

Realization of a Simplified Controllability Computation Procedure: a MATLAB- SIMULINK Based Tool

Ali Al-Lawati and Afaq Ahmed

Department of Information Engineering, College of Engineering, Sultan Qaboos University, P.O.Box 33, Al Khod 123, Muscat, Sultanate of Oman, Email: Lawati@squ.edu.om.

ABSTRACT: With the use of MATLAB-SIMULINK a simple tool has been developed to compute the controllabilities of nodes as well as branches of digital system networks. The primary advantage of this tool may be for such conditions where the study of controllability analysis of a system with different possible designs is required to give an optimal design solution. The ease of network simulation procedures together with the simple methodologies for inputting the different assigned parameters to the respective elements of the system in dynamic environments, are additional attributes of this tool.

KEYWORDS: Testability, Digital Electronic System, Controllability, Controllability Transfer Function, MATLAB-SIMULINK.

1. Introduction

The increasing functional complexity of digital electronic components and systems makes testing a challenging task, particularly under the constraints of maintaining a high quality of products at low and competitive market prices. Reports reveal that, on average, a proportion of up to 70% of the total production cost is required as against the testing cost (Bennetts, 1984; Williams, 1986; Könemann, 1998; Zorian, 1999). Thus, an optimal test strategy can give a substantial competitive advantage in a market comprising billions of electronic components and systems. It is, therefore, not surprising that the major technology organizations like "National Technology Roadmap for Semiconductors" of the United States treat testing as a key technology component with significant importance (Könemann, 1998).

Although testing incurs a lot of effort, it is an important means of reducing the overall cost significantly. While the actual material cost is only a negligible proportion of the product value, the cost of repair increases by a factor of more than 10 (see Figure 1) with each production stage (Bennetts, 1984; Bardell *et al.* 1987). It is much cheaper to reject several dies than to locate and exchange a defective chip in a complete system. As a consequence, no customer is willing to bear the risk of using defective components and most likely can only accept suppliers that guarantee low defect rate and/or often perform an incoming test for supplied parts.

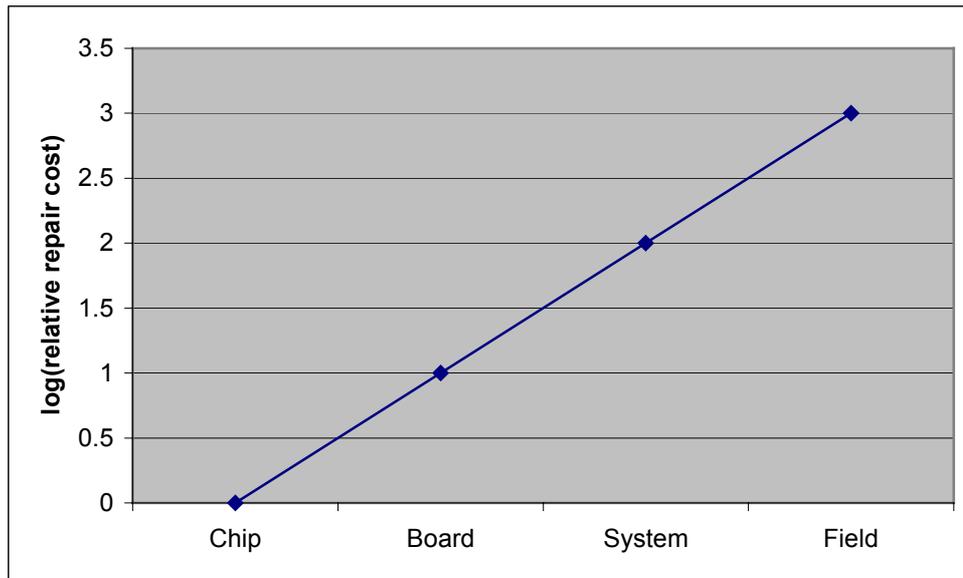


Figure 1. Testability cost.

Low defect rate of a product can be guaranteed by extensive outgoing product tests only. VLSI chips have reached an enormous complexity, and yet their density doubles every 2 years (Williams, 1986; Sun and Serra, 1992; Könemann, 1998). This makes it impossible to rule out faults during design and production, even with the best design tools and fabrication processes available. However, short time-to-market is critical to profitability. If testing facilitates rapid diagnosis and thus provides a means to avoid fatal production delays resulting from excessive debug time or shipping defective products, it is worth the additional cost.

Availability of a system (or of redundant system's components) can be significantly increased if testing is employed to allow rapid diagnosis after a failure. These facts clearly show the economic potential of an efficient test strategy in the rapidly growing area of dependable systems. Although the two fields of design and test application appear quite disjoint it is technically and economically attractive to merge them. It will be shown that the concepts of Design-For-Testability (DFT) and Built-In Self-Test (BIST) provide an ideal starting point for a unified test approach for the complete life cycle of a digital electronic system.

