Application of Binary Decision Diagram in digital circuit analysis.

Jyoti Kukreja

University of Southern California
For – Dr. James Ellison

Abstract:

Binary Decision Diagrams (BDDs) are one of the biggest breakthroughs in CAD in the last decade. BDDs are a *canonical* and *efficient* way to represent and manipulate Boolean functions and have been successfully used in numerous CAD applications

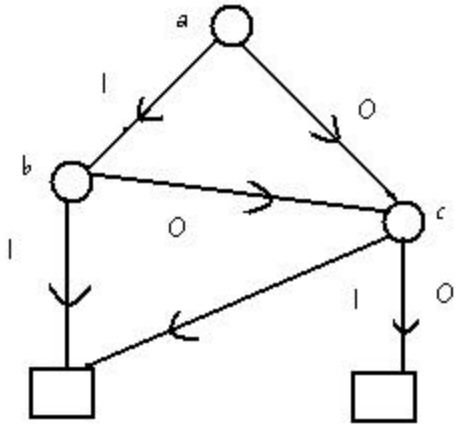
Introduction

The design and synthesis of digital circuits has gained importance in both the industrial and academic worlds. Timed circuits are a class of asynchronous circuits that incorporate explicit timing information in the specification. This information is used throughout the synthesis procedure to optimize the design. Meeting timing requirements is an important constraint imposed on highly integrated circuits, and the verification of timing of a circuit before manufacturing is one of the critical tasks to be solved by CAD tools. Several algorithms to analyze and speed up gate level timing simulation have been proposed in various research papers. Some propose gate delay analysis and some path delay analysis. Just as in case of asynchronous circuits, delay analysis of synchronous circuits is also becoming a region of active research.

Binary decision diagrams (BDD) form an integral and important part of such researches and theories. In the following pages I have tried to analyze various theories for digital circuit delay analysis and the role played by BDD (in the form of reduced order binary decision diagrams, (ROBDD) [Ref #7], structurally synthesized binary decision diagrams (SSBDD) [Ref #4] Structural Binary Decision Diagrams (SBDD) [Ref #1] Ordered Binary Decision Diagrams (OBDD) [Ref # 3] for the same

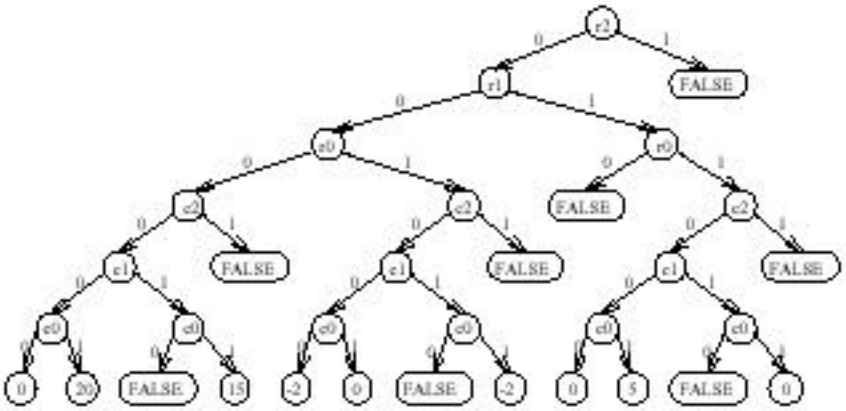
1.1 Overview of Binary Decision Diagrams:

A binary decision diagram is a directed acyclic graph in which every node represents some Boolean function. There are exactly two outgoing edges for every non-terminal node. There are only two terminal nodes representing the constants 1 and 0



BDD for the function F(a,b,c)=a.b+c

On the other hand, MTBDDs are a type of BDD which allow terminal nodes to contain data



