



Test Minimization for Full adder Using Integer Linear Programming (ILP) and Design for Testability (DFT) Tools

¹J.L.V. Ramana Kumari, ²M. Asha Rani, ³N. Balaji

¹Department of ECE, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India

²Department of ECE, Jawaharlal Nehru Technological University, Hyderabad, India

³Department of ECE, Jawaharlal Nehru Technological University, Vizianagaram, India

Abstract:-Design For Test is a method to merge design and testing process which helps in improving the test coverage and test minimization and thereby reducing the testing cost for complex digital designs. This paper presents an effective method to enhance the test coverage of various complex combinational circuits. This work deals with two phases. In the first phase, Scan insertion is done. In the second phase, compact test patterns are generated using ATPG and test coverage is done using Fast Scan technique. The test patterns are generated in chain, serial and parallel formats. There are many optimization techniques used to minimize the test patterns. Among them Zero One Linear programming (ZOLP) is widely used. It uses Duality theory for fault set identification and test minimization. In this paper, these two methodologies are used and compared for their effective test coverage and CPU processing time.

Keywords:-ATPG, DFT, FAST SCAN, Test Coverage, ZOLP.

I. INTRODUCTION

Deviation in component from its intended function is called a fault. To detect these faults test vectors/patterns are required. Test vectors are applied by means of primary inputs and responses are observed at primary outputs. Testing is an experiment in which a system is exercised and the resultant behavior is examined to make sure that the system behaves correctly.

Response of vectors (patterns) from a good circuit is compared with the response of vectors (using same patterns) from a DUT (Design Under Test). If the response is same or matches, circuit is good, otherwise faulty. Such testing of complex ICs consumes more time and hence expensive. The time required to test a device needs to be decreased in order to decrease the cost of testing. For effective test coverage, sets of test vectors are applied to the circuits by a tester. Digital systems are built with more and more complexity, the fault testing and diagnosis of digital circuits becomes an important and indispensable part of the manufacturing process.

Test optimization is an essential requirement for efficient testing for which Integer linear programming (ILP) is an effective mathematical method. It is used for test minimization and has been used for both

combinational and sequential circuits [1,2]. Effective tools for Automatic Test Generation are needed to obtain compact test sets with high defect coverage. Once a chip is fabricated, it must be tested to ensure that it functions as desired. The set of tests applied should be able to cover all possible faults that could occur in manufacturing process. Obtaining a set of tests that will detect most possible defects is a difficult problem, and this is the objective of an Automatic Test Pattern Generator. The approach commonly used is to generate a test set that detects all single stuck-at faults in the circuit. After high fault coverage for single stuck-at fault is achieved, additional test vectors may be generated that target other fault models, such as the delay fault model.

Quality demand increase and cost reduction makes Integrated Circuit manufacturers constantly attempt to improve their test methods in order to detect the largest amount of possible fabrication defects in the shortest period of time. Scan is a well known technique that can detect most of the faults of a digital circuit using patterns generated by an ATPG tool. With this technique, it is possible to obtain high values of coverage on pure digital circuits efficiently.

Test generation is the process of determining the stimuli necessary to test a digital system. The task of finding an optimum number of test vectors becomes complex when number of circuit inputs becomes large. In such cases, automatic test vector generation is generally used. Test vector generation can also include design verification, testing, design debug or characterization testing. Time and efficiency are of utmost importance during design and test development.

Design for Test (DFT) is a name for design techniques that add certain testability features to a microelectronic hardware product design. DFT plays an important role in the development of test programs and as an interface for test application and diagnostics. ATPG is much easier if appropriate DFT rules and suggestions have been implemented. DFT often is associated with design modifications that provide improved access to internal circuit elements such that the local internal state can be controlled (controllability) and/or observed

(observability) more easily. Structural test makes no direct attempt to determine the functional correctness of the entire circuit. Instead, it tries to make sure that the circuit has been assembled correctly from some low-level building blocks as specified in a structural netlist.

The most common method for delivering test data from chip inputs to internal Circuits Under Test (CUTs), and observing their outputs, is called Scan-Design. In Scan-Design, registers, flip-flops and latches in the design are connected in one or more scan chains, which are used to gain access to internal nodes of the chip. Test patterns are shifted in via the scan chain(s) and the results are then shifted out to chip output pins and compared against the expected "good machine" results. The output of a scan design may be provided in forms such as Serial Vector Format (SVF), to be executed by test equipment. In addition to being useful for manufacturing, testing, scan chains can also be used to "debug" chip designs.

II. IMPLEMENTATION

2.1 Methodology 1

The main focus of this proposed work is to minimize the number of tests performed to find faults in combinational circuits. A new technique is used which comprises of three phases. The first phase identifies independent faults. The second phase generates test vectors for faults identified in first phase. The third phase minimizes the number of test vectors generated in second phase. Zero One Linear Programming (ZOLP) technique is employed in third phase to minimize the number of test vectors. This problem is formulated using duality theory of linear programming [8]. In Dual ILP, independent fault set identification is done and test minimization is done in primal ILP. Dual problem solution is Conditionally Independent Fault Set (CIFS). Primal solution is minimized set of test vectors.