It is the fact that testability relates to cost, which needs to satisfy some technical requirements within a budgeted testing allowance. However, this fact necessitates the measurements of testability levels at each node of digital electronic circuits. Therefore, numerical assessment of the controllability features of a circuit design, leading to a measurement of circuit's testability, can be of great importance particularly during the design stage of the circuit. The method of analyzing digital electronic systems for its controllability is a problem that is not being adequately addressed during the design phase. Continuous controllability monitoring and analysis is essential to refine the system design for the better goals like enhancements of transparency, reliability, availability, and maintainability. There is no specific tool available which can give the complete system's controllability analysis for different behavioral responses of nodes. Researchers in the field of digital circuit testing have been constantly working in the process of developing a tool to study the controllability / observability as well as testability of digital systems (Bennets, 1981, 1984; Berg and Hess, 1982; Cerny and Mauras, 1990; Fritzemeier, *et al.* 1989; Goldstein and Thigpen, 1980; Grason, 1979; Kapur *et al.* 1991; Koren, 1979; Ratiu *et al.* 1982; Stephenson and Grason, 1976). These tools cannot be easily implemented for analyzing systems. Their program data files are not user friendly; therefore it is difficult to describe a system and its related input and output files. Hence the need to overcome these problems is the motivation for this research work. Here we present the development, design and use of a user-friendly tool to compute and scan the complete controllabilities information of digital system circuits.

2. Controllability

Controllability reports the cost of placing a node in a circuit at a predetermined logic value. Placing this value on a primary input is “free”. However, it is possible to assign a minimal cost, say 1, to any primary input. The cost increases as the node depth in the circuit increases. This cost will also depend on the type of gate, the logic value to be imposed on the line, and whether the circuit is combinational or sequential. For example, controlling the output of a multi-input AND gate to logic 1 requires the control of all of its inputs to logic 1, while for an OR it is sufficient to control only one of the inputs to logic 1. Controlling a node at a second level requires controlling the input at level 1 as well as those at level 0 (the primary inputs). Thus, there is a cost for stepping from one level to another.

Controllability indicates how easily the system state can be controlled by the primary inputs (pins) of a circuit. Quantitative controllability ratings can be determined for each node in the circuit; most often an average over all nodes is given. In the example in Figure 2 the controllability of node d is a measure of how easy it is to set the output of block A to a logical 1 or 0 by assigning values to the primary inputs of the circuit. It is a function of the controllability of nodes a, b and c.

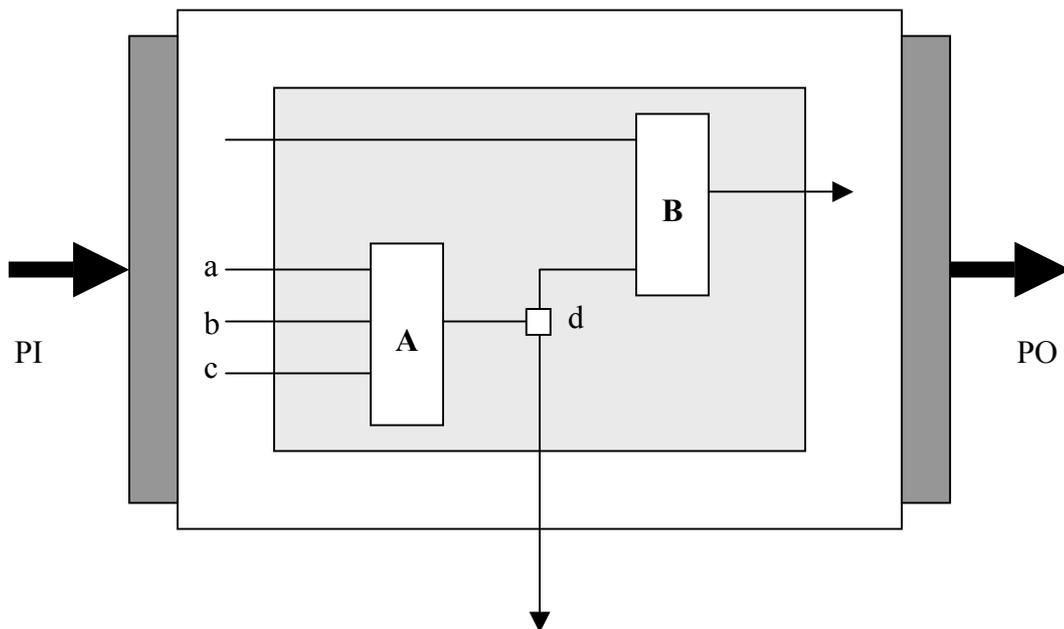


Figure 2. Test point insertion.

Testability can be substantially improved if more nodes are made available directly at primary inputs and outputs. As a consequence, test points are inserted in areas of low testability to allow direct observation or control of an otherwise internal node. Although heuristic approaches exist, optimal placement of test points is still an open issue (Eichelberger *et al.* 1991; Seiss, *et al.* 1991). Controllability of a node is an important property which indicates how easy it is to control the node on its excitations / responses, operations, and behaviors. Node designers (developers) look at the "controllability" of a node in three aspects (Aas and Mercer, 87; Butler *et al.* 1992; Jerry and Youjin, 1999; Savir, 1983; Savkin, 1998):

- a) behavior control,
- b) feature customization, and
- c) installation and deployment.

