

A Graphics User Interface Simulink Model For Design for Testability

Vijesh Mokati ^{*1}, Rakesh Mandaliya²

^{*1,2} Department of Electronics and Communication, BMCT(Indore)

Abstract: Design for Testability is a method for testing each and every node in the design for structural and other faults. Higher the number of nodes which can be tested through the targeted number of patterns greater is the test coverage of the design. For this to be possible every node in the design has to be controllable and observable. MATLAB[®] Simulink[®] can be proved a fugitive tool in analysis the controllability and Observability. Here with the Simulink[®] Model the tedious job of calculating the formulas can be avoided.

Keywords: SCOAP, Controllability, Observability, Combinational component, Sequential Component, GUIDE GUI.

I. INTRODUCTION

The era of integrated chips have been evolved to such an extent that led to packages of increasing the complexity and size. The Design for testability refers to how a circuit is either designed or modified so that the testing of the circuit is simplified. Testability is achieved by using extra logic and/or test point. There are two key concepts in designing for testability: Controllability and Observability. The ability to apply input patterns to the primary inputs of a circuit to set up appropriate logic value at desired locations of a circuit is known as Controllability. The fault refers to physical defect in a circuit, the ability to observe the response of a fault on an internal node via the primary outputs of a circuit is denoted as Observability. The Sandia Controllability and Observability Analysis Program (SCOAP) introduced by Goldstein in 1979 is widely used for calculation of Controllability and Observability. In SCOAP, integers reflect the difficulty of controlling and observing the internal nodes. Here higher numbers indicate more difficult to control or observe.

II. MATLAB[®] GUIDE AND SIMULINK[®]

The MATLAB is high-performance language for technical computing integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. GUIDE, the MATLAB[®] Graphical User Interface Development Environment, provides a set of tools for creating graphical user interfaces (GUIs). These tools greatly

simplify the process of laying out and programming GUIs. It has many useful element like edit boxes, sliders, push button, graph axes etc. SIMULINK[®] software models, simulates, and analyzes dynamic systems. It enables you to pose a question about a system, model the system, and see what happens. SIMULINK[®] software is tightly integrated with the MATLAB[®] environment. Invoking of SIMULINK[®] model from MATLAB[®] GUIDE can be done in many ways and one of them is discussed here. The design of new custom user model and subsequently adding of its in library browser can be accomplished by using help in matlab[®] help at the heading “adding Libraries to the library browser. Similarly the new model can be created using the help in MATLAB[®] help under the heading “Creating a Subsystem by Grouping Existing Blocks”. The simulation of Simulink[®] model from GUI can be understand using the help[13].

III. COMBINATIONAL CONTROLLABILITY AND OBSERVABILITY ANALYSIS MODEL

Using the SCOAP, cited a number of gates and construct the corresponding Simulink[®] Models and arrange them in a user define library, my library, figure(1).

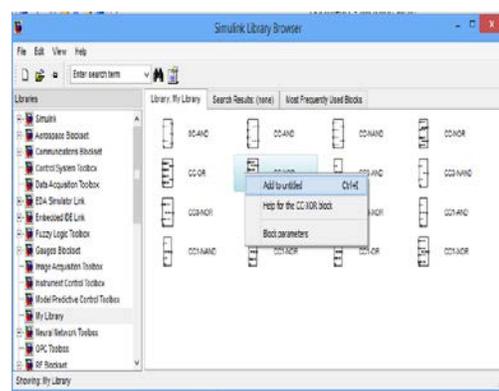


Fig.1: My Library

Here we develop custom model for different gates according to the formulas prescribed by[3]. We introduces the following definitions for calculation:

CC: Combinational controllability calculated as the number of elementary operations necessary for setting the circuit-specified unit to certain logical state.

CC0: Combinational controllability calculated as the number of elementary operations necessary for setting the circuit-specified unit to logical zero (0-Combinational controllability).

CC1: Combinational Controllability calculated as the number of elementary operations necessary for setting the circuit-specified unit to the logical unit (1-Cominational Controllability).

OC: Combinational Observability calculated as the number of elementary operations necessary for the logical state observation in a given circuit unit representing the output of a certain gate.

