RESEARCH ARTICLE OPEN ACCESS

Detection of Stuck at Fault Indigital Circuits at Register Transfer Logic (RTL)

Ms. Nainaa. Udgedr, S. A. Ladhake, Prof. P. D. Gawande
Assistant Professor Principle Faculty of Electr. &Telecom SipnaCOET, Amravati (M.H.)
Assosicate Professor Sipna COET, Amravati (M.H.)
Faculty of Electr. & Telecom. Sipna COET Amravati (M.H.)

Abstract
Due to the increasing complexity of modern circuit design, verification has become the major bottleneck of the entire design process. Most efforts are to verify the correctness of the initial Register-Transfer Level (RTL) descriptions written in Hardware Description Language (HDL).Major drawback of high level design methodologies such as RTL can be seen in the following facts. First, they lack of sufficiently precise fault models - compared to sophisticated models available for low level description levels such as logic gate level. Second, since the structure of a design changes significantly with every logic synthesis run, testability analysis is typically performed only after final logic synthesis.So in this paper, we detect the stuck-at fault using the concept of textio.

Keywords-stuck-atfaults; fault coverage ;testpoints; validation sets

I. Introduction

Sinceintegrated circuit designs are accordinglybecoming more and more complex. As a result of this, VLSI testing has becomeexpensive in terms of cost. Existinggatelevelfault simulation techniques exhibitpoor performance standards whenapplied to such designs and are unsuitable for earlytestabilityanalysis or fault simulations. Also test generation and fault simulation efforts in the post synthesis phase do not contribute to the improvement in the design. Therefore a design methodology for fault simulation athigherlevels of abstraction ishighlydesired.

Test patterns for large VLSI systems are oftendeterminedfrom the knowledge of the circuit function. A fault simulator isthenused to find the effectiveness of the test patterns in detectinggate-level “stuck-at” faults. Existinggate-levelfault simulation techniques sufferprohibitivelyexpensive performance penalties whenapplied to the modern VLSI systems of larger sizes. Also, findings of such test generation and fault simulation efforts in the post logic-synthesis phase are toolate in the design cycle to beuseful for design-for-test (DFT) relatedimprovements in the architecture. Therefore, an effective register-transferlevel (RTL) fault model ishighlydesirable.

Basicallyfaultmodellingconsists of three fundamentals terms i.e. defect, fault & error. Defect - A defect is the unintended difference between the implemented hardware and its intended design Defects occur either during manufacture or during the use of devices the use of devices Fault - A representation of a defectat the abstracted function level Error - A wrong output signal produced by a defective system. An error is caused by a Faultor a design error The problems of ideal tests are : Ideal tests detect all defects produced in the manufacturing process manufacturing process. Also,Ideal tests pass all functionally good devices. „Very large numbers and varieties of possible defectsneed to betested.Difficult to generate tests for some real defects. †

The problems of Real tests are ; Real tests „Based on analyzablefaultmodels, whichmay not map on real defects,Incompletecoverage of modeledfaults due to high complexity,Some good chips are rejected. The fraction (or percentage) of such chips is called the yield loss „Some bad chips pass tests. The fraction (or percentage) of bad chips among all passing chips iscalled the defectlevel.

So in thispaperwe are emphasising on the detection of stuck-atfaults in RTL circuits using the feature of vhdllangauage i.e. textio, withoutcompromising in faultcoverage.

II. Related work

Various methods are implemented to detect the stuck-at fault in digital circuits.

A. Adding Buffer to each ports of RTL Circuits.

The very first method is adding buffer to each ports of RTL circuits to create a new faulty circuit[7].

1. Firstlytestbench is developed and the simulation is first run on a good circuit and then on each of the faulty circuits using any simulator.
2. The outputs obtained in each case of the faulty circuits are compared with the output of the good circuit to determine which faults are detected. That is the new faulty circuit and the fault free circuit is simulated and the outputs so obtained are compared. The fault list is tabulated.
3. The ratio of the numbers of RTL faults detected to the total number of RTL faults gives the RTL fault coverage.

B. Using Validation Test Sets.

Next method is that in which validation test sets are used to generate test sequences that detect a majority of stuck-at faults in the datapath[1].

