International Journal of Control and Automation
Vol. 10, No. 12 (2017), pp.139-146
http//dx.doi.org/10.14257/ijca.2017.10.12.13

A New Method for Boolean Function Simplification

Maher A. Nabulsi[1], Ahmad AA Alkatib[2] and Fatima M. Quiam[3]

[1]Department of Computer Science, Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Amman, Jordan
[2]Department of Computer Information Systems, Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Amman, Jordan
[3]Department of Software Engineering, Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Amman, Jordan
[1]nabulsi@zuj.edu.jo, [2]Ahmad.Alkhatib@zuj.edu.jo, [3]f.quiam@zuj.edu.jo

Abstract

The Karnaugh map technique is the most common technique for academic teaching and can be used by digital designers to minimize Boolean functions to simpler forms. Simplifying functions are necessary for reducing the number of used gates (digital components) in the digital circuits, which reflect the chip size, cost, power, and speed. The K-map technique was proposed by M. Karnaugh. Later, Quine and McCluskey reported tabular algorithmic techniques for Boolean function minimization. Then, many computer-aided modelling languages, such as object-oriented and design packages, were proposed.

Minimization of digital switching functions is a classical problem, but certainly not a dead one. This paper provides a systematic and simple technique for the simplification of logical functions, which effectively reduces the time and unnecessary manipulation comparisons with individual Karnaugh-map minimization methods. The paper reveals that simplification can be accomplished based on the combination of minterms using only a truth table by following simple rules.

***Keywords**: Boolean, Functions, Digital Circuit, Truth Table, Karnaugh Map, Minterms, Symbolic Manipulation*

1. Introduction

Digital logic gates are semiconductors/electronic all digital circuit. A digital gate performs functions based on number of inputs to produce one output with high or low voltage. Logically, logic gate can present any Boolean Function (B.F.), the output voltage values can be translated to '1s' and '0s'. A B.F. is an algebraic expression composed from number of variables (having values 'true' or '1' and 'false' or '0') and number of logical operators ('OR', 'AND', and 'NOT'). It is therefore important to find the simplest representation for the function to simplify the digital circuit. There is significant amount of literature available on digital design and digital circuit. Normally, a Boolean expression can be given using two forms: sum of products and product of sums [1] [2].

1. Sum of products (SOP)

"This is the more common form of Boolean expressions. SOP Boolean expressions may be generated from truth tables quite easily by forming an 'OR' of the 'ANDs' of all input variables (standard products or minterms) for which the output is 1."

Received (December 12, 2016), Review Result (October 18, 2017), Accepted (October 23, 2017)

ISSN: 2005-4297 IJCA

2. Product of sums (POS)

"POS expressions are based on the '0s' in a truth table and generated oppositely as SOP by taking an 'AND' of the 'ORs' of all input variables (standard sums or maxterms). This is the less commonly used form of Boolean expressions."

The Object Oriented, Unified Modeling Language (UML) and other languages created as a result of unification of different object-oriented design methodologies[3] [4] and [5]. Originally, it was applied in software systems design and hardware systems as well.

Different simplification methods have been introduced by many researchers for multiple input/output switching circuits, such as the work of Shannon in 1938 [6], which simplified the representation and manipulation of B.F.s based on classical representation forms, such as truth tables, Karnaugh maps, SOP, POS, and canonical forms [1] [7]. Examples including a reduced SOP in [5] [8].

A combined method mapping by Sahadve and Chandan (2016) [9] to reduced minterms was derived from a truth table and manipulated in a recursive algorithm to reduce the logical circuit design and improve the plotting.

Arunachalam and Rajupillai [10] proposed a novel technique of optimal simplification of B.F using K-map and object-oriented algorithm. The proposed technique is based on the inserting redundant terms in loops.

Debajit *et al.* [11] proposed a new method focused on the order of the variable should be chosen according to the Shannon entropy measurement. Then a matrix is created, where columns represent the variables and rows represent disjoint cubes.

A fast simplification method has been introduced by Hazem M. El-Bakry [12]. The idea depends on using the cross-correlation in the frequency, instead of the time domain, to define the groups on the visualization map.

Das and Mondal [13] introduced the extended version of the K-map, which is based on cluster generation and selection algorithm.

Boolean Algebra (B.A.) is the soul of digital design and computer science and digital circuits. Problems in digital design, testing and artificial intelligence (AI) can be presented as a sequence of B.F.s. Such applications will benefit from representing B.F.s through efficient algorithms manipulation. B.A. simplification is touted as one of the most important skills to IT and engineering students all over the world. This paper proposes a new simple technique that relies only on truth tables to provide a minimal solution based on the breaking of minterms and arranging them in adjacent groups without using the Karnaugh map. This method can save time and effort with respect to designer and academic teaching.

2. Methodology

Any Boolean function can be simplified as a SOP using a truth table first to define the min and max terms, and then to find the simplified functions by grouping the minterms using the Karnaugh map. Our proposed method relies only on the truth table to find the simplified function in SOP (from minterms with value one), thereby saving time and effort.