The first deals with the controllability of its behaviors and responses corresponding to its operations and excitations. The next refers to the built-in capability of supporting customization and configuration of its internal functional features. The last refers to the control capability on node

installation and deployment. However, once the system is designed and installed, the continuous study of controllability of its behaviors is an essential aspect to maintain the system.

3. Controllability computational background

Since the controllability of a node is a measure of probability to set the response with desired signal using the excitation signals, controllability of a system depends on how correct a node is transferring its set (designed) controllability property to its output lines.

Definition 1: Controllability Transfer Function (CTF) is a measure of the level of controllability transfer of a node to its succeeding node(s) or branch(es).

In a digital system, Controllability Transfer Function (CTF) of any node can be computed by simulating the test inputs and their behavioral output characteristics with respect to the node. Using these characteristics, a mathematical relationship can be established as given below (Bennets *et al.* 1981; Bennets, 1984).

$$CTF = 1 - \frac{\text{Absolute}[N(0) - N(1)]}{[N(0) + N(1)]} \quad (1)$$

where $N(0)$ is the total number of ways that a logic 0 can be produced on the node output and similarly, $N(1)$ is the total number of ways of producing a 1. Thus, the CTF is 1 in the case where $N(0)$ and $N(1)$ are equal (as, for example, in the case of using Exclusive OR gate as a node). In the unlikely event of $N(0)$ or $N(1)$ being 0, the CTF would be 0, indicating that there is no control of the output state. Generally, $0 < CTF < 1$. Table 1 contains examples of CTF calculated values for basic building blocks of digital systems having different number of input lines.

Table 1: Calculated Values Of CTF For Basic Logic Gates.

Gate Logic	Number of Inputs			
	2	3	4	5
AND	0.5000	0.2500	0.1250	0.0625
NAND	0.5000	0.2500	0.1250	0.0625
OR	0.5000	0.2500	0.1250	0.0625
NOR	0.5000	0.2500	0.1250	0.0625
EXOR	1.0000	1.0000	1.0000	1.0000
EXNOR	1.0000	1.0000	1.0000	1.0000
NOT	1			

Definition 2: Controllability input to a node is a single valued controllability measurement of different inputs to that node.

In practice, multiple input lines with different controllability values are connected to a single node. Thus, the overall controllability of the input (CY_{Input}) to a node will be the average of the controllabilities of the individual input lines ($CY_{Input-1}, CY_{Input-2}, \dots, CY_{Input-i}, \dots, CY_{Input-N}$) feeding the node (Bennets *et al.* 1981), i.e.

$$CY_{Input} = \frac{1}{N} \sum_{i=1}^N CY_{Input-i} \quad (2)$$

Definition 3: Controllability output of a node is the value of the controllability available at the output of that node.

REALIZATION OF A SIMPLIFIED CONTROLLABILITY COMPUTATION PROCEDURE

The controllability of the output (CY_{Output}) of a node will depend on CTF and CY_{Input} (Bennets *et al.* 1981), which can be related as shown in Equation 3:

$$(C_{\text{OUTPUT}}) = \text{CTF} \times CY_{\text{Input}} \quad (3)$$

Definition 4 : The branches bifurcating from a stem branch are known as fan-outs branches.

The problem occurs at points of reconvergence of fan-out branches where the lack of independence of signal values on reconvergent paths can cause many remade decisions during a test cycle. This is effectively a lack of controllability at these points of reconvergence and is modeled by artificially reducing the controllability value at the destination of such fan-out branches (Bardell *et al.* 1987).

Let f denote the number of branches (destinations) for a particular fan-out stem. Stephenson and Grason (1976), found that good results are produced when the controllability assigned to each destination is obtained by dividing the controllability of the fan-out stem by: $1 + \log_{10}(f)$.

Definition 5: Primary inputs are those essential signals, which are the basis of the design of the system as a resource.

Definition 6: Primary outputs are those essential signals, which are the basis of the design of the system's goal.

4. Developed tool

Using MATLAB-SIMULINK, a simplified tool is developed to study the exhaustive controllability analysis of digital system circuits where the user is supposed to define only the connectivity of the circuit. To illustrate the design, a digital system circuit of full adder is considered (see Figure 3). Figure 4 shows its realized model through the use of MATLAB-SIMULINK.

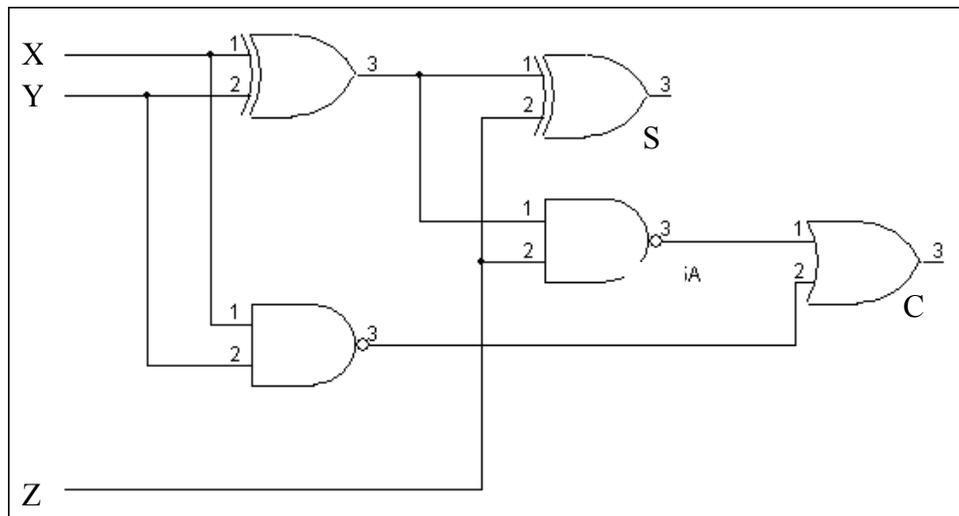


Figure 3. A circuit model of a full adder (version 1).

