Controllability and Observability

Kalyana R. Kantipudi
Dept of Electrical & Computer Engineering
Auburn University
Auburn, AL-36849
kantikr@eng.auburn.edu

Abstract: *Testability measures* play an important role in VLSI testing. The circuit complexity is increasing every day and so is the demand for efficient testability measures. So a brief overview of the testability measures: Basic Definitions, their need, their classifications, formulation, their significance on every day testing life and their present limitations are presented.*

Introduction:

As circuits become larger, their testing complexity raises exponentially. The demand for efficient testing techniques increases and every testing technique needs a basis to analyze the circuits. This basis called the testability measures play an important role in analyzing the circuit and also while building test patterns for them. The testability of a circuit is defined by the controllability and observability of it. As the circuits are becoming complex, there is high need for much more efficient testability measures. So this paper gives you a brief overview on the various testability measures available. First the basic definitions of controllability & observability are given, their need explained and then various testability measures are discussed followed by an overview on: their prominence in VLSI testing, their limitations and the need for future improvements.

Definitions:

Controllability of a digital circuit is defined as the difficulty of setting a particular logic signal to 0 or 1. Observability for a digital circuit is defined as the difficulty of observing the state of a logic signal [1]. These definitions are applicable for both the combinational testability measures and the sequential testability measures. But, the combinational testability measures are cost functions measuring the difficulty of setting or observing a signal in spatial domain, while the sequential testability measures are cost functions which estimate difficulty in temporal (time) domain [2].

The sequential controllability gives a rough measure of the number of times various flip-flops must be clocked to control a signal and the sequential observability measures the number of times various flip-flops must be clocked to observe a signal. Generally, these sequential measures characterize the test length.

Need for Testability Measures:

- An efficient ATPG should have the circuit testability information along with the circuit structural information. It uses testability measures to find testability information of the nodes, so that it can find the nodes which can be set easily.
- Every chip designer wants to have some testability information of his chip. He wants to know how much time it will take to fully test his chip. He wants to know the problem areas in the design where modification can ease the testability problem. It will be more advantageous if he gets that information in the early stages of the design cycle. For those designers, testability measures will be of great use. Using the testability measures they can find “the hard to test areas” in their design.

The Testability Measures:

Initial controllability measures ranged from 0 to 1 and the controllability of node A in a given logic circuit is correlated with the percentage of nodes in the circuit that must be set to specified logical values in order to justify a logic value on the node A. The observability is correlated with the percentage of nodes in the circuit that must be set to specified logical values in order to justify a logic value on the node and to observe the node value at the output.

---

*Note: Earlier, the gate count, the number of test vectors needed to achieve a given fault coverage and also the detection probability [5] are considered as testability measures. But, in this paper only the controllability and the observability are considered as testability measures.

Gate-Level Measures:

L. H. Goldstein, with the intention of providing a quantitative measure of the difficulty of controlling and observing the logical values of internal nodes from consideration of circuit topology, introduced a new set of testability measures in his paper on the controllability/observability analysis. These measures known famously as the "SCOAP measures", considers circuit at the gate level and computes six values for each node in the circuit: combinational zero controllability (CC0), combinational one controllability (CC1), combinational observability (CO), sequential zero controllability (SC0), sequential one controllability (SC1) and sequential observability (SO). Sequential 0 and 1 controlabilities of a node N, SC0(N) and SC1(N), estimate the minimum number of sequential nodes that must be set to specified logical values in order to justify a 0 or 1, respectively, on node N. They are cost functions measuring the difficulty in a temporal sense of accomplishing the node justifications required to control or observe a specified node in the circuit.