In this method, the following rules must be considered carefully to get the right answer:

1. A pair of minterms are to be considered as adjacent terms in the truth table; they must be different by only one variable, which is primed (logic 0) in one term and unprimed (logic 1) in the other [*e.g.* (001, 011)].

2. The number of terms that can be selected in one combination (group) must be of the power of 2 ($2^0 = 1$, $2^1 = 2$, $2^2 = 4$, *etc.*).

3. We must terminate forming the combinations when all ‘1s’ are covered.

The proposed method can be applied in the truth table for two, three, and four-variable functions and more.

Two-Variable Function

A two-variable function will produce a truth table with four terms. If a group of two terms is formed, then the answer will be one variable, which is the common one.

Ex. 1. Simplify the following B.F. in SOP form: F (x, y) = **x' y' + x' y**
- Solution using the proposed method:

X	Y	F
0	0	1
0	1	1
1	0	0
1	1	0

F=X'
- Proof using B.A.: F = x' y' + x' y = x' (y' + y) = x'. 1 = x'

Ex. 2. Simplify the following B.F. in SOP form: F (x, y) = **x' y' + x y'**
- Solution using the proposed method:

X	Y	F
0	0	1
0	1	0
1	0	1
1	1	0

F=Y'
- Proof using B.A.: F = x' y' + x y' = y' (x' + x) = y'. 1 = y'

Three-Variable Function

This type of variable function will produce a table with eight terms.

1. In case we have adjacent terms as a group of two, the answer will be the two common variables in the form of AND term (ex.1). If we have more than one group we take the AND terms with OR operation between them (ex.2).

Ex. 1. Simplify the following B.F. in SOP form: F (x, y, z) = **x' y z' + x' y z**
- Solution using the proposed method:

X	Y	Z	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

F = X'.Y

- Proof using B.A.: F = x' y z' + x' y z = x' y (z' + z) = x' y

Ex. 2. Simplify the following B.F. in SOP form:
F (x, y, z) = **x' y' z' + x' y' z + x y' z + x y z**
- Solution using the proposed method:

X	Y	Z	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

F = X'.Y'+ X.Z
- Proof using B.A.: F = <u>x' y' z'</u> + <u>x' y' z</u> + **x y' z** + **x y z**
  = x' y' (z' + z) + x z (y' + y)
  = x' y' + x z

2. If four adjacent terms- each term is adjacent with other two- are found in one group, the answer will be the one common variable between them.

Ex. 3. Simplify the following B.F. in SOP form:
F (x, y, z) = **x' y' z + x' y z + x y' z + x y z**
- Solution using the proposed method:

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

F = Z
- Proof using B.A.: F = <u>x' y' z</u> + <u>x' y z</u> + **x y' z** + **x y z**
  = x' z (y' + y) + x z (y' + y)
  = x' z + x z = z (x' + x) = **z**

3. Also, one term can be considered as a member in more than one group if needed for best simplification, as shown in the following example:

Ex. 4. Simplify the following B.F. in SOP form:
F (x, y, z) = **x' y' z + x' y z + x y z**
- Solution using the proposed method:

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0

0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

- The result of the first group (terms 1 & 3) is X'Z
- The result of the second group (terms 3 & 7) is YZ

Therefore, **F = X'Z+YZ**

-

- Proof using B.A.: F = x' y' z + **x' y z** + **x y z**
  = x' z (y' + y) + y z (x' + x)
  = x' z + y z

Four-Variable Function

In a four-variable function, a truth table of 16 terms can be listed, with the following possible formed groups:

- Two adjacent terms, which lead to one AND term with three common variables; or
- Four adjacent terms, which lead to one AND term with two common variables; or
- Eight adjacent terms - each term is adjacent with other three-, which leads to term with one common variable.

Ex. 1. Simplify the following B.F. in SOP form:
F (w, x, y, z) = **w' x y z + w x y z**

- Solution using the proposed method:

W	X	Y	Z	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

F = XYZ

- Proof using B.A.: F = w' x y z + w x y z
  = x y z (w' + w) = x y z

Ex. 2. Simplify the following B.F. in SOP form:
F (w, x, y, z) = **w' x' y z + w' x y z + w x' y z + w x y z**
- Solution using the proposed method:

W	X	Y	Z	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

F = YZ
- Proof using B.A.: F = <u>w' x' y z</u> + <u>w' x y z</u> + **w x' y z** + **w x y z**
= w' y z (x' + x) + w y z (x' + x)
= w' y z + w y z = y z (w' + w) = y z

Ex. 3. Simplify the following B.F. in SOP form:
F (w, x, y, z) = **w' x' y z' + w' x' y z + w' x y z' + w' x y z +**
+ w x' y z' + w x' y z + w x y z' + w x y z
- Solution using the proposed method:

W	X	Y	Z	F
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

F = Y

- Proof using B.A.: F = w' x' y z' + w' x' y z + **w' x y z'** + **w' x y z** +
  + w x' y z' + w x' y z + w x y z' + w x y z
  = w' x' y (z' + z) + w' x y (z' + z) + w x' y (z' + z) + w x y (z' + z)
  = w' x' y + w' x y + **w x' y** + **w x y**
  = w' y (x' + x) + w y (x' + x)
  = w' y + w y = y (w' + w) = y