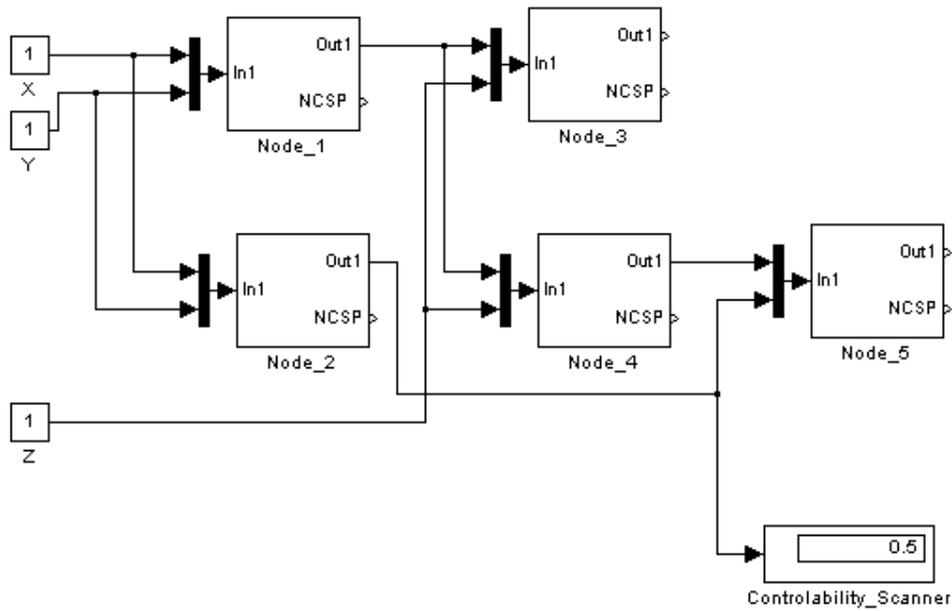


Figure 4. Simulation model of Figure 3.

In the simulated design (see Figure 4) each node box (Node_1, Node_2,...) has the same structure so that it can be copied to create as many nodes as is required in the system. This simplifies the procedure of defining the system's nodes. The node block requires information about the CTF of the node and the number (f) of fan-out branches to propagate the controllability accordingly to its f output lines. The user can provide the information about CTF and f , by invoking the respective Windows (see in Figures 5 and 6).

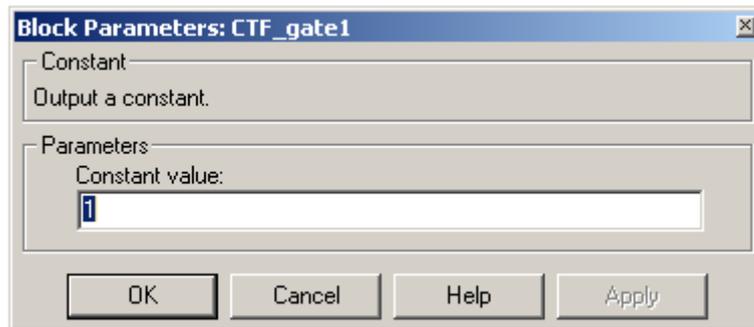


Figure 5. Control transfer function (CTF) window.

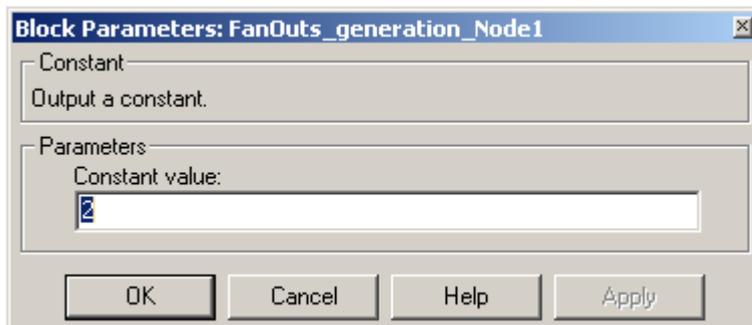


Figure 6. Fan-out window.

Table 2: Node Controllabilities of Simulated Models.