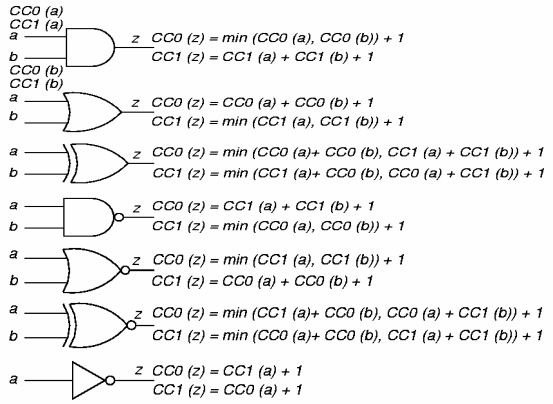
If I is a primary input node of a digital circuit, then the Controllabilities of node I are defined as follows:

$$CC0(I) = 1$$
$$CC1(I) = 1$$
$$SC0(I) = 0$$
$$\text{and } SC1(I) = 0$$

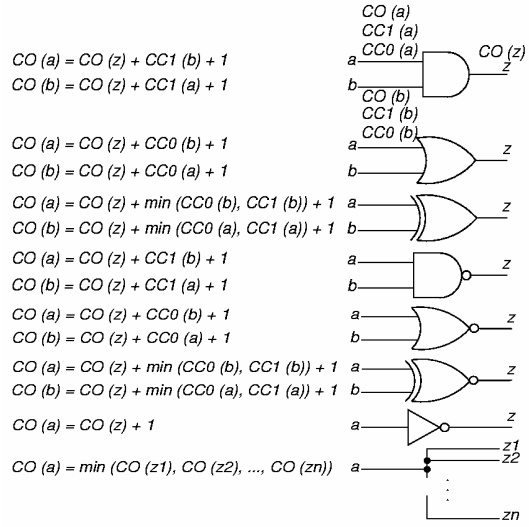
If U is a primary output node of a digital circuit, then the observabilities of node I are defined as follows:

$$CO(U) = 0$$
$$\text{and } SO(U) = 0$$

The combinational controllabilities are formulated as: [1]

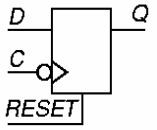


The combinational observabilities are formulated as:[1]



The sequential testability measures are incremented only when they pass through a Flip-flop.

Assuming a Flip-flop with synchronous RESET, the sequential testability measures are given by:



SC1 (Q) = SC1 (D) + SC1 (C) + SC0 (C) + SC0(RESET) + 1

SC0 (Q) = min [SC1 (RESET) + SC1 (C) + SC0 (C), SC0 (D) + SC1 (C) + SC0 (C)] + 1

SO (D) = SO (Q) + SC1 (C) +SC0 (C) + SC0(RESET) + 1

SO (RESET) = SO (Q) + SC1 (Q) + SC1 (RESET) + SC1 (C) + SC0 (C) + 1

SO (C) = min [SO (Q) + SC1 (Q) + SC0 (D) + SC1 (C) + SC0 (C),
SO (Q) + SC1 (Q) + SC1 (RESET) + SC1 (C) +SC0 (C),
SO (Q) + SC0 (Q) + SC0 (RESET) + SC1 (D) + SC1 (C) + SC0 (C)] + 1

The same formulae apply for the combinational measures but like sequential measures, they are not incremented by one.

The SCOAP testability measures analyze the circuit at the gate-level; the values they generate are just estimates because of the simplifying assumptions made by its algorithm. These assumptions will produce some erroneous values which will be discussed at the end of this paper.

SCOAP does not directly produce fault-specific data, but we can generate the specific testability data [8]. For example, if we want to find the whether a s-a-0 fault at a fault site P can be detected or not, we can find its testability by,

$$\text{Testability(p, s-a-0)} = CC1(P) + CO(P)$$

The computation of SCOAP testability measures is improved by a new testability analysis tool named "HISCOAP"[10]. It is a hierarchical testability analysis tool which makes use of the hierarchy of the circuit. The tool significantly reduces the memory and computational resources while computing the testability measures.

The need for other types of testability measures arrives form the fact that there will be thousands of logic gates to analyze in a circuit and also using gate-level testability measures for DFT is of no use as every IC will be having groups of modules which cannot be redesigned.
There are some techniques that deal with testability measures at higher levels of the circuit. Some deal with testability measures at the module level [7] and some deal at the register-transfer level [12]. They use the testability measures to aid in designing circuits with improved testability. So, let's first look at module level testability measures.

Module-Level Measures:
In these measures, the circuit is described in terms of modules, each having its own independent characteristic function. So, the module level testability analysis uses spectral technique [7]. Initially, every output of a module is represented as a weighted-function of the primary inputs and a truth table is designed for each primary output of the module. Then a probability spectrum of the module is obtained form those truth tables.