These ILP formulations use a fault detection table which contains information about faults detected by each vector. The ILP formulation for minimizing test sets used for full-response dictionary based diagnosis requires a matrix representation that not only tells which tests detect which faults, but also at which outputs the discrepancies were observed for each fault-test pair. For this reason we define a new fault diagnostic table.

Primal-Dual method is applied after obtaining the diagnostic matrix. Stuck-at faults are only taken into consideration. The redundant faults are removed from the fault list to obtain the final fault list. We used integers to represent the output response for each test vector. As faults detected by different test vectors are already distinguished, there is no need to compare the corresponding output responses. Hence we assigned indices for the failing output responses for each test vector.

2.1.1 BASIC IMPLEMENTATION OF TEST MINIMIZATION ALGORITHM FOR FULL ADDER

The three inputs a, b, c are applied to the Full adder test generation circuit to generate the sequences t1,t2, t3,t4,t5,t6 and t7 as shown in figure 1, which are applied as inputs to ZOLP for test minimization to generate the minimized test vectors tv[1:7].

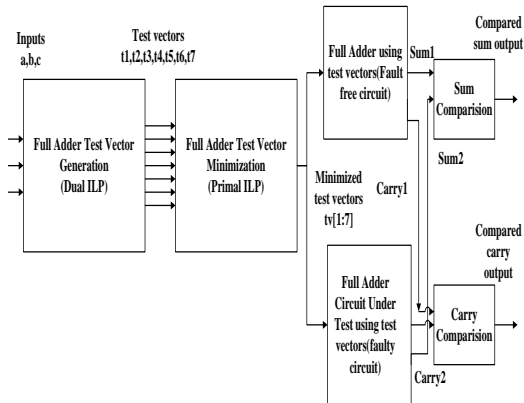


Figure 1:Block diagram of full adder test minimization using ZOLP

These minimized test vectors are applied to both the Full adder Circuit Under Test (CUT) which is obtained by fault insertion method and Fault free circuit. The sum and carry outputs of both the circuits are compared and the compared sum and carry outputs are logic high if there is a fault in the circuit and is logic low if it is fault free.

2.1.2 Fault Insertion

Fault testing and test vector generation requires the insertion of faults into a copy of the circuit and comparison of the outputs with the working version

of the circuit. At every checkpoint, a multiplexer is inserted as shown in figure 2, with one input connecting to the original gate connection and the second input connecting to the stuck-at-1 or 0 fault being simulated.

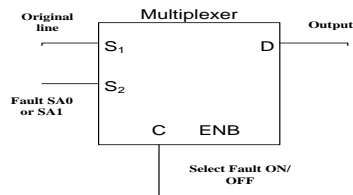


Figure 2: Fault Insertion Multiplexer

2.2 Methodology 2

Testability is one of the most important dependability factors that are considered during design cycle along with reliability, speed, power consumption, cost and other factors. Especially in the area of multi-criteria design-space exploration and optimization methods, testability is required to offer information about easiness of testing a given design both with minimal error and in

reasonable time by using Design For Testability techniques (DFT). One of the most popular DFT technique is the Scan technique. This structural testing is based on inserting registers into one or more scan chains, each of them forming serial path from primary input to primary output of the system. To maximize testability, minimal set of registers must be selected into the scan chains. Increasing size and complexity of digital designs has made the manufacturing process more complicated and enforces more intricacy in verification of designs. This makes it essential to address critical verification issues at the early stages of design cycle. Such a complicated design needs to be tested for fabrication faults as well as functional faults. Several attempts have been made to raise the quality of testing methods with automatic test pattern generation (ATPG) and Design For Testability (DFT) methods in logic and lower levels.

There are two inputs for scan insertion, one is library of components and second is a net list. Both the inputs are stored in separate text files. Library of components can be provided by component vendor or can be created or modified by a user. It is important to know that the accuracy of the results is significantly affected by the accuracy of information stored in the library. Netlist is utilized to describe inter-connections between interfaces of particular component instances involved in the design. The netlist can be created manually by a user or it can be easily generated from VHDL, Verilog or EDIF file utilized by an EDA tool.

Controllability, observability, accessibility of signals from primary inputs and primary outputs are analyzed. These outputs from scan insertion i.e., test procedure files and do files are applied as inputs to ATPG. Fast scan and test patterns are generated in serial, parallel and chain formats. The test coverage, ATPG effectiveness along with CPU process time and number of faults both detectable and undetectable are given as output.

The Behavioral Simulation result for the Full Adder test minimization using integer linear programming is shown in Figure 3. For the inputs a, b, c, h1, h2, h3, h4, h5, h6, h7, h8, h9, h10, h11, h12, h13, h14, h15 the outputs sum and carry are generated.

The Behavioral Simulation result for the Full Adder using integer linear programming is shown in Figure 4. For the inputs a, b, c, h1, h2, h3, h4, h5, h6, h7, h8, h9, h10, h11, h12, h13, h14, h15, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10, c11, c12, c13, c14, h15 the compared outputs sum and carry are generated.