Simulated Models	Nodes								
	1	2	3	4	5	6	7	8	9
Figure 3	1.0000	0.5000	0.8843	0.4422	0.2355	x	x	x	x
Figure 8	0.5000	0.5000	1.0000	0.3843	0.7686	0.1922	0.2882	0.4884	0.1681

It can be seen from Table 2 that the nodal controllabilities at the output nodes are different despite the fact that both circuits perform the same function. Node 3 (Figure 4) and Node 8 (Figure 9) represent the sum bits for the respective circuits of Figures 3 and 8 and have different values of 0.8843 and 0.4884, respectively. Similarly it can also be observed that Node 5 (Figure 4) and Node 9 (Figure 9) represent the carry bits for the respective circuits of Figures 3 and 8 and have different values of 0.2355 and 0.1681 respectively. It is also observed that the controllabilities of connecting links to these nodes are quite different as shown in Table 3.

Table 3: Controllabilities of connecting links of some nodes of simulated models.

Simulated Models	Connecting Links							
	a ₃	b ₃	a ₅	b ₅	a ₈	b ₈	a ₉	b ₉
Figure 3	0.7686	01.000	0.4422	0.5000	x	x	x	x
Figure 8	x	x	x	x	0.1922	0.7686	0.3843	0.2882

6. Practical complex systems and validation of simulation results

To demonstrate the validation of results we further consider the practical approach of the system simulation. We simulated the complex digital systems by portioning the system into pieces of building blocks. To experiment our developed tool we hereby consider three cases of system buildings. As shown in Figures 10–12, these cases are hybrid; series (cascaded) and parallel,

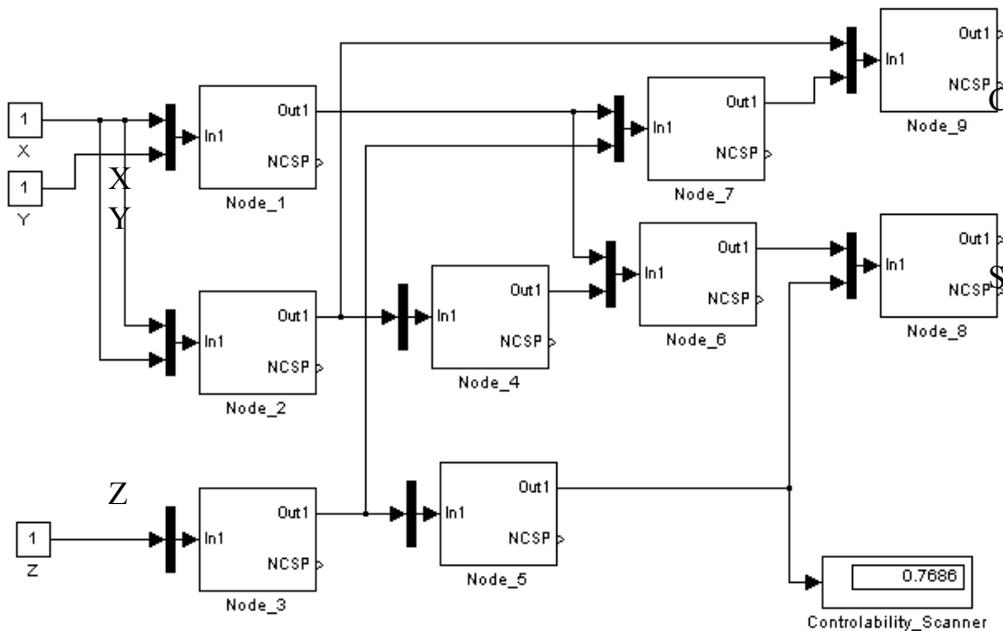


Figure 9. Simulation model of Figure 8.

REALIZATION OF A SIMPLIFIED CONTROLLABILITY COMPUTATION PROCEDURE

respectively. Let us consider the building block of the system shown in Figure 13. The scanned nodal controllability of this basic building block is found as: 1, 1, 1, 1, 0.75, 0.375, 0.2969, 0.2617, and 1123, respectively, for the nodes from 1 – 9. The average level of controllability of the block is 64.4%. The scanned nodal controllability's graphs of each case are shown, respectively, in Figures 14 – 16. This demonstrates that the use of the developed tool is feasible in total practicality.

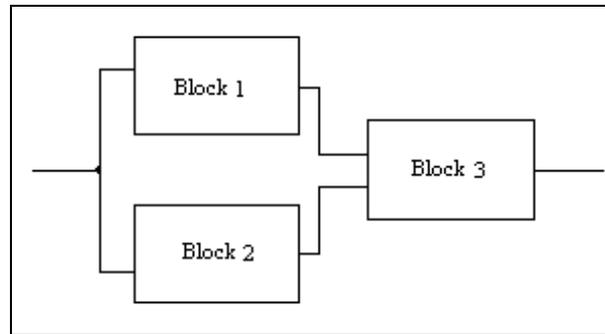


Figure 10. Hybrid model (case 1).

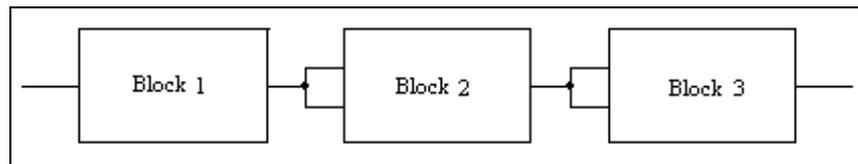


Figure 11. Series model (case 2).

