IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 5, May 2021

DOI: 10.17148/IARJSET.2021.8533

Implementation of Quine Mc-Cluskey Method on FPGA

Sunil Kumar G R[1], Ayesha Rosheen[2], Bindu J[3], Chaithra V M[4]

Assistant professor, Department of ECE, K.S Institute of Technology, Bangalore, India[1]

Final Year B.E, Department of ECE, K.S Institute of Technology, Bangalore, India[2,3,4]

Abstract: In this paper the Quine Mc-Cluskey method is used for minimization of Boolean functions. The Quine-Mc-Cluskey method is one of the popular method, and is particularly used when both karnaugh method and variable entrant map (VEM) approach fails, it is useful in minimizing logic expressions for larger number of variables when compared with minimization by Karnaugh Map or Boolean algebra. It was developed in 1956 by Edward J.MC Cluskey. Karnaugh map could solve only for smaller variables whereas Quine Mc-Cluskey can solve for higher variables. The concept that runs behind Quine Mc-Cluskey is the combaining method approach. It has easy algorithm than Karnaugh, so it is efficient and can be easily implemented in computer algorithms.

Keywords: Quine Mc-Cluskey, FPGA, Hardware implementation, Karnaugh map

I. INTRODUCTION

The Quine-Mc-cluskey algorithm is one such algorithm which is easy to implement on the hardware. This tabular method of prime implicants is used for minimizing the Boolean expression, Simplification of Boolean expression is a practical tool to optimize programing algorithms and circuits. Several techniques have been introduced to perform the minimization, including Boolean algebra (BA), Karnaugh Map (K-Map). K-Map is a diagrammatic technique based on a special form of Venn diagram, It is easier to use than BA but usually it is used to handle Boolean expression with no more than six variables, When the number of variable in Karnaugh map exceeds six, the complexity of the map is exponentially enhanced, and it becomes more and more cumbersome, with increasing in the number of inputs, the pattern recognition in Karnaugh map can be difficult or sometimes even impossible. The quine mc-cluskey method does not require pattern recognition. It includes steps like finding all prime implicants of the function and selecting a minimal set of prime implicants of the function. Functionally identical to K-Map, QM method is more executable when dealing with larger number of variables and is more suited for programming the computer and gives the simplified minimal form of Boolean functions that can be reached.

II. LITERATURE SURVEY

[1] Computer Simulation Codes for the Quine Mc-Cluskey Method of Logic Minimization by Sourangsu Banerji Department of ECE, RCC-Institute of Information technology. In this paper they have included the codes for the implementation of Quine-McCluskey method using the computer languages C and C++ which provides the insight about the inherent advantage we get in the object oriented design paradigm as compared to the procedural languages.

[2] Optimization of the quine-mccluskey method for the minimization of the boolean expressions by Tarun Kumar Jain,, Dharmender Singh Kushwaha, and Arun Kumar Misra. In Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on. IEEE, 2008. In this paper it involves the basic principle in designing digital circuit hovers thus reducing the cost too. To achieve this, they have used Boolean expression that helps in obtaining minimum number of terms and does not contain any redundant pairs.

[3] Boolean functions simplification algorithm of O (N) complexity by BASÇÍFTÇÍ, Sirzat KAHRAMANLI Fatih Mathematical & Computational Applications 8.3 (2003): 271-278. In this paper they have discussed about finding the minimal set of prime implicants of O(n) complexity instead of O(2^n).

[4] Quine-McCluskey classification by Javad Safaei and Hamid Beigy department of Computer Engineering, Sharif University of Technology. In this paper the Karnaugh and Quine-McCluskey methods are used for symbolic classification

This work is licensed under a Creative Commons Attribution 4.0 International License

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 5, May 2021

DOI: 10.17148/IARJSET.2021.8533

problem because the data in classification problem is very large, some changes should be applied in the original Quine-McCluskey algorithm. GQMC results are most of the time equal to QMC.

[5] Quine-McCluskey algorithm on GPGPU by Vladimír Siládi, Matej Bel University, Tomáš Filo. This paper deals with parallelization of the Quine–McCluskey algorithm. The problem computed by this algorithm is NP-hard and runtime of the algorithm grows exponentially with the number of variables. The goal is to show that parallel implementation of the Quine–McCluskey algorithm on graphics processing units brings significant acceleration of computing process.

[6] Minimization of Switching Functions using Quine-McCluskey Method by Vladislav Manojlović, Kneza Miloša Kosovska Mitrovica Faculty of Technical Sciences. This paper presents Quine-McCluskey algorithm for minimizing switching functions, with additional specific elements, such as starting part (that is decoding DNF form) and cost of circuit.

[7] Quine-McCluskey method for many-valued logical functions by Milan Petr´ık, Center for Machine Perception, department of Cybernetics Faculty of Electrical Engineering, Czech Technical University Technick´a 2, 166 27 Prague 6, Czech Republic. In this paper the generalized method can find a normal form for any finite-valued logical function. More-over this normal form is simpler than that found by the intuitive method using the table of values. The method has been successfully implemented and tested on exam.

[8] Effect of Quine-McCluskey Simplification on Boolean Space Complexity by P. W. Chandana Prasad, Azam Beg, Ashutosh Kumar Singh. This paper describes a new model for the estimation of circuit complexity, based on Quine-McCluskey simplification method. The proposed method utilizes data derived from Monte-Carlo simulations for any Boolean function with different count of variables and product term complexities. The model allows design feasibility and performance analysis prior to the circuit realization.

[9] Programming implementation of the Quine-McCluskey method for minimization of Boolean expression by Jiangbo Huang department of Biological Sciences, Faculty of Science National University of Singapore, Singapore 117604. QM simulation code based on C programming is introduced. Theoretically it is able to handle any number of variables and has taken the Don't-Care conditions into account.

[10] Modified Quine-McCluskey Method by Jadhav Vitthal and Buchade Amar. This paper proposes E-sum based optimization to Quine-McCluskey Method to increase its performance by reducing number of comparisons between minterm list in determination of prime implicants. Modified Quine-McCluskey method (MQM) can be implemented to any number of variables