Here, the XOR gate has taken for example and its Simulink[®] model has been depicted in fig(2). and the analysis

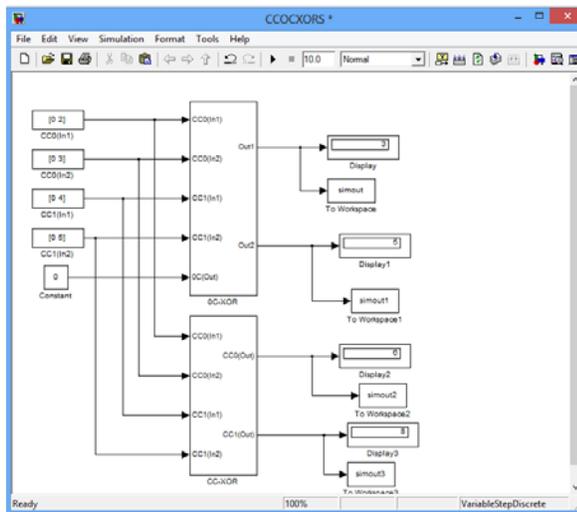


Fig.2: XOR ‘ CC and OC

formula for the XOR gate is as followed:

$$\text{Out} = \text{XOR}(\text{In1}, \text{In2}).$$

$$\text{CC0}(\text{Out}) = \min \{ \text{CC0}(0) + \text{CC0}(1), \text{CC1}(0) + \text{CC1}(1) \} + 1;$$

$$\text{CC1}(\text{Out}) = \min \{ \text{CC1}(0) + \text{CC0}(1), \text{CC0}(0) + \text{CC0}(1) \} + 1;$$

$$\text{OC}(0) = \min \{ \text{CC0}(1), \text{CC1}(1) \} + \text{OC}(\text{Out}) + 1;$$

$$\text{OC}(1) = \min \{ \text{CC0}(0), \text{CC1}(0) \} + \text{OC}(\text{Out}) + 1;$$

Similarly, other Gate formulas can find on[3]. The input and output for XOR gate can be vied on

MATLAB[®] GUIDE GUI as shown in the figure(3).



Fig.3: GUI Intreface

The result extended to combinational circuit shown the figure (4).

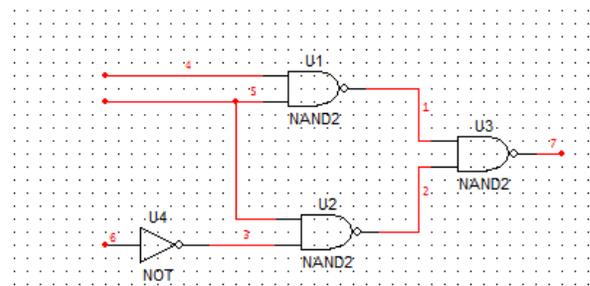


Fig.4: Combinational Circuit

The resultant Simulink[®] model for the combinational circuit have drawn with the help of user defined block of NAND and NOT gate and input have been given with the help of GUI and consequently result evaluated in figure(5).

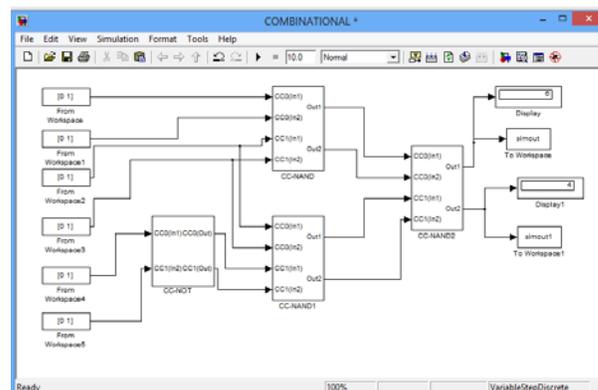


Fig.5: CC and OC for Combinational Circuit

IV. SEQUENTIAL CIRCUIT CONTROLLABILITY AND OBSERVABILITY ANALYSIS MODEL

The Sequential circuits are differ from the Combinational circuit in the timing and presence of memory element. The assessment of Testability parameters for sequential circuit shown in the figure(6) ,which composed of two FF1,FF2 flip-flops with corresponding feedbacks (from the FF1 flip-flop output to G4,G5 gate inputs and from the FF2 flip-flop output to the G2 gate input).