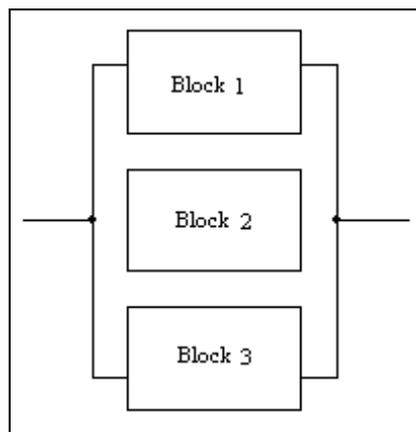


Figure 12. Parallel model (case 3).

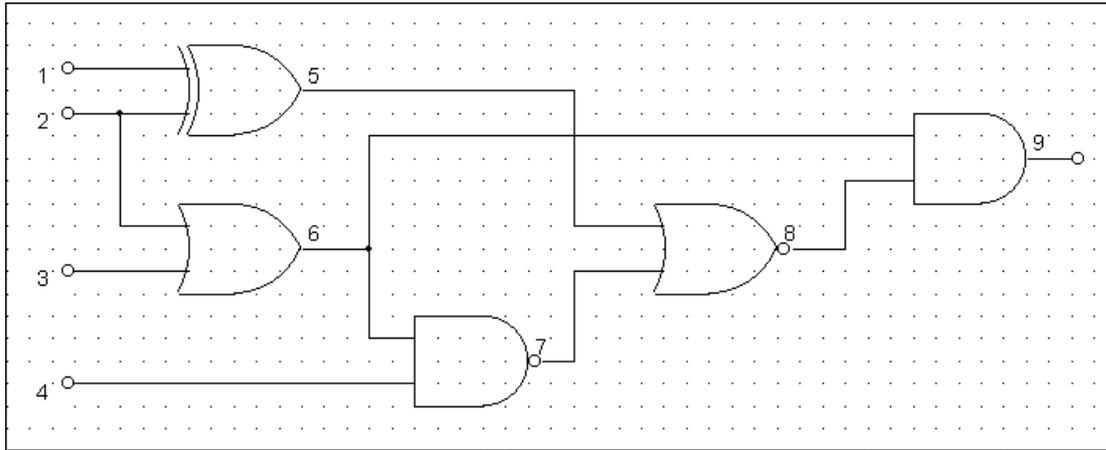


Figure 13. Model of basic building block.

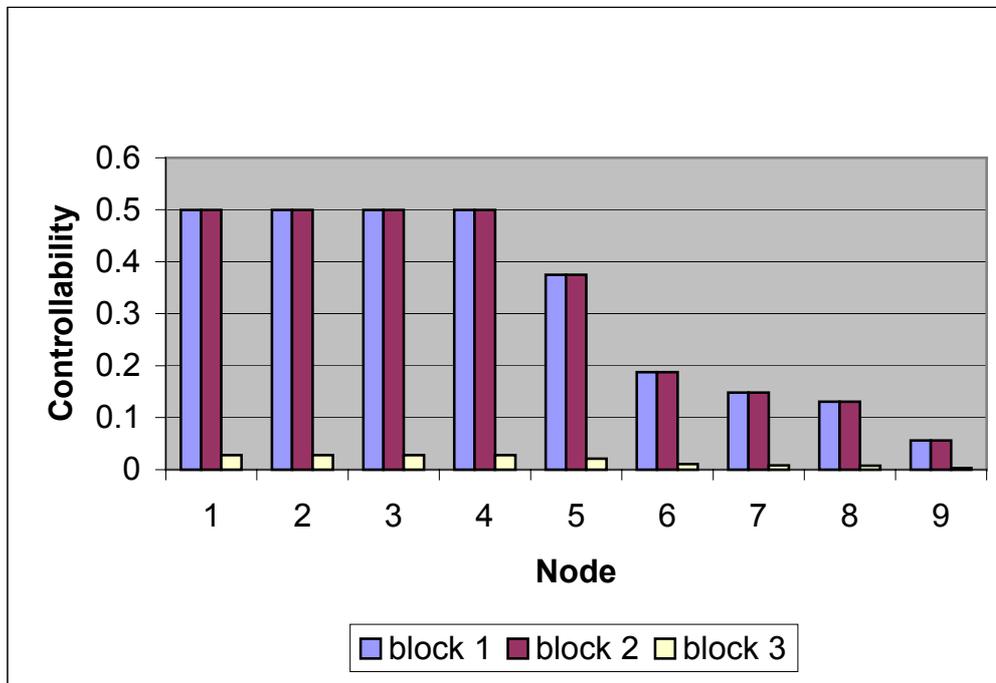


Figure 14. Scanned nodal controllability graph for case 1.

7. Discussions and conclusions

It can be seen from the results in Table 2, which were obtained from the developed MATLAB-SIMULINK-based tool, that the full adder realized in Figure 3 is much better, in terms of controllability than the one in Figure 8. The circuit of Figure 3 is much more controllable than the circuit given in Figure 8. This demonstrates that the developed tool is helpful in the design of better controllable systems.

