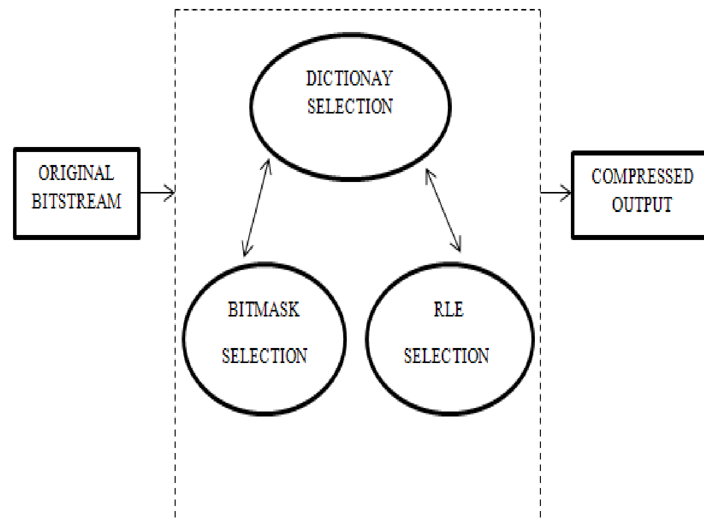

Figure 4: Bitmask and RLE Based Compression

RLE based compression

RLE is the simplest of the data compression algorithms. It replaces runs of two or more of the same character with a number which indicates the length of the run, followed by the real character. Only onecharacteris coded as runs of 1. The important task of this algorithm is to recognize the runs of the source file, and also to record the symbol and the length of each run. The RLE algorithm uses those runs to compress the real source file while keeping all the non-runs without using for the compression process.

Discussion

A larger memory can accommodate more and large applications but increases cost, area, as well as energy requirements.The proposed input vector monitoring concurrent BIST usingefficient RLE and bitmask based compression technique that can generate substantial reduction in code size, testing time and memory requirements than the existing method. And also using this compression technique, input data is compressed and it is used for further proceedings. This reduces the low hardware overhead and power consumption.

Conclusion

BIST method constitutes an attractive solution to the problem of testing VLSI devices. Input vector monitoring concurrent BIST methods perform testing during the circuit normal operation without striking a necessary to set the circuit offline to perform the test; therefore they can circumvent problems appearing in offline BIST techniques. Inthe proposed input vector monitoring concurrent BIST using efficient RLE and bitmask based compression technique that can generate substantial reduction in code size, testing time and memory requirements. And also using this compression technique, input data is compressed and it is used for further proceedings. This reduces the low hardware overhead and power consumption.

Reference

[1] Ioannis Voyiatzis and Costas Efstathiou ” Input Vector Monitoring Concurrent BIST Architecture Using SRAM Cells” IEEE transactions on very large scale integration (VLSI) systems, vol. 22, no. 7, july 2014.

[2] I. Voyiatzis, T. Haniotakis, C. Efstathiou, and H. Antonopoulou, “A concurrent BIST architecture based on monitoring square windows,” in Proc.5th Int. Conf. DTIS, Mar. 2010, pp. 1–6.

[3] Kanad Basu, Student Member, IEEE, and Prabhat Mishra,”Test Data Compression Using Efficient Bitmask and Dictionary Selection Methods” IEEE transactions on very large scale integration (VLSI) systems, vol. 18, no. 9, september 2010.

[4] M. A. Kochte, C. Zoellin, and H.-J.Wunderlich, “Concurrent self-test with partially specified patterns for low test latency and overhead,”in Proc. 14th Eur. Test Symp., May 2009, pp. 53–58.

[5] S. Almukhaizim and Y. Makris, “Concurrent error detection methods for asynchronous burst mode machines,” IEEE Trans. Comput., vol. 56, no. 6, pp. 785–798, Jun. 2007

[6] S. Almukhaizim, P. Drineas, and Y. Makris, “Entropy-driven parity tree selection for low-cost concurrent error detection,” IEEE Trans. Comput.Aided Design Integr. Circuits Syst., vol. 25, no. 8, pp. 1547–1554, Aug. 2006.

[7] I. Voyiatzis, A. Paschalis, D. Gizopoulos, N. Kranitis, and C. Halatsis, “A concurrent BIST architecture based on a selftestingRAM,” IEEE Trans. Rel., vol. 54, no. 1, pp. 69–78, Mar. 2005.

[8] I. Voyiatzis and C. Halatsis, “A low-cost concurrent BIST scheme for increased dependability,” IEEE Trans. Dependable Secure Comput., vol. 2, no. 2, pp. 150–156, Apr. 2005.

[9] K. K. Saluja, R. Sharma, and C. R. Kime, “A concurrent testing technique for digital circuits,” IEEE Trans. Comput. AidedDesign Integr. Circuits Syst., vol. 7, no. 12, pp. 1250–1260, Dec. 1988.

[10] R. Sharma and K. K. Saluja, “Theory, analysis and implementation of an on-line BIST technique,” VLSI Design, vol. 1, no. 1, pp. 9–22, 1993.