1. The scheme first derives the controller behaviors from validation test sequences and reuses them for simplifying justification/propagation analysis corresponding to precomputedtest vectors/responses of datapath RTL modules.
2. A heuristic is used to identify controller behaviors that are compatible with a given set of precomputed test vectors/responses. It requires only a single pass through the CDFG corresponding to a validation test sequence and is accu- rate, resulting in a small number of test generation runs.
3. Test generation is performed at the RTL and the controller behavior is prespecified,which results in very small ltest generationtimes.First step is identification of compatible controller behaviors consisting of Augmented Fault Simulation to Derive Activation-Detection Time Frame Pair and Analysis of Requirements to Identify Compatible Faults. Next step is SAT-based RTL ATPG is used to obtain a test sequence that reuses the controller behavior to justify and propagate the precomputed test vector and response to primary inputs and outputs, respectively.

SAT- based RTL ATPG uses an ILA model of the circuit under test.The circuit is unrolled for a predetermined number of cycles determined by the number of vectors in the validation test sequence from which the controller behavior is extracted. Test generation is performed on the entire circuit description comprising the controller and datapath by first identifying the paths from the inputs of the module under test to primary inputs and from the output of the module under test to primary outputs in the ILA model. These paths are then translated into Boolean clauses by translating the functionality of the individual modules in these paths. Once the Boolean clauses that capture the RTL test generation problem and the controller behavior are generated, a SAT solver is invoked to resolve these clauses. If the solver returns with a satisfiable solution, then a test sequence can be extracted from the Boolean variable assignments corresponding to the primary inputs. This sequence is guaranteed to deliver the precomputed test vector to the inputs of the corresponding RTL module and propagate the fault effect from the output of the module to a primary output. If the Boolean clauses are not satisfiable, then test generation fails, indicating that the targeted precomputed test vector/response cannot be justified/propagated by reusing the controller behavior that was found to be compatible by using the heuristic.

C. Implementation of Automatic Test Paterrn Generation.

Next method for detection of stuck-at fault consisting of an algorithm for generating test patterns automatically from functional register-transfer level (RTL) circuits that target detection of stuck-at faults in the circuit at the logic level.

1. In order to do this, a data structure named assignment decision diagram are used[5]. The algorithm is very versatile and can tackle almost any type of single- clock design, although performance varies according to the design style. The first step is to convert each process and concurrent RTL statements in each leaf component of the circuit into ADDs.
2. After that, each ADD is selected and its I nternals targeted for testing. First the arithmetic operations are targeted, then logic arrays, then untagged registers, latches, and memories, then untagged ADNs, and finally random logic blocks and black boxes.

During testing of each RTL element, a nine-valued symbolic RTL justification and propagation is done to trace outpaths from the PIs to the element inputs and element outputs to POs to obtain a symbolic test environment for the module. The search is a branch and bound type of search with backtracking and has a backtrack limit and search time limit that may be adjusted. It requires hierarchy traversal in case of a hierarchical design.

III. RTL Fault Model, Fault-Injection and Fault Simulation

Hardware description languages (HDLs) are used to model VLSI circuits. HDL constructs are classified into three types: structural, register-transfer level (RTL) and behavioral. RTL constructs represent a subset of HDL constructs with the corresponding design guidelines meant to ensure the consistent synthesis of gate-level netlists by logic synthesis tools. With event scheduling and resource allocation information built-in, an RTL model represents the micro-architecture of a circuit.

Properties of the RTL fault model:

- Language operators (which map onto Boolean components in the gate-level netlist) are assumed to be fault-free.
- Variables (which map onto signal lines in the gate-level netlist) contain faults:
- a stuck-at-zero (s-a-0) fault when the logic level is fixed at value 0
- a stuck-at-one (s-a-1) fault when the logilevel is fixed at value 1
- The proposed RTL fault model follows the single fault assumption and therefore only one fault is applied at a time when a test set is evaluated.
- The RTL fault-list of a module contains input as well as fan-out faults. RTL variables that are used more than once in executable statements or the instantiations of lower level modules of the design-hierarchy are considered to have fan-out. Input faults of a module at the RT level have one-to-one equivalence to input faults of the module at the gate-level. The fan-out faults of variables inside a module at the RT level represent a subset of the fan-out faults of a possible gate-level implementation.