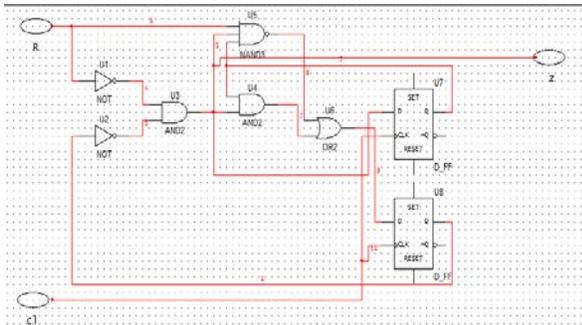


Fig.6: The Sequential circuit

The feasible method of testability parameter calculation for such a circuit is its transformation into a combinational circuit as shown in the figure(7)

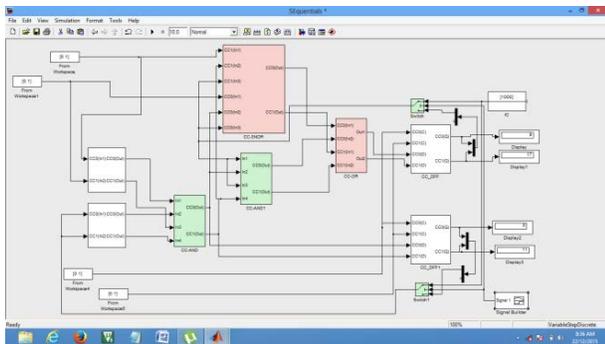


Fig.7:Calculation of Sequential CC and OC

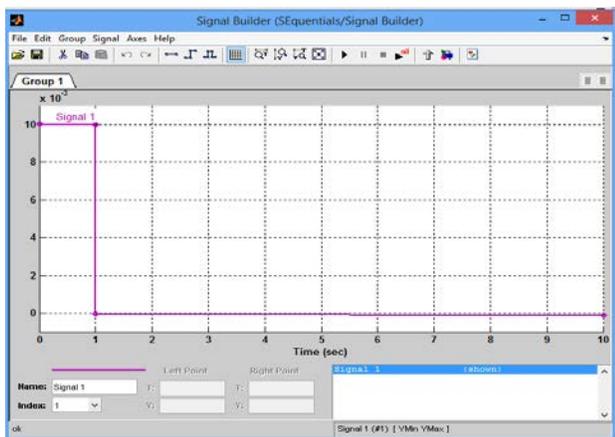


Fig 8:Signal Builder Output

by eliminating the flip-flops and adding a corresponding number of primary inputs and primary outputs instead of broken feedback. The Signal Builder block allows you to create interchangeable groups of piecewise linear signal sources and use them in a model, which used in our sequential model simulation and depicted in the figure(8).

V. CONCLUSION

The system level design solution provide by Simulink and it interface with GUIDE GUI, produce result as per required. The Simulink® provides top-down design clear-cut specification design and test for the entire system prove early detection faults and in this case design can be verified with the help of Graphics user Interface without going to minute detail for design. The Combinational Controllability and Observability are proven technology and their simulation and verification at early stages of design put the designer in better position and with the help of Simulink it can be proven.

REFERENCES

- [1] Parag K. Lala, Introduction to logic Circuit Testing.
- [2] Bushnel and Agrawal, ESSENTIALS OF ELECTRONIC TESTING FORDIGITAL, MEMORY AND MIXED-SIGNAL VLSI CIRCUITS.
- [3] Evgeni Perelrouzen, Digital Integrated Circuits: Design-for-Test Using Simulink® and Stateflow®.
- [4] MATLAB® The language of Technical Computing.
- [5] UNIVERSITY OF NEWCASTLE UPON TYNE, Matlab/Simulink Tutorial .
- [6] Patrick Marchand, O.Thomas Holland, Graphics and GUIs with MATLAB®
- [7] Harold Klee, Randal Allen, Simulation of Dynamic System with MATLAB® and Simulink®.
- [8] Todd Wittman, Building a Matlab GUI.
- [9] Vikram A Bose-Mullick, Matlab6: GUI Tutorial
- [10] Experiment5: GUI & SIMULINK ON web.
- [11] Yasir Moshe, Advanced MATLAB and GUI.
- [12] R.Campos-Rodriguez, M.Alcaraz-Mejia, AMATLAB/Simulink Framework for the design of Controllers and Observers for Discrete-Event Systems.
- [13] Dr.S.S.Limye, MATLAB® GUI Tutorial.
- [14] Xilinx, Simulink® Basics
- [15] Library, Tutorial, Christopher Lum
- [16] David Harris, Introduction to CMOS VLSI Design, Lecture 17: design for Testability