MTBDD

1.2 Delay faults and their analysis

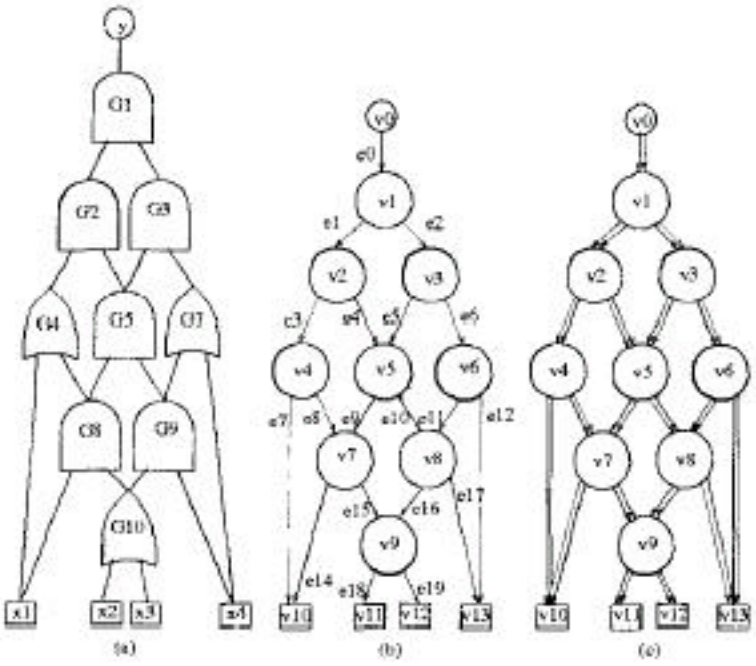
A delay fault is a fault that causes incorrect data to be latched into a memory element or appear at an output. The maximum allowable path delay in a synchronous circuit is determined by its clock rate. If the delay on a path of a manufactured circuit exceeds the time period of the clock, erroneous values may be latched at the output. The objective of the delay testing is to ensure that the manufactured circuit operates correctly at the functional clock rate.

Two common delay faults models are the gate and line/path delay fault models. These have been investigated for test generation and fault simulation. The path delay model is capable of modeling distributed failures resulting due to statistical variation in the manufacturing process. It is thus very critical for circuit designing using statistical timing analysis.

The main problem with path delay fault model is that there can be an exponential number of path faults in a circuit. Thus making it practically impossible to enumerate all paths for the purpose of test generation and fault simulation

1.3 OBDD for path delay computation

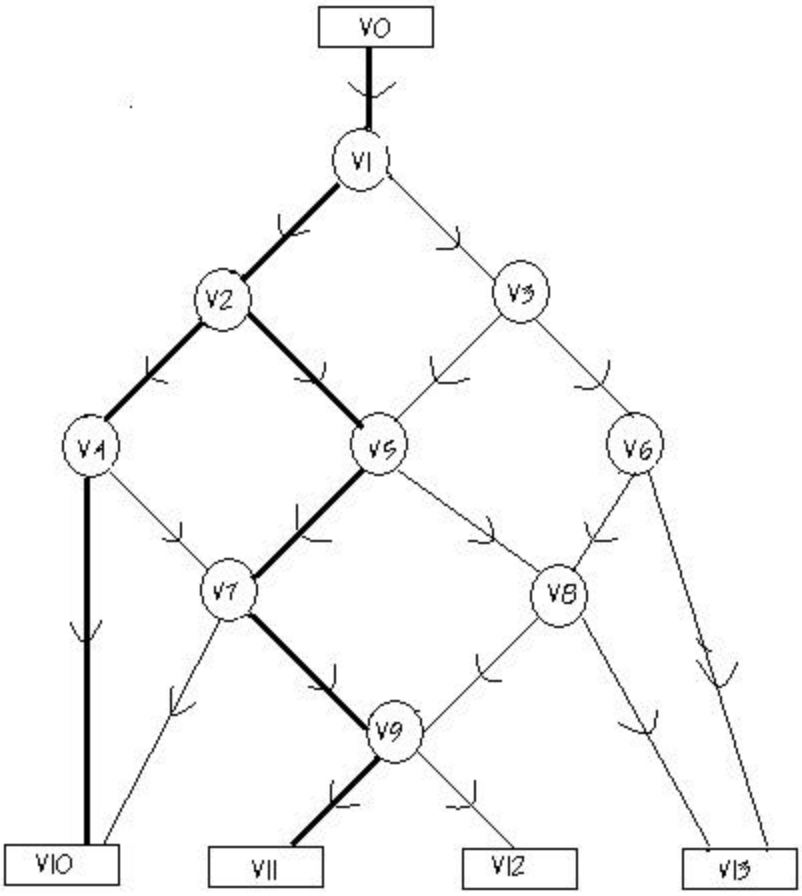
In [Ref # 3], the researchers have represented the combinational circuit as a directed acyclic graph (DAG).
"In a DAG, the gates correspond to vertices and the wires correspond to edges in the graph. Such a modeling is as shown below.