Controllability monitoring (scanning) for each of the connecting branches as well as nodes of the circuits is an essential process to realize the targeted goal of the system. As an example, it can be visualized through the results of Table 3 that the circuit of Figure 8 needs test-point insertions to enhance the testability of the designed circuit, which demonstrates the applicability of the developed tool. Since, the program creates an exhaustive table where all the possible input instructions are demonstrating their propagated effects that are very helpful in the analysis and design of transparent systems and helps in isolating the critical nodes.

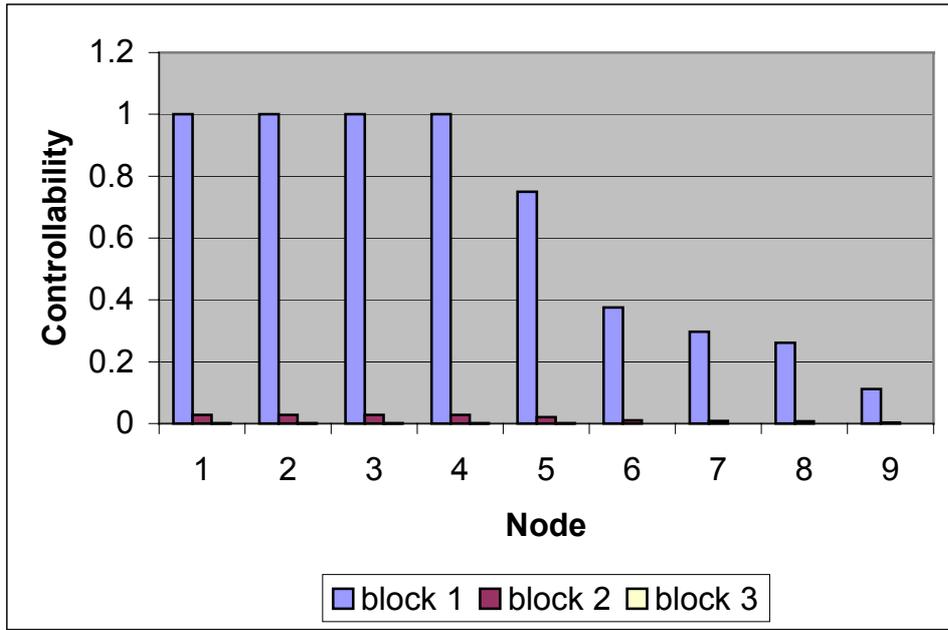


Figure 15. Scanned nodal controllability graph for case 2.

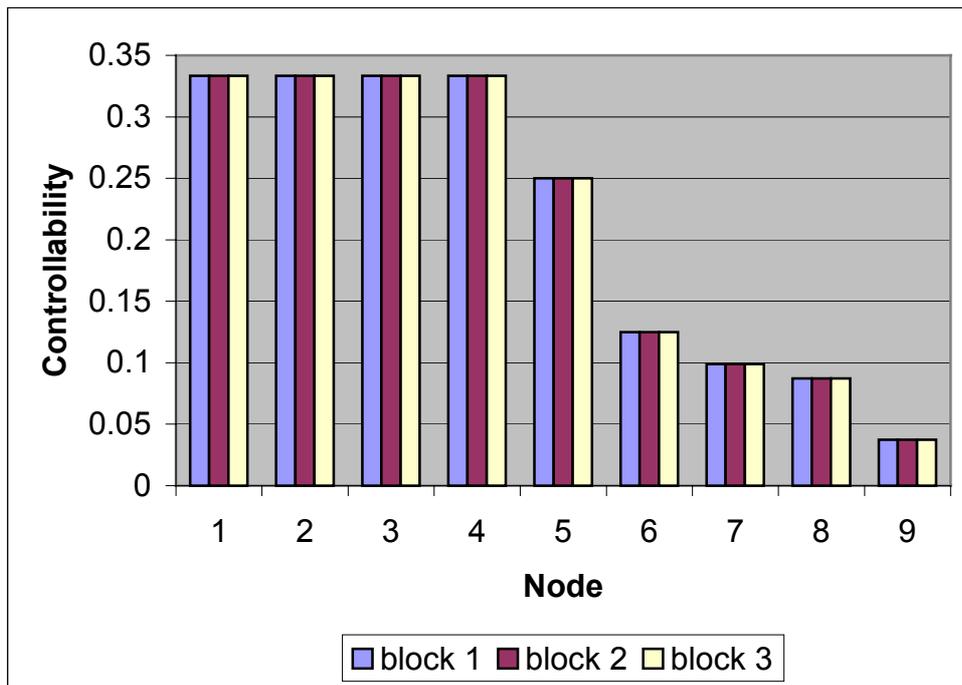


Figure 16. Scanned nodal controllability graph for case 3.

Further, Figures 13-16 demonstrate how simply the developed simulation model measures the levels of the controllability for complex digital systems. This in-house developed program has the ability to measure controllability at any node of systems consisting of any number of blocks with any type of connectivity.