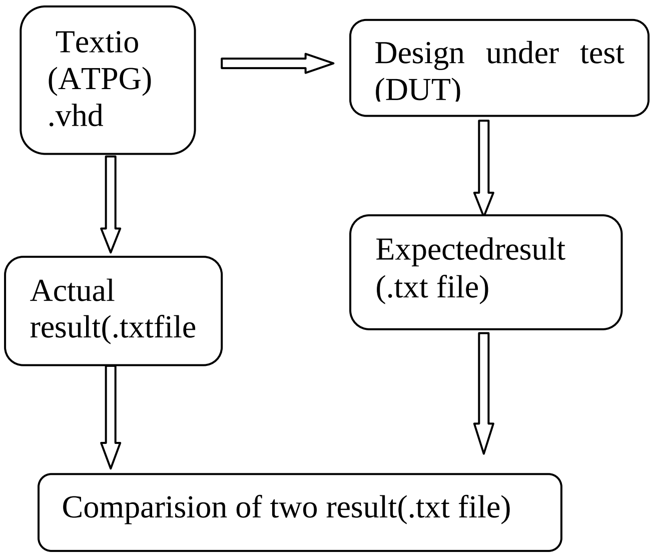
3.1 Fault Model And Fault Injection

The definition of the RTL fault model and the fault- injection algorithm encompasses modeling of faults for synthetic, Boolean and logical operators, sequential elements and fan-out/stem variables, as well as the collapsing of RTL faults. RTL faults are depicted with crosses (“x”) in Figure 1. When RTL constructs contain synthetic operators, faults are injected only on the input variables of such operators. During logic synthesis, the synthetic operators are replaced by combinational circuits implementing the respective functions, e.g., adder, subtracter, comparator, etc. The internal details of such functions represented by synthetic operators are not available at the RT level and therefore only the subset of the checkpoint faults of the gate-level representation of these operators, namely, the primary input faults are modeled. When a function is represented using RTL constructs that contain Boolean operators, faults are injected on variables that form Boolean equations. Some internal signals of these constructs are available at the RT level and therefore RTL faults are placed at primary inputs and internal nodes including signal stems and fan-outs. The post-synthesis gate-level representation of such a construct may be structurally different from the RTL Boolean representation. However, some RTL faults have equivalent faults in a collapsed gate-level fault-list of any post-synthesis design.

3.2 Fault Simulation

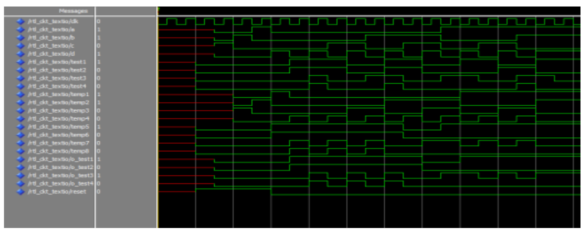
The operation of DUT is analysised on the basis of inputs and outputs. The variation in the output indicates the presence of fault in the circuit.

The steps of testing are demonstrated in the below flow diagram:



The faulty circuit is simulated first using software modelsim.After this,detection of fault is done by analysis the individual paths of transmission of output inside the circuit, they are known as test points.
In our circuit four testpoints are there.Finally, the results of good cicuit i.e. expected results and the results of faulty circuit i.e. actual results , both are compared by simulating the circuit in which expected results are taken as inputs of the circuit.

The values of signals sf_test1, sf_test2, sf_test3, sf_test4 indicateds the presence of stuck-at faults in the circuit. The signals values ie. sf_test1, sf_test2, sf_test3, sf_test4, for faulty circuit has the values of testpoints, which indicates the fault :



```
Messages
/rtl_ckt_textio/clk      0
/rtl_ckt_textio/a        1
/rtl_ckt_textio/b        1
/rtl_ckt_textio/c        0
/rtl_ckt_textio/d        1
/rtl_ckt_textio/test1    1
/rtl_ckt_textio/test2    0
/rtl_ckt_textio/test3    0
/rtl_ckt_textio/test4    0
/rtl_ckt_textio/temp1    1
/rtl_ckt_textio/temp2    1
/rtl_ckt_textio/temp3    0
/rtl_ckt_textio/temp4    0
/rtl_ckt_textio/temp5    1
/rtl_ckt_textio/temp6    0
/rtl_ckt_textio/temp7    0
/rtl_ckt_textio/temp8    0
/rtl_ckt_textio/o_test1  1
/rtl_ckt_textio/o_test2  0
/rtl_ckt_textio/o_test3  1
/rtl_ckt_textio/o_test4  0
/rtl_ckt_textio/reset    0
```

IV. Results

A versatileRTL-ATPG algorithm is presented that can generate test vectors for almostany type of single-clock functional RTL design. The algorithmuses a data structure called ADD that helps it to tackle controland data flow in an unified fashion and a nine-valued algebrathat helps it to do justification and propagation at the RTL.