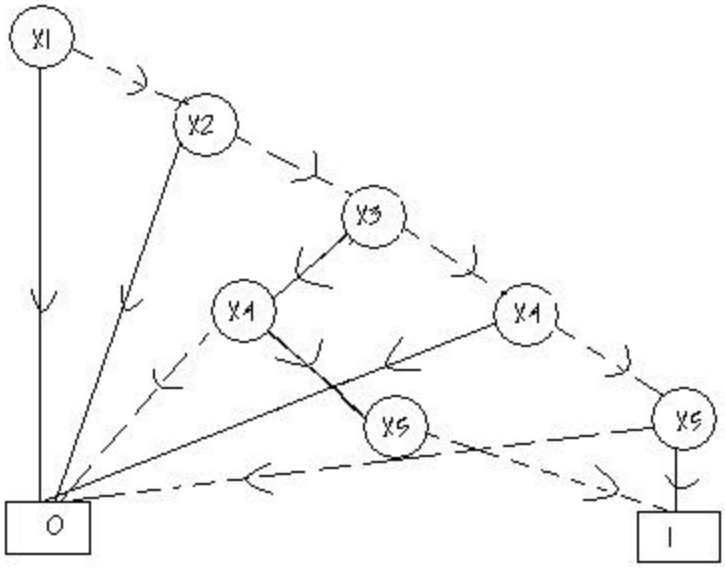
DAG Representation

A set of tested path delay faults can then be represented using an OBDD of the corresponding Boolean functions. Symbolic representation of path delay faults can achieve high degree of compaction relative to more explicit forms. "OBDD provide

convenient data structure to represent these large number of path delays during the process of fault simulation which computes the path delay-fault coverage for a given delay test-set."[Ref # 3] --> **Advantages of OBDD**



A sequence of delay tested paths



OBDD for the above sequences

Thus circuit-→ Dag→ OBDD conversion technique is used for path delay analysis in [Ref # 3]. From the study of the above technique, it can be concluded that Ordered binary decision diagrams are a useful representation of Boolean functions, if a good variable ordering is known. Variable orderings are computed by heuristic algorithms and then improved with local search and simulated annealing algorithms.

1.4 MTBB Application

In his PhD thesis, Wendy A. Belluomini discusses, how representing the timed state space using implicit methods produces very significant memory savings when applied to large examples. The use of MTBB optimization is most apt for large examples which run out of memory using explicit approach. [Ref #9]

1.5 BDD for test generation for path delay faults

A new test generation technique for path delay faults in circuits employing scan/hold type flip-flops is presented in [Ref # 7]. Reduced ordered binary decision diagrams (ROBDDs) are used to represent Boolean functions. Here, we consider 2 time frames. These frames constitute a transition. A pair of constraint function corresponding to these 2 time frames are evaluated. If the constraint function in the second time frame is a non-zero, then robust- hazard-free-test generation for the delay is attempted. If a robust test cannot be found, the existence of a non-robust test is checked. When Boolean algebraic manipulation of the constraint function is done, it's concluded that in the absence of robust or non-robust tests, the fault is undetectable. This method is applicable to all circuits that are capable of being to analyzed using ROBDDs" [Ref # 7]

1.6 Timing simulation with BDD

In [Ref # 4], The researchers have proposed a gate level timing simulation where, in-order to speed up gate timing simulation, instead of gate delays, path delays for tree like sub circuits (macros) have been used. These macros are represented by structurally synthesized binary decision diagrams (SSBDD). SSBDD help in fast computation delays in macros. Of the various delay models, i.e. Zero delay, unit delay and multiple delay model, the researchers have adopted the multiple –delay model.