The probability function of a signal Y is represented as P(Y=1) and using the probability functions of independent signals, probability functions of Boolean functions of those signals are calculated:

If $Y = Y_1.Y_2$ then
$Y = P(Y=1) = P(Y_1=1).P(Y_2=1) = Y_1.Y_2$

If $Y = Y_1 + Y_2$ then (if Y_1 and Y_2 are dependent)
$Y = P(Y=1) = P(Y_1=1) + P(Y_2=1) + P(Y_1=1).P(Y_2=1)$
$= Y_1 + Y_2 + Y_1.Y_2$

Then the controllability and observability spectra are obtained from the probability spectrum of the module.

The controllability of a wire x_i in the circuit (denoted as $C(x_i)$ or C_i) is defined as follows:

$C(x_i) = P(x_i=1) - P(x_i=0) = 2.P(x_i=1) - 1$
So, the value of C ranges in [-1,1].

The smaller the value of $|C_i|$, the closer it is to $P(x_i=1)$ than to $P(x_i=0)$. So, it means it is easy to control the wire x_i.

Using a complex algorithm, the controllability and observability values are determined. The algorithm also takes care of the reconvergent fanouts. But the problem with these measures is that every IC has a number of unlike modules, each having its own responses. Also the modules may range from a logic gate to a large circuit. Unlike the gate level analysis they cannot list the controllability/ observability formulae. So, the algorithm will be NP-Complete due to its complexity.

Register-Transfer-level Measures:
These high-level measures are presented by J. E. Stephenson and J. Grason in 1976, but were initially not used due to their complexity of calculation. Later, J. Grason developed a Testability Measurement Program named "TMEAS", which made the algorithm linear and so useful for analyzing the testability [13].

Let's look at these testability measures in detail. These testability measures assume that a circuit is a network of components interconnected by unidirectional links. If a link fans out to several destinations, each portion of the link from its fan-out point to a destination is termed a wire. The structural model assumes that the goal in testing is to test each component and/or link in the circuit for possible faults. Testing a given component involves two related tasks. First, the inputs to the component must be controlled to some desired values. This is called the control task. Then, somehow the values that appear on the outputs of the component must be observed to see if they are correct or not. This is called the observation task [12].

All primary inputs are assumed as perfectly controllable and the primary outputs to the circuit are perfectly observable. The task of testing a component is that of first propagating controllability from the circuit primary inputs, through other components, to the inputs of the component and then propagating observability from the outputs of the component through other components to the primary outputs of the circuit.

The testability measures are normalized between 0 and 1 to reflect the ease of controlling and observing the internal nodes. For each signal line

(s), the controllability of s is denoted as CY(s) and the observability of s as OY(s). The values for the CY s and the OY s of all the signal lines are derived by solving a system of simultaneous equations with the CYs and the OYs as unknowns.

If x_1, x_2,……., x_n represent the input variables of a component, and z_1, z_2,……., z_m represent the output variables.
Then the value of CY for each output z_j is calculated by:

$$CY(Z_j) = CTF \times \frac{1}{n}\sum_{xi=1}^{n} CY(x_i)$$

Where, CTF is the "Controllability Transfer-Factor" of the component.

Let $N_j(0)$ and $N_j(1)$ be the numbers of input combinations for which z_j has value 0 and 1, respectively. Then

$$CTF \cong \frac{1}{m}\sum_{j=1}^{m}\left(1 - \frac{|N_j(0) - N_j(1)|}{2}\right)$$

So, the value of CTF lies in between 0 & 1 and each output controllability is assigned the same CTF value.

The expression used to calculate OY for each input x_i is

$$OY(x_i) = OTF \times \frac{1}{m}\sum_{j=1}^{m} OY(z_j)$$

Where, OTF is the Observability Transfer Factor of the component.

Let NS_i be the number of input combinations for which the change of x_i results in a change of output. Then OTF is given by

$$OTF \cong \frac{1}{n}\sum_{i=1}^{n}\frac{NS_i}{2^n}$$

NS_i gives the number of input combinations that can sensitize a path from x_i to the output.

The OTF measures the probability that a faulty value at any input will propagate to the outputs and it ranges between 0 & 1. Each input observability of the component is assigned the same OTF value.