Here, we are using the concept of textio for the detection of stuck-at fault in the circuit.” Textio” is one of the easier way of fault modelling and fault coverage is calculated without any consicounses.

Below given output file which is generated after comparison of actual results and expected results indicates the presence of stuck-at fault.

In the above table of output, A,B,C,D& RESET are taken as inputs.Test1,Test2,Test3,Test4 are taken as the test points of the circuit. The values of outputs i.e. SF_Test1, SF_Test2, SF_Test3, SF_Test4 indicates the presences of stuck-at fault in the circuit.If there is the fault for any one of the inputs then, SF_Test1,SF_Test2,SF_Test3,SF_Test4 takes the values of Test1,Test2,Test3,Test4 otherwise it has the null ’z’ value.

Since for the set of 15 inputs, we get the proper outputs and fault is detected for any particular combination of inputs,therefore fault coverage for given circuit is 100%.

V. Conclusion

The very first method to fault modelling methodology has used a validation test sets to generate test sequence that have goos stuck-at fault coverage for RTL circuits.This method results in very small test generation times.

After this method,RTL-ATPG algorithm was presented to generate test vectors for single clock functional RTL design.thealgoritm uses a data structure called ADD.

But here in this paper concept of textio is used to detect the stuck-at fault in RTL circuit.Also, this method results in easier estimation of fault coverage.This leads to the better efficiency of this method than previously used methods of fault coverage in RTL circuits.

References

[1] *R. C. Ho and M. A. Horowitz*, “Validation coverageanalysis for complex digital designs,” in Proc. Int. Conf. Comput.-Aided Des., Nov. 1996.

[2] *I. Ghosh, A. Raghunathan, and N. K. Jha*, “A design for testability technique of RTL circuits using control/data flow extraction,” in Proc. Int. Conf. Comput.-Aided Des., Nov. 1996.

[3] *D. J. Moundanos, J. A. Abraham, and Y. V. Hoskote*, “Abstraction techniques for validation coverageanalysis and test generation,” IEEE Trans. Comput. Jan. 1998.

[4] *IndradeepGhosh and Srivaths Ravi*, “On AutomaticGeneration of RTL Validation Test BenchesUsing Circuit Testing Techniques” Fujitsu Laboratories of America, Sunnyvale, CA 94086,NECLaboratoriesAmerica, Princeton, NJ 08540, 2001.

[5] *IndradeepGhosh and MasahiroFujita*, “Automatic Test Pattern Generation for FunctionalRegister-Transfer Level Circuits UsingAssignmentDecisionDiagrams” ieee transactions on computer-aided design of integrated circuits and systems, vol. 20, no. 3, march 2001.

[6] *L. Lingappan and N. K. Jha*, “Unsatisfiabilitybasedefficient design for testability solution for register-transferlevel circuits,” in Proc. VLSI Test Symp., May 2005.

[7] *SumaM.S. ,K.S.Gurumurthy “*FaultCoverage for digital circuits at RTL*”2011.*

[8] *SarveshPrabhu, Michael S. Hsiao, LoganathanLingappan and VijayGangaram,* “A SMT-based Diagnostic Test GenerationMethod for Combinational Circuits” ieee 30th vlsi test symposium (vts) 2012.

reset	A	B	C	D	O_test1	O_test2	O_test3	O_test4	test1	test2	test3	test4	SF_test1	SF_test2	SF_test3	SF_test4
0	1	0	0	0	0	0	0	0	0	0	0	0	z	z	z	z
0	0	0	0	1	1	1	0	0	1	1	0	0	z	z	z	z
0	0	0	0	0	1	1	1	1	1	1	1	1	z	z	z	z
0	0	0	0	1	1	1	0	0	1	1	0	0	z	z	z	z
0	0	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	0	0	0	0	1	0	0	0	1	z
0	0	1	0	0	1	1	1	1	1	1	1	1	z	z	z	z
0	0	1	0	1	1	1	0	0	1	1	0	0	z	z	z	z
0	0	1	1	0	0	0	0	0	0	0	0	0	z	z	z	z
0	0	1	1	1	0	0	1	0	0	0	1	0	z	z	z	z
0	1	0	0	0	1	0	1	0	1	0	1	0	z	z	z	z
0	1	0	0	1	1	0	0	0	1	0	0	0	z	z	z	z
0	1	0	1	0	1	0	1	0	0	1	0	0	0	1	0	z
0	1	0	1	1	1	0	0	0	0	1	1	1	0	1	1	1