This paper has demonstrated the simplified procedure to study the exhaustive controllability analysis of digital system circuits. Using the MATLAB-SIMULINK, the developed design has been made user-friendly where the simplified mechanism (although it invokes Windows) can be adapted to furnish the dynamics of the network. Also, the incorporated program for this development can easily be modified to cope with the requirement of the user. Further, graphical,

numerical and tabular forms of the results can be obtained using this incorporated program as per the suitability and the data can be easily transported as well. Unlike VHDL or pSPICE, MATLAB is the only language which provides all these attributes in totality. Since MATLAB is a common teaching tool for many of the courses of engineering curricula, the developed simulation model is not only useful for commercial purposes but it is also helpful for teaching purposes. Thus, this developed tool is of much importance particularly in such cases where it is desired to analyze the controllability of a network of a large size with different and changing parameters.

References

- AAS, E.J. and MERCER, M.R. 1987. Algebraic and Structural Computation of Signal Probability and Fault Detectability in Combinational Circuits, Proceedings of the 17th International Symposium on Fault-Tolerant Computing, pp. 72-77.
- BARDELL, P.H., McANNEY, W.H., and SAVIR, J. 1987. *Built-In Test For VLSI (Pseudorandom Techniques)*, John Wiley & Sons, pp. 43-60.
- BENNETTS, R.G., MAUNDER, C.M., and ROBINSON, G.D. 1981. CAMELOT: A Computer-Aided Measure For Logic Testability, *J IEEE Proceedings*, **E-128**: 177-189.
- BENNETTS, R.G., 1984. *Design of Testable Logic Circuits*, Addison Wesley.
- BERG, W.C., and HESS, R.D. 1982. COMET: A Testability Analysis And Design Modification Package, Proceedings International Test Conference, pp. 364-378.
- BUTLER, K.M., KAPUR, R., ROSS, D.E., and MERCER, M.R. 1992. The Roles of Controllability and Observability in Design for Test, Proceedings IEEE VLSI Test Symposium.
- CERNY, E., and MAURAS, C. 1990. Tautology checking using cross-controllability and cross-observability relations, Proceedings IEEE International Conference on Computer-Aided Design, pp. 34-37.
- EICHELBERGER, E.B., LINDBLOOM, E., WAICUKAUSKI, J.A., and WILLIAMS, T.W. 1991. *Structured Logic Testing*, Prentice Hall, Englewood Cliffs, N.J.
- FRITZEMEIER, R.R., NAGLE, H.T., and HAWKINS, C.F. 1989. Fundamentals of testability – a tutorial, *J IEEE Transactions on Industrial Electronics*, **36(2)**: 117-128.
- GOLDSTEIN, L.H., and THIGPEN, E.L. 1980. SCOAP Sandia Controllability / Observability Analysis Program, Proceedings 17th ACM / IEE Design Automation Conference, pp. 190-196.
- GRASON, J. 1979. TMEAS, A Testability Measurement Program, Proceedings 16th ACM / IEE Design Automation Conference, pp. 156-161.
- JERRY, G., and YOUJIN, Z. 1999. Tracking Software Components, Technical report, San Jose State University, USA, (<http://www.engr.sjsu.edu/gaojerry/report/main.htm>).
- KAPUR, R., BUTLER, K.M., ROSS, D.E., and MERCER, M.R. 1991. On Bridging Fault Controllability and Observability and Their Correlations to Detectability, Proceedings of The European Test Conference , pp. 333-339.
- KÖNEMANN, B. 1998. Creature from the Deep Submicron Lagoon, Keywords to the 10 th ITG Workshop on Test methodenund Zuverlässigkeit von Schaltungen, Herrenberg, Germany, March.
- KOREN, I. 1979. Analysis Of The Signal Reliability Measure And An Evaluation Procedure,” *J IEEE Transactions (Computers)*, **C-28 (3)**: 244-249.
- RATIU, I., SANGIOVANNI-VINCENTELLI, A., and PEDERSON, D.O. 1982. VICTOR: A Fast VLSI Testability Analysis Program, Proceedings International Test Conference, pp. 397-401.
- SAVIR, J. 1983. Good Controllability And Observability Do Not Guarantee Good Testability, *J IEEE Transactions (Computers)*, **C-32 (12)**: 1198-1200.
- SAVKIN, A.V. 1998. Controllability of Complex Switched Server Queueing Networks Modelled as Hybrid Dynamical Systems, Proceedings of 37th IEEE Conference on Decision and Control, pp 289-293.

REALIZATION OF A SIMPLIFIED CONTROLLABILITY COMPUTATION PROCEDURE

- SEISS, B., TROUBORST, P., and SCHULT, M. 1991. Test Point Insertion for Scan-Based BIST", Proc. International Test Conference, *IEEE CS Press*, pp. 253-262.
- STEPHENSON, J.E., and GRASON, J. 1976. A Testability Measure For Register Transfer Level Digital Circuits, Proceedings of 6th Annual Fault-Tolerant Computing Symposium, pp. 101-107.
- SUN, X., and SERRA, M. 1992. Merging Concurrent Checking and Off-line BIST, Proc. International Test Conference, Sept. 20-24 1992, Baltimore, MD, *IEEE CS Press*, pp. 958-967.
- WILLIAMS, R.W. 1986. IBM Perspectives on the Electrical Design Automation Industry, Keywords to IEEE Design Automation Conference.
- ZORIAN, Y. 1999, Testing the monster chip, *IEEE Spectrum*, **36(7)**: 54-60.
-

Received 7 June 2001

Accepted 3 December 2003