Formulations for Fan-outs:
Let s be a fan-out stem and k be the number of its branches. Then the CYs of each fan-out branch is given by:

$$CY = \frac{CY(s)}{(1 + \log k)}$$

And the observability of the fan-out stem s is given by:

$$OY(s) = 1 - \prod_{i=1}^{k}[1 - OY(b_i)]$$

Where, b_is are fan-out branches of s.

While handling sequential components, the sequential components are modeled by adding feedback links around the components that represent the internal states of the sequential components.

The testability measures dealt so far are just static. They perform topological analysis of the circuit to determine the testability measures. These testability measures are not efficient for helping designers to design fully testable circuits for testability. So now, lets looks at an interactive testability measure which is not only useful for DFT but also significantly reduces the generation time.

An interactive testability analysis tool named "ITTAP" is presented in 1982. It holds testability information of every standard cell in a library and uses the library to map the cells. So it considers a digital circuit as an interconnection of standard cells available within an ITTAP library. The ITTAP library can be updated by adding new cells which are needed for the circuit design. Once the testability measures of the entire circuit are calculated the tool stores them. So, when a part of the circuit is modified to improve its testability, then it won't have to calculate the entire testability measures of the circuit. Only the values in the effected portion of the circuit are calculated. This greatly reduces the generation time. In fact, the ITTAP testability measure generation time is several orders less than the normal method used to calculate the SCOAP measures.

Testability Measures for Detection-Probability:

The controllability and observability measures are not just for calculating the testability of a node, gate or circuit. Some latest fault analysis techniques make use of these testability measures to calculate the fault coverage. STAFAN (STatistical Fault ANalysis) redefines the controllability and observability of a line as the probabilities of controlling and observing the line [4]. Controllability of a line is estimated by collecting the statistics of activity on that line and the Observability is then computed from the estimated Controllabilities. The product of the appropriate controllability and observability gives the detection probability of a fault.

Suppose a fault with detection probability p(t), whose range lies in between 0 and 1, can be represented as:

$$P(t) = e^{-\alpha t}$$

For SCOAP, 't' is the sum of the combinational observability and the combinational controllability of the faulty line.

If a circuit is having N faults having testability measures $t_1, t_2, \ldots\ldots, t_N$ and if the fault detection performance of each test vector is statistically independent of each other, then after applying V vectors, the probability of detecting the i^{th} fault is

$$1 - [1 - p(t_i)]^v$$

The fault coverage is then given by

$$f(v) = \frac{1}{N} \sum_{i=1}^{N} \left\{ 1 - [1 - p(t_i)]^v \right\}$$

So, the fault coverage is

$$f(v) = 1 - \frac{1}{N} \sum_{i=0}^{N} \left[1 - e^{-\alpha t_i} \right]^v$$

Where, t_i is the testability measure of the i^{th} fault.

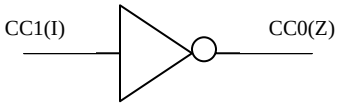
This way gives only an approximate fault coverage for a circuit. Later, a precise relationship is developed between circuit testability and fault coverage. The testability so developed takes into consideration not only the circuit topology, but also the characteristics of test vectors [6].

Limitations of the Present Testability (C/O) Analysis:

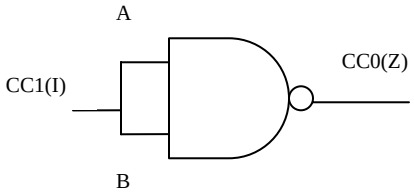
- While computing the SCOAP testability measures, static analysis is done based on the structural description of the circuit considered. But the actual testing process also includes a test pattern generator which is not considered by the testability measure. The values SCOAP generates will not be accurate because it relies on a restricted information source, the structural description of the circuit [8].

- The testability analysis has to be linear. The analysis cannot be NP-complete because we can generate a set of ATPG test patterns in that exponential time. In order to be linear, all testability measures make some simplifying assumptions. So, the testability measures are only estimates, which cannot be fully relied on.