This is because, zero delay models are best used to analyze combinational circuits without memories, and unit delay models to verify logical behavior of synchronous sequential circuits. "For timing behavior, a multiple delay model should be used. In such a model, each circuit element is assigned a delay which is an integer multiple of a time unit. [Ref # 4]

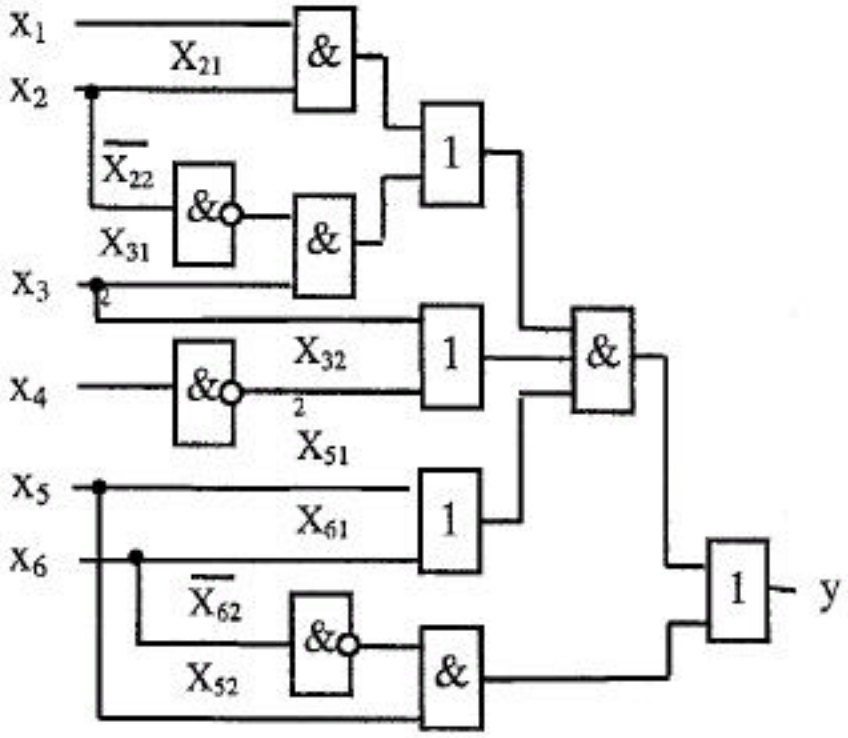
The advantage of using a macro-level simulation is that when representing complex gates by macros, the number of macros is equal to the number of tree-like sub circuits in the complex gate. In this paper, it has been graphically depicted that macro level simulation is far simpler and takes much lesser simulation in comparison to gate level simulation. In [Ref # 1] and [Ref # 4], the digital circuit is represented by Equivalent Parenthesis form (EPF). "For synthesizing the EPF for a given circuit, numbers are first assigned to the gates and letters to the nets. Then starting at an output and working back towards primary inputs, EPPF replaces individual literals by products or sums of literals. When an AND gate is encountered during backtracking, a product term is created in which the literals are the names of nets connected to the inputs of the AND gate. An OR gate causes a sum of literals to be formed, whole encountering an inverter causes a literal to be complimented" [Ref # 1].

1.7 Multivalued simulation of digital circuits using SBDD

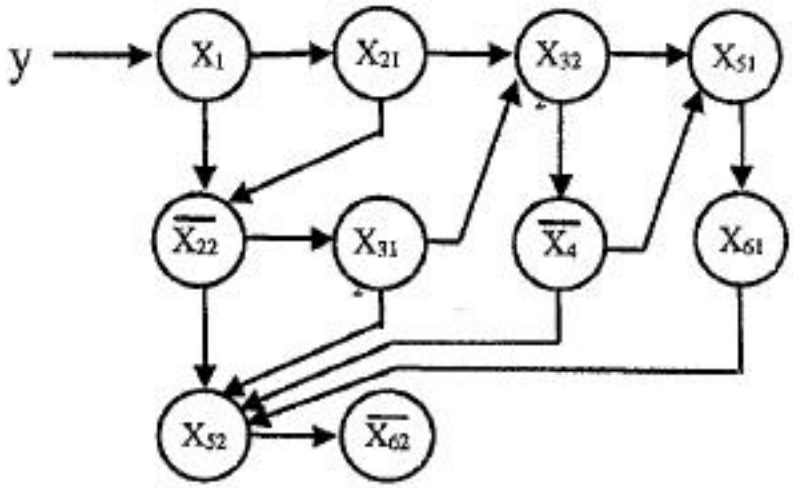
The Structured Binary Decision Diagram (SBDD) is used for the purpose of calculation of maximum of Boolean derivative on SBDD. This forms the basis of multivalued simulation. R Ubar in [Ref #1] discusses how multivalued simulation has its advantages over two-value simulation of digital circuits.