An Example of How to Solve Problems Using the Proposed Method
Ex – Simplifying the following B.F. given in a truth table in the SOP form:

W	X	Y	Z	F	Group 1	Group 2	Group 3
0	0	0	0	1	m0		m0
0	0	0	1	1	m1		
0	0	1	0	1			m2
0	0	1	1	0			
0	1	0	0	1	m4		
0	1	0	1	1	m5	m5	
0	1	1	0	0			
0	1	1	1	1		m7	
1	0	0	0	1	m8		
1	0	0	1	1	m9		
1	0	1	0	0			
1	0	1	1	0			
1	1	0	0	1	m12		
1	1	0	1	1	m13	m13	
1	1	1	0	0			
1	1	1	1	1		m15	

So, if we have eight or more '1s' in the truth table, we first search if there is a common variable in any group of the eight (group 1 = y'). Next, we find out if there are two common variables in any group of the four, not all minterms are included in the previous groups of eight (group 2 = XZ). Finally, we find out if there are three common variables in any group of the two, not all are included in the previous groups of eight or four (group 3 = W'X'Z'). Therefore, this function can be simplified as follows:

F = Y'+ XZ+W'X'Z'.

3. Conclusion

This paper proposes an efficient technique of multiple input digital circuit minimization. The proposed method is very simple and is a less laborious approach to determine the minimal gate count for multiple input digital switching circuits. This exact method of minimization provides a minimal solution based on adjacent groups in K-maps. This paper proposes a new technique for finding adjacent groups and SOP using only the truth table to reduce time, efforts, and complexity of the process. The technique can be useful for both students and designers.

References

[1] M. Mano and M. Ciletti, "Digital Design", 4th edition, USA: Pearson, **(2000)**.
[2] K. P. Parker and E. J. McCluskey, "Probabilistic treatment of general combinational networks", IEEE Transaction on Computers, vol. 24, no. 6, **(1975)**, pp.668–670.
[3] OMG, UML Superstructure specification, v2.0. Available from: http://www.omg.org/cgi-bindoc?formal/05-07- 04. [Accessed 28th April 2016].
[4] G. Booch, J. Rumbaugh and I. Jacobson, "The Unified Modeling Language User Guide", Twelfth Reprint, India: Pearson Education, **(2004)**.
[5] T. Roff, "UML: A Beginner's Guide", Fifth Reprint, Tata McGraw-Hill Edition, **(2006)**.
[6] S. Roy and C. T. Bhunia, "On synthesis of combinational logic circuits", International Journal of Computer Applications, vol. 127, no. 1, **(2015)**, pp. 21–26.
[7] F. J. Hill and G. R. Peterson, "Introduction to Switching Theory and Logical Design", New York: Wiley, **(1974)**.
[8] J. P. Roth, "Computer Logic, Testing, and Verification", Potomac, MD: Computer Science Press, **(1980)**.
[9] A. Roy and C. Bhunia, "A set combine map method for manual synthesis of logic circuits", International Journal of Applied Engineering Research, vol. 11, no.4, **(2016)**, pp. 2663–2670.
[10] A. Solairaju and R. Periyasamy, "Optimal Boolean function simplification through K-map using object-oriented algorithm", International Journal of Computer Applications, vol. 15, no.7, **(2011)**, pp. 28-32.
[11] D. Sensarma, S. Banerjee, K. Basuli, S. Naskar and S. Sarma, "On an optimization technique using binary decision diagram", International Journal of Computer Science, Engineering and Applications, vol. 2, no.1, **(2012)**, pp.73-86.
[12] H. El-Bakry, "Fast Karnaugh map for simplification of complex Boolean functions", 10th WSEAS International Conference on Applied Computer Science Iwate Prefectural University, Japan, **(2010)**, pp.478-483.
[13] P. Das and B. Mondal, "Extended K-map for minimizing multiple output logic circuits", International Journal of VLSI design & Communication Systems, vol. 4, no. 4, **(2013)**, pp.1-8.

Authors

Maher A. Nabulsi, he is an Associate Professor of Computer Science at Alzaytoonah University of Jordan. He got his Phd in automatic control and computer engineering. He participated in many conferences and published in many journals. His research interests include computer architecture, logic design, discrete structures, microprogramming, automatic control and robotics.

Ahmad A.A. Alkhatib, he is a Doctor of Telecom & Network Engineering at Alzytoonah University – Jordan. He holds the Ph.D. from University of Wales, Newport UK with major of Wireless Sensor Network; an MSc, earned at The University of Technology, Sydney Australia with major of Telecommunication Engineering and Telecommunication Network; an BSc, from Yarmouk University in Telecommunication Engineering. Dr Alkhatib Published in Scholarly Journals and conferences in the areas Wireless Sensor Networks.

Fatima M. Quiam, she is a lecturer of Software Engineering at Al-Zaytoonah University of Jordan. She received her B.Sc and M.S. degree in computer science from the Jordan University of Science and Technology, Jordan. Her research interests include parallel and distributed computing, data mining and discrete structures.