For example, SCOAP assumes that all inputs to all logic elements are independent. But, that will not be the case for reconvergent fan-outs. Also consider the following example:



CC0(Z) = CC1(I) + 1



CC0(Z) = CC1(A) + CC1(B) + 1
= 2 CC1(I) + 1

In the above two simple realizations of the inverter, due to the assumption that all the inputs are independent, both the realizations will be having different controllability values [8].

- As they are just estimations, they should not be the only criteria while designing a circuit which is designed for testability. [8] gives a very interesting example. Since the testability values in SCOAP increase with required test generation effort, the faults with low testability values are quickly

detected and their probabilities of detection quickly approach one. In contrast, faults with the greatest testability values are detected more slowly, and their final probability of detection is only about 0.7. This looks good from the detectability prospective, but it means even the least testable faults have a 0.7 probability of detection. Thus if a designer tries to improve the circuit testability by reducing the testability values of all those faults in the circuit with high testability values, he will be wasting 70% of his effort.

- A drawback of using the SCOAP measures is that it lumps all the faults together without distinguishing hard-to-test faults form easy-to-test faults [9]. This extenuates the significance of using SCOAP in DFT.

Conclusion:

The circuits are becoming complex and they are to be designed for testability. So, a need for accurate testability measures is imminent. The gate-level testability measures are needed for designing an efficient ATPG. But, the higher level testability measures (mainly the Register-Transfer level measures) are to be seriously considered, because they are the ones which are helpful in DFT. Unless a 100% accuracy is achieved in those testability measures, they should not be considered as the only means for verifying the circuit; whether its design is fully testable or not. The ATMG (Automated Testability Measures Generation) tools like the TMEAS [13] and ITTAP [11] are of significant use while designing circuits for testability. The ATMG tools can be further improved in terms of accuracy and speed.

References:

[1] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing – for Digital, memory & mixed-signal VLSI circuits*. Boston: Kluwer Academic Publishers, 2000.

[2] L. H. Goldstein, "Controllability / Observability Analysis of Digital Circuits," *IEEE Transactions on Circuits and Systems,*Vol. CAS-26, September 1979, pp. 685-693.

[3] L. H. Goldstein and E. L. Thigpen, "SCOAP: Sandia Controllability/ Observability Analysis Program," *Proceedings of 17th Design Automation Conference*, Minneapolis, MN, June 1980, pp. 190-196.

[4] S. K. Jain and V. D. Agrawal, "STAFAN: An alternative to fault simulation," *Proceedings of 17th Design Automation Conference*, Albuquerque, New Mexico, June 1984, pp. 18-23.

[5] H. Wunderlich, "PROTEST: a tool for probabilistic testability analysis," *Proceedings of the 22nd ACM/IEEE conference on Design automation,* Las Vegas, Nevada, June 1985, pp.204-211.

[6] S. C. Seth, V. D. Agrawal, H. Farhat, "A theory of testability with application to fault coverage analysis," *Proceedings of the 1st European Test Conference*, Paris, April 1989, pp. 139-143.

[7] Z. Zuan, "The spectral technique in Controllability/ observability measures of digital circuits," *Multi-valued logic Proceedings of the 19th International Symposium on*, Guangzhou, China, May 1989, pp. 100-106.

[8] V. D. Agrawal and M. R. Mercer, "Testability Measures-What Do They Tell Us?" *Proceedings of the International Test Conference*, Philadelphia, PA, November 1982, pp. 391-396.

[9] L. Huisman, "The reliability of approximate testability measures," *IEEE Design and Test of computers,* December 1988, pp. 57-67.

[10] C. P. Ravikumar and H Joshi, "HISCOAP : A Hierarchical Testability Analysis Tool," *The 8th International Conference on VLSI Design*, January 1995, New Delhi, India, pp. 272-277.

[11] D. K. Goel and R. M. McDermott, "An interactive testability analysis program-ITTAP," *Proceedings of the 19th Design Automation Conference*, January 1982, pp. 181-186.

[12] J. E. Stephenson and J. Grason, "A testability Measure for Register Transfer Level Digital Circuits," *Proceedings of 1976 International Symposium on fault Tolerant Computing*, Pittsburgh, June 1976, pp. 101-107.

[13] J. Grason, "TMEAS, A Testability Measurement Program," *Proceedings of the 16th Design Automation Conference*, June 1979, pp. 156-161.