The following figure shows a representation of a combinational circuit described by the equivalent parenthesis for (EPF)

$$y = (x_1\, x_{21} + \neg x_{22}\, x_{31})\,(x_{32}\, x_{51} + \neg x_4\, x_{61}) + x_{52}\neg x_{62}. \quad (3)$$



EPF Representation

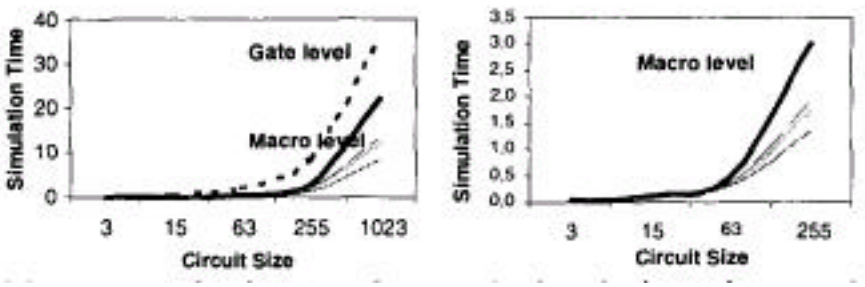


Corresponding SBDD

A one is to one correspondence exists between the nodes and the paths in the circuits in the above figure; the SBDD contains 10 nodes, whereas each of them represents a signal path in the circuit or a literal in the EPF. In this paper [Ref #1], Mr. S Ubar explains how for multivalued simulation, a procedure based on calculation of Boolean Derivatives on SBDD is adopted. This helps in delay fault analysis, hazard detection or dynamic test analysis of the digital circuit.

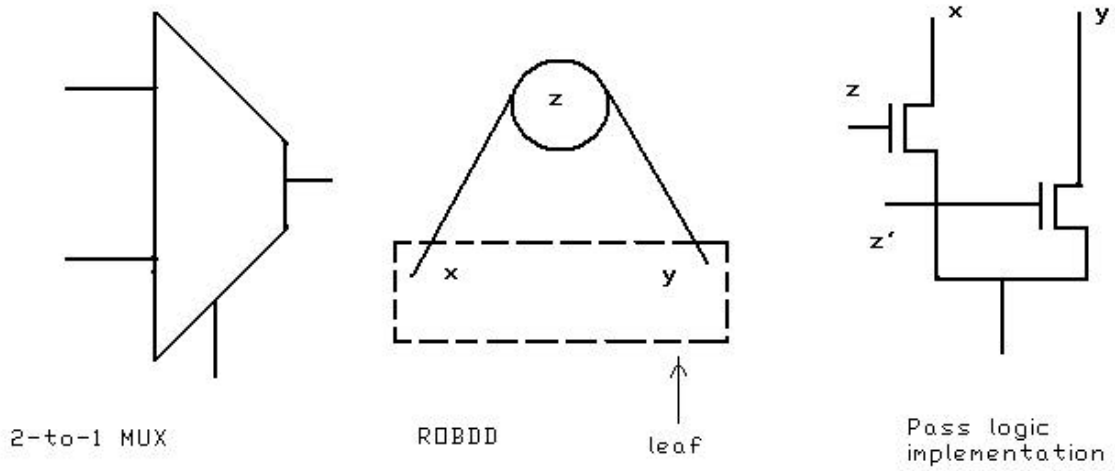
Advantages of SBDD What is of most interest to us is the fact that how introduction of SBDDs reduced the complexity of the model by replacing low level two-input gate networks with higher macro-level circuits. “Experimental results prove that macro-level simulations are 3,29 times faster than gate-level simulations. For macro-level simulations there is no need to create a separate model for each macro. Instead, from the gate level description automatically an SBDD model is created and a single general procedure can be used for all types of macros” [Ref #1].