III. RESULTS

3.1 Test minimization using Integer Linear Programming

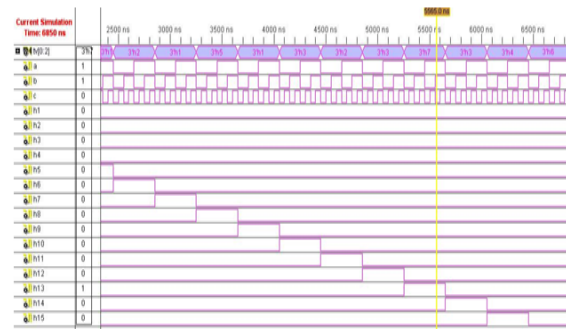


Figure 3: Simulation result of Full Adder test minimization using Integer Linear Programming

3.2 Full adder ILP application

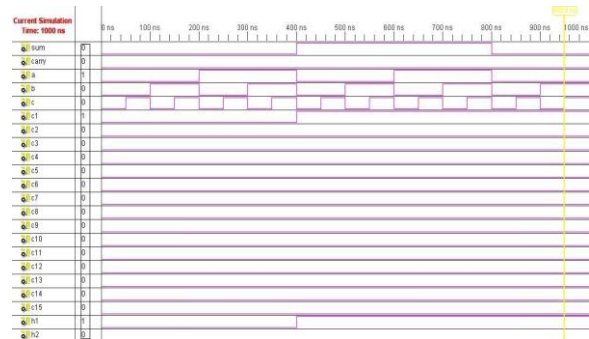


Figure 4: Simulation result of Full Adder using integer linear programming

3.3 Full adder test coverage using DFT tool

The DFT reports are generated for Test Coverage and process CPU time for Full Adder, Parallel Adder and patterns are generated in serial, parallel and chain formats.

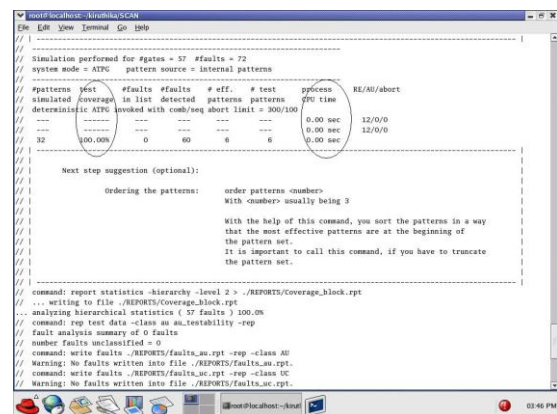


Figure 5: DFT reports for Test Coverage and CPU process time for Full Adder

Table 1: Comparison results for test minimization using ZOLP and DFT

For full adder	Using ZOLP	DFT
Test coverage (%)	87.5	100
Process CPU time (sec)	10.34	0.00

IV. CONCLUSIONS AND FUTURE SCOPE

The test minimization for combinational circuits like full adder has been done using ZOLP algorithm and simulated using Modelsim Simulator. Synthesis was done with Xilinx ISE tool. Successful Simulation and Synthesis reports have been obtained.

Effective test coverage is obtained by generating the compact test sets using the DFT mentor graphics tools. Scan insertion was done using the DFT ADVISOR and the output test procedure files, **do files** are taken as input to ATPG tool. These inputs were applied to Fast Scan to generate chain, serial and parallel test patterns.

The compact test patterns are generated by DFT and ATPG tools which have effective test coverage when

compared to other optimization techniques like Integer Linear Programming and patterns are generated in very less CPU time.

A further scope of the work can be carried out to generate compact test patterns for sequential circuits

and memory BIST using DFT tool to improve the test coverage and thereby reducing the testing time.



REFERENCES

- [1] P. Drineas and Y.Makris, "Independent Test Sequence Compaction through Integer Programming", in Proc International Conf. Computer design, 2003, pp 380-386.
- [2] N. Yogi and V.D Agarwal, "High-Level Test Generation for Gate-Level Fault Coverage", in Proc. IEEE North Atlantic Test Workshop, May 2006, pp 65-74.
- [3] K.T. Cheng, S. Dey, M. Rodgers, and K.Roy, "Test Challenges for Deep Sub-micron Technologies," in Proc. 37th Design Automation Conf., June 2000, pp. 142-149.
- [4] D.H Lee and S.M Reddy, "A New Test Generation Method for Sequential Circuits", Proc.Int.Conf. Computer Aided Design, pp. 446-449, Nov 1991.
- [5] M. L. Bushnell and V. D. Agarwal, "Essentials of Electronic Testing for Digital, Memory, and Mixed-Signal VLSI Circuits", Springer, 2000.
- [6] SM.Thamarai, K.Kuppusamy, T.Meyyappan, "A New Algorithm for Fault based Test Minimization in Combinational Circuits", International Journal of Computing and Applications, ISSN 0973- 5704, Vol.5, No.1, pp.77-88, June 2010.
- [7] J.Kuppusamy, T.Meyyappan , SM.Thamarai "Fault Based Test Minimization for Adder and Subtractor Circuits" International conference on Computer, Communication & Electrical Technology – ICCET2011, 18th & 19th March 2011