ACIT'2017
The International Arab Conference on Information Technology
Yassmine Hammamet, Tunisia
December 22-24, 2017

# Simplified Digital Logic Circuits Using Binary Decision Tree

Hamed A. Fawareh and Abla Suliman Hussien
ZarqaUniversity/ software Engineering Department, Zarqa, Jordan

***Abstract*—**Ordered Binary decision tree (OBDT) is a graphical representation which looks like a tree with root and branches; it played a key role in digital circuits verification and manipulation which leads to a compact circuit after elimination many redundant branches. In this paper a Boolean equation was created for the proposed digital circuit, and with the aid of Boolean Algebra fundamentals, the proposed Boolean equation was simplified and reduced in size, afterword a truth table was constructed from which a Binary Decision Tree was sketched, then followed with a procedure of reducing and eliminating redundancy in the branches, a final schematic BBD will be constructed, two case studies were presented for verification of this technique.

***Index Terms*—***binary decision tree (BDT), digital logic circuit and systems, binary decision diagram (BDD, Binary Algebra*

## I. INTRODUCTION

Binary decision tree (BDT) as shown in Fig.1 is a schematic structure of logic system. BDT is comprises a relationship between outputs (0 and 1) and inputs (x1, x2) which could be characterized with Boolean function and by utilizing different Boolean algebra techniques. It is possible to reduce and simplify the Boolean formula of the system to a reduced form that meets the same relationship between inputs and outputs.

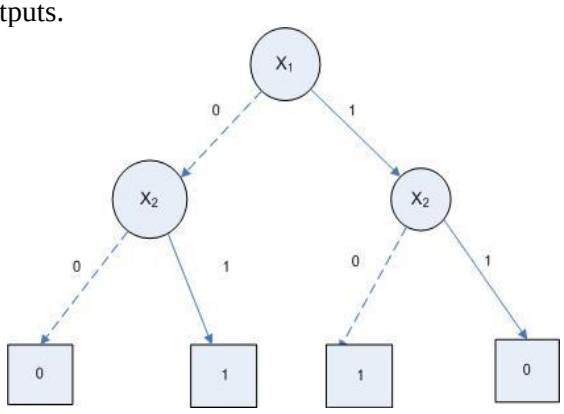


Fig.1Binary Decision Tree (BDT)

After reduction and simplification of BDT, Fig 2 represents the achieved short structure which is called Binary Decision Diagram (BDD). BDD is used to symbolize a digital circuit with Boolean equivalence formula. The procedure for BDD will be detailed in steps later in this paper.

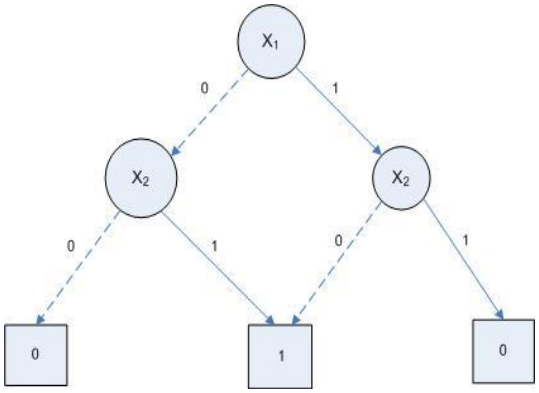


Fig.2. Binary Decision Diagram (BDD)

BDDs are logic representation structures diagram first introduced by Lee [9] and Akers [10]. Also, BDDs is one kind of Graphic-Based Algorithm used for the purpose of designing, verifying and manipulating of digital systems which supported by Boolean function. The top of the diagram is the root and the bottoms are the leaves which are labeled in term of 0 or 1.If the given variables follow linear order x1<x2<…. <xn in all paths through the graph. It will be called Ordered Binary Decision Diagram (OBDD) [1]. Reduced Order Binary Decision Diagram (RODBB) is a smaller version of OBDD after exposed to a procedure of removing and eliminating redundancy for the purpose of achieving a unique graph that characterized with the same input output relationship of the original OBDD.

There are two main differences between BDT and BDD, BDT allow redundant in testing Boolean variable, while BDD is a simplified version where Boolean variable from the high level that has the same removed value [2].

## II. RELATED WORK

There are many problems and cases have been explained and solved using BDD due to its simplicity and its direct relationship between root and branches of the diagram but that does not mean BDD is the

perfect choice for such a problem like 8 Queens puzzle, the computational digital circuits and its equivalence circuits are such problems that have been solved using ROBDD [2]. BDDs also used in verification, manipulation and circuit testing for the purpose of avoiding the problem of exponential explosion of nodes which produced when we want to find good variable orderings.

Jacobi and Calazans proposed an incremental method to reduce the number of nodes which speeds up the algorithms of BDDs for circuits testing and evaluation [3].

BDD's are also used in reliability analysis. For the purpose of providing an efficient means of analyzing a system especially in fault tree analysis. But there are many problem in analyzing strategies that increase the likelihood of getting a DBB from fault tree; especially in ordering the a basic events (variable) which will effect to the size of the minimal BDD with few nodes in the result. There's no global scheme that produced a BDD for all fault tree. REAY and ANDREWS in their paper [4] proposed an efficient and accuracy ordering scheme based on neural network to select an appropriate schema for individuals fault tree module.

BRYANT surveyed the most important application that has been solved using OBDD standards [5].

Bryant proposed a reduced, ordered binary decision diagram and defines it as a canonical representation that could be characterized by a Boolean function, switch level of MOS circuits was the application that implemented which guided toward the initial development of BDD [10-11].

Luca researched the Biconditional Binary decision diagram (BBDD) and put into practice on electric design automation (EDA), in which BDD improves the efficiency of EDA task [13].

In this paper we used a BDD to represent digital logic circuits and then reducing and eliminating redundancy in the branches to reduce amount of count. The method will verify the correctness and increase the efficiency in using the digital circuits. The research proposed in this paper started by writing Boolean equation for a proposed digital circuit in section three, in section four, two case studies were presented for verification finally we concluded the paper.

## III. BOOLEAN ALGEBRA FUNDAMENTALS

Boolean algebra was found to be useful technique to design and analyze digital systems and circuits, logic circuits in logic controls and computer logic systems [15, 16, 18]. Engineers and technicians can utilize these methods to model any complicated logic control systems that could be represented by single Boolean equation[14, 17]. In this paper we used Boolean equation in addition to binary decision tree to simplify the digital circuit for the proposed case studies. Table .1 shows Boolean Algebra Functions that we used in circuit simplification [8].

Table .1 Boolean Algebra Functions

| No. | Boolean Algebra Function |
|---|---|
| 1. | $A + 0 = A$ |
| 2. | $A + 1 = 1$ |
| 3. | $A.0 = 0$ |
| 4. | $A.1 = A$ |
| 5. | $A + A = A$ |
| 6. | $A + \bar{A} = 1$ |
| 7. | $A\ .\ A = A$ |
| 8. | $A\ .\bar{A}\ = 0$ |
| 9. | $\bar{\bar{A}} = A$ |
| 10. | $A + AB = A$ |
| 11. | $A + \bar{A}B = A + B$ |
| 12. | $(A + B)(A + C) = A + BC$ |

The paper strategy used to simplify the digital circuit is divided into four main steps:

1- Steps to construct BDD from Digital Circuits: In this step, we used Boolean algebra fundamentals to produce, the proposed Boolean equation the detail is shown in section IV.

2- Digital circuits, Boolean Function Simplification: in this step the equations are simplified and reduce in size of digital circuit.
3- Digital circuits, Boolean Function Simplification:in this step we constructed from which a Binary Decision Tree will be sketched.
4- Reduced Circuit and Truth Table: the final step a procedure of reducing and eliminating redundancy in the branches, a final schematic BBD will be established.

In the following two secton we will explain the four steps in details with two case studies.

## IV. First Case Study:

The digital circuit shown in Fig 3 is composed of two OR gates, two NOT gates and one AND has been tested where A, B, C are inputs and Y is the output, T1 is the output of the upper OR gate and T2 is the output of the lower OR gate. This study will present the theory followed by the four steps provided in the previous section. The simplification strategy detailed as follows:

### *A. Steps to construct BDD from Digital Circuits*

The following procedure will be put in to action to accomplish the design of BDD diagram:

1. Building the proposed digital circuit under consideration for the case study
2. Writing the Boolean equation for the circuit which represent the relation between input and outputs
3. Reducing the resulted Boolean formula using Boolean Algebra represented in table 1
4. Writing the Truth table for the reduced formula where A,B,C are input and Y is the output
5. Drawing the resulted Binary decision tree structure
6. Reducing the tree structure size by eliminating any redundancy
7. Drawing the final structure, which will represent the Binary Decision Diagram

### *B. Digital circuits, Boolean Function Simplification*

The circuit shown in Fig.3 is composed of 5 gates (two NOT gates, two Or gates and one AND, where A, B ,C are the inputs and Y is the output.

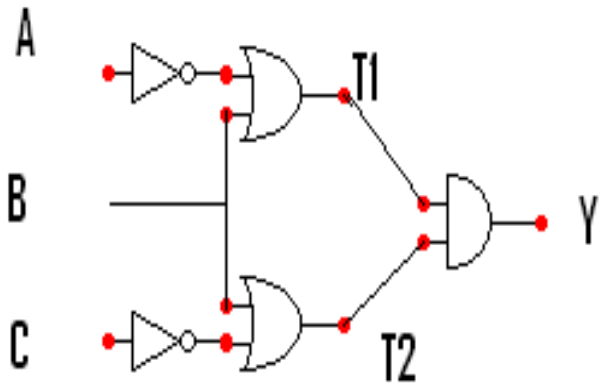


Fig.3 Digital Circuit before Simplification

Boolean function of the digital circuit shown in Fig.3.the digital circuit is represented by the following simplification steps starting from equation 1 to equation 8.

$$T1 = \bar{A} + B \qquad \text{Equation (1)}$$

$$T2 = B + \bar{C} \qquad \text{Equation (2)}$$

$$Y = T1.T2 = (\bar{A} + B).(B + \bar{C}) \qquad \text{Equation (3)}$$

$$Y = \bar{A}.B + \bar{A}.\bar{C} + B.B + B.\bar{C} \qquad \text{Equation (4)}$$

Since $B.B = B$, (Function.7of Section 3 Table.1), Then

$$Y = \bar{A}.B + \bar{A}.\bar{C} + B + B.\bar{C} \qquad \text{Equation (5)}$$

Factor $B$ from the first, third and the fourth terms, we get

$$Y = B(\bar{A} + 1 + \bar{C}) + \bar{A}.\bar{C} \qquad \text{Equation (6)}$$

$$\text{Since } 1 + \bar{C} + \bar{A} = 1 \qquad \text{Equation (7)}$$

In Boolean algebra, 1 plus any other quantity will yield to 1 always.

Then, after substitution of formula 7 in 6, the resulted formula is:

$$Y = B + \bar{A}.\bar{C} \qquad \text{Equation (8)}$$

Equation 8 is the simplified and reduced formula for the proposed digital circuit shown in Fig. 3 utilizing Boolean algebra reduction procedures and techniques, it is remarkably compact and practically easier to be implemented. Fig 4 is the simplified digital circuit represented by Equation 8.

### C. *Reduced Circuit and Truth Table*

The equivalent circuit diagram of the reduced formula of Fig.3 is shown in Fig.4.

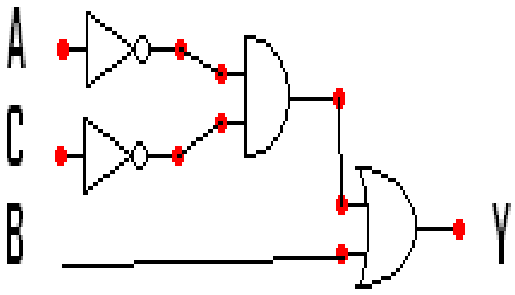


Fig.4 Digital circuit After Simplification

As shown in Fig.4, the circuit has been simplified and the circuit size was reduced for better digital function but both circuits should satisfy the same relationship between input and output except that Fig.4 is reduced size of Fig.3. Table.2 is the truth table of the Equivalent Reduced Digital Circuit of Fig.4, as represented by Equation 8.

Table.2 Truth Table of Equation 8

| **input** | | | output |
|---|---|---|---|
| **A** | **B** | **C** | $Y = B + \overline{A}.\overline{C}$ |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

### D. *Binary Decision Tree and Reduction steps*

Diagram shown in Fig.5 is the BDT as derived from the truth table shown in Table .2

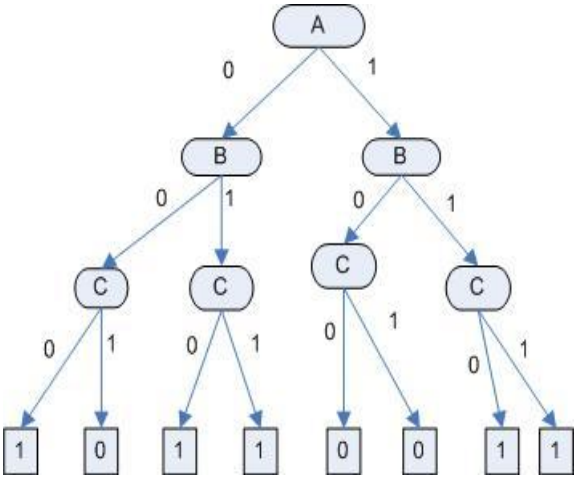


Fig. 5. Ordered Binary Decision Tree of table 2

Fig 6 represents the procedure of reducing and eliminating redundancy existed in the branches of BDD diagram shown in fig 5.

Diagram shown in Fig.7-a and Fig.7-b signify necessary steps toward reduction of the tree by reducing and elimination any redundancy and similarity branches of the tree. Where Fig.8 is the final reduced diagram BDD, where the root is A and the leaves are 0 and 1

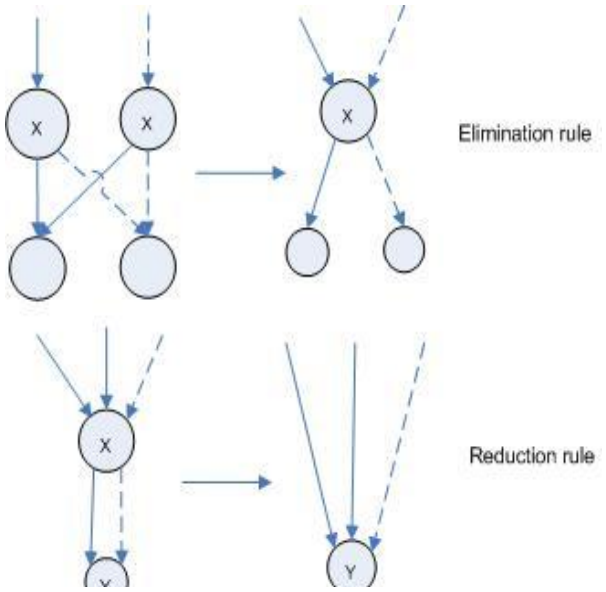


Fig. 6 Elimination and reduction rules

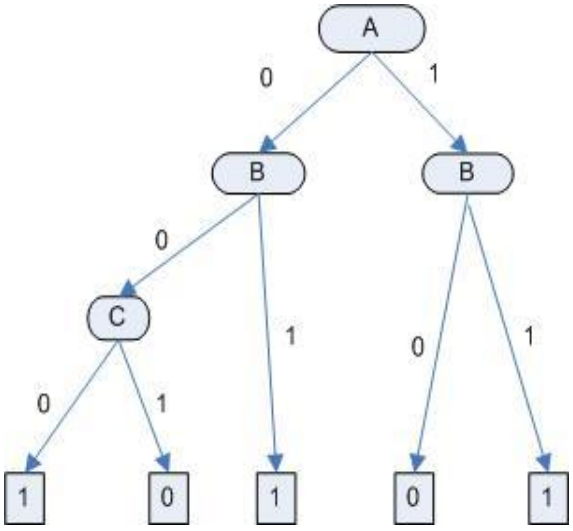

Fig.7-a Reducing and Elimination Redundancy

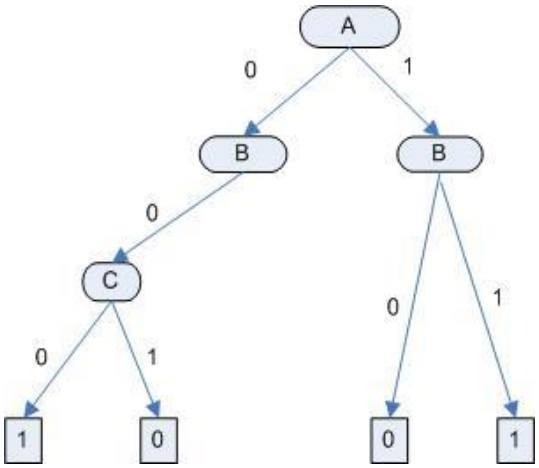

Fig.7-b Reducing and Elimination Redundancy

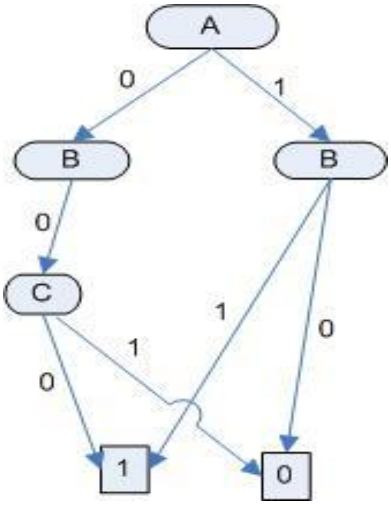

Fig.8 Final reduced diagram BDD

## V. SECOND CASE STUDY

The digital circuit shown in Fig.9 is a second digital circuit example which composed of 3 NOT gates, one OR gates with three inputs and one OR gate with two inputs and two AND gates .

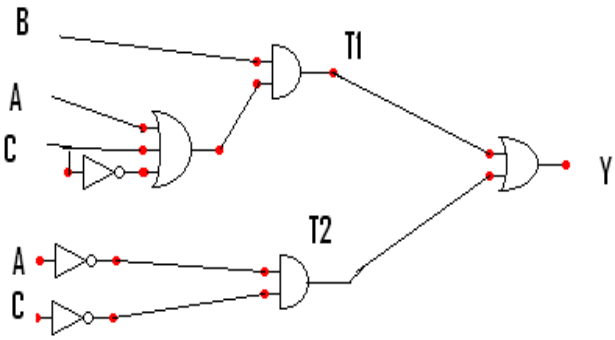


Fig.9 Digital Circuit before Simplification

The simplification steps of this case study are shown in equation 9 to equation 13.

$$T1 = B.(C + \bar{C} + A) \qquad \text{Equation 9}$$

Since $C + \bar{C} = 1$, (using Equ.6 of Table.1), then

$$T1 = B(1 + A) \qquad \text{Equation 10}$$

Since $1 + A = 1$ using Equ 2 of Table.1), therefore

$T1 = B$ Equation 11

$T2 = \bar{A} . \bar{C}$ Equation 12

Then the overall result is

Y = T1 + T 2 = B +$\bar{A} . \bar{C}$ Equation 13

The output simplified Boolean formula for the Circuit shown in Fig.9 is the same as the simplified Boolean formula of circuit shown in Fig. 3, this means that might be there are many circuits that could lead to the same Boolean formula.

## VI. Conclusion

Manipulation of binary decision Diagram and its connection to Boolean algebra and digital circuit has been presented and discussed with two case studies, where simplification of digital circuit and drawing its equivalent Binary decision tree was accomplished with the assistance of the reduced digital circuit truth table. Reduction and elimination process of redundancy in the BDD tree was performed for Ordered Binary decision Diagram.

In Digital circuits, simplifications procedure using Boolean formulas could help reduce large circuits with big number of gates to small digital circuit with limited number of gates, BDD is another alternative way of reducing diagram that lead to smaller one which represented by Boolean Functions. As seen equation 8 and equation 13 looks like the same, this means that digital circuit of case study I and digital circuit of case study II yield to the same digital Boolean function but with different digital circuit combinations, both circuits could lead to the same BDD diagram.

In practical point of view, Boolean function should be established first which could lead toward BDD construction.

## References

[1] HenrikReif Andersen, "An Introduction to Binary Decision Diagrams"1997.
[2] Frank Pfenning, "Lecture Notes on Binary Decision" 2010.SA
[3] R. Jacobi, N. Calazans, C. Trullemans"Incremental Reduction of Binary Decision Diagrams" TELKOMNIKA Indonesian Journal of Electrical Engineering (1991).
[4] REAY, K.A. and ANDREWS, J.D. "A fault tree analysis strategy using binary decision diagrams" Reliability Engineering and System Safety, 78(1), pp.45-56, 2004.
[5] Randal E. Bryant, "Symbolic Boolean Manipulation with Ordered Binary-Decision mKliagrams"
[6] VP Nelson, HT Nagle, JD Irwin "Logic Circuit Analysis and Design" - New Jersey, USA: Prentice-HallDigital(1995)
[7] David Money Harris and Sarah L. Harris "Digital Design and Computer Architecture" (2013)
[8] C Roth Jr, L Kinn"Fundamentals of logic design" (2013)
[9] C. Y. Lee, "Representation of switching circuits by binary-decision programs," *Bell Syst. Tech. J.*, vol. 38, no. 4, pp. 985–999, 1959.
[10] S. B. Akers, "Binary decision diagrams," *IEEE Trans. Comput.*, vol.100, no. 6, pp. 509–516, 1978.
[11] R. E. Bryant. Symbolic manipulation of Boolean functions using a graphical representation, In *Proceedings ofthe 22nd Design Automation Conference*, June 1985.
[12] R. E. Bryant. Graph-based algorithms for Boolean function manipulation, IEEE *Transactions on Computers*,C-35(8):677–691, August 1986.
[13] P. E. Gaillardon ; G. De Micheli, "Biconditional Binary Decision Diagrams: A Novel Canonical Logic Representation Form, Ieee Journal On Emerging And Selected Topics In Circuits And Systems, Vol. 4, No. 4, December 2014 .
[14] JashanKoshal, Monark Bag "Cascading of C4.5 Decision Tree and Support Vector Machine for Rule Based Intrusion Detection System" I. J. Computer Network and Information Security, Volume 4 No. 8 (2012)
[15] Sahadev Roy ChandanTilakBhunia, "On Synthesis of Combinational Logic Circuits" International Journal of Computer Applications, Volume 127 – No.1, ( 2015)
[16] Vijayakumari C K, Mythili P, Rekha K James, "Genetic Algorithm Based Design ofCombinational Logic Circuits using ReedMuller blocks", Proceedings of the World Congress on Engineering,Vol IWCE 2015, July 1 - 3, 2015, London, U.K.
[17] B.Reshma Yusuf, Dr.P.Chenna Reddy, "Mining Data Streams using Option Trees", I. J. Computer Network and Information Security, Volume 4 No. 8 (2012)
[18] BahramRashidi, "Implementation of An Optimized and Pipelined Combinational Logic Rijndael S-Box on FPGA", I. J. Computer Network and Information Security, Volume 5 No. 1 (2013).

Authors' Profile

**Dr. Hamed Al-Fawareh,** received his Bachelor degree in Computer Science from Yarmouk University, Jordan, in 1994. He obtained his M.Sc. in Computer Science from University Putra Malaysia (UPM), Malaysia, in 1998. He Completed his Ph.D. in Software Engineering from University Putra Malaysia (UPM), Malaysia, in 2001. Currently he is an associated professor of software engineering at Zarqa University Jordan. He was a dean of the faculty science and information technology at Zarqa

University, Jordan from 2010-2013. He is serving as the Secretary General of the Collages of Computing and Information Society (CCIS) at the association of Arab Universities, Editor-in-Chief of the International Arab Journal of Information Technology (IAJIT), and Secretary General of the International Arab Conference of Information Technology (ACIT) from Aug. 2010 to Aug. 2013. His area of interest Software Maintenance, Software Quality, Measurement & Evaluation, Software Reverse Engineering and Reengineering, Software Design Pattern, CARE (Computer Aided Re-Engineering) Tools, Software Testing, and Bioinformatics. He is a member of IEEE and ACM.

**Abla Suliman Hussein,** received her B.S.C and M.C.S degrees in Computer Science from Zarqa University, Jordan, in 2009 and 2015, respectively. Currently she is a Computer Labs- IT Supervisor at Zarqa University, Jordan. Her research interests include wireless ad hoc networks, mobile computing, Digital Logic and Circuits.