The observations in [Ref # 4] are very similar to those mentioned above. Juts like in the above case, gate level timing simulation is sped up by replacing gate delays with delays for macros (represented by SSBDD). Here it was observed that using SSBDD for representing macros avoids exponential explosion as the complexity of the model increases. The macro level timing simulation was observed to be 3,54 times faster as compared to gate-level simulation.



2.1 ROBDD for synthesis of low power digital circuits.

It is a well-known fact that in order to achieve viable and competitive products, power dissipation becomes a dominant issue. In [Ref # 2] the authors address the issue of realizing logic functions based on pass logic which lead to power saving. Logic functions are synthesized using ROBDD. The researchers used the base function depicted below as a basic building block. This way, extremely complex cells were developed limited to a maximum of three pass transistors in series.



Base Function

Conclusion:

Hence, from the above discussion, it is clearly observed that BDDs and its various modifications (ROBDD, SSBDD, SBDD, MTBDD) are in some way or the other involved with the path delay fault analysis, functional representation, power dissipation reduction, hazard detection and dynamic test analysis.

This is primarily because BDD are one of the most compact data structures known to represent Boolean functions. As for ROBDD, there exists a unique ROBDD for each Boolean function once the variable ordering is determined. The advantages of SBDD have been discusses in the previous pages.

These advantages justify the continued interest of researchers in this depiction technique for analysis to digital circuits.

REFERENCES

Ref #1 **Multi-valued simulation of digital circuits**
Ubar, R.
Microelectronics, 1997. Proceedings., 1997 21st International Conference on , Volume: 2 , 1997
Page(s): 721 -724 vol.2

Ref #2 **Synthesis of low power CMOS digital circuits based on fully restored pass logic**
Al-Khalili, D.; Kechichian, K.; Liqin Dong
Electrical and Computer Engineering, 1996. Canadian Conference on, Volume: 1, 1996
Page(s): 142 -145 vol.1

Ref# 3 **Computing exact path delay-fault coverage using OBDDs**
Kapoor, B.; Nair, V.S.S.
Circuits and Systems, 1994., Proceedings of the 37th Midwest Symposium on , Volume: 1 , 1994
Page(s): 234 -237 vol.1

Ref #4 **Timing simulation of digital circuits with binary decision diagrams**
Ubar, R.; Jutman, A.; Peng, Z.
Design, Automation and Test in Europe, 2001. Conference and Exhibition 2001. Proceedings , 2001
Page(s): 460 –466

Ref #5 **Path delay fault testability analysis**
Sosnowski, J.; Wabia, T.; Bech, T.
Defect and Fault Tolerance in VLSI Systems, 2000. Proceedings. IEEE International Symposium on, 2000
Page(s): 338 -346

Ref #6 **The evolution of ROBDDs: preliminary results and a first analysis**
van Remortel, P.; Lenaerts, T.; Manderick, B.
Evolvable Hardware, 1999. Proceedings of the First NASA/DoD Workshop on, 1999
Page(s): 249 –254

Ref #7 **Test generation for path delay faults using binary decision diagrams**
Bhattacharya, D.; Agrawal, P.; Agrawal, V.D.
Computers, IEEE Transactions on, Volume: 44 Issue: 3 , March 1995
Page(s): 434 –447

Ref #8 I**BDDs: an efficient functional representation for digital circuits**
Jain, J.; Abadir, M.; Bitner, J.; Fussell, D.S.; Abraham, J.A.
Design Automation, 1992. Proceedings., [3rd] European Conference on, 1992
Page(s): 440 -446

Ref # 9 Algorithms for synthesis and verification of timed circuits and systems
Wendi A. Belluomini, Department of Computer Science, University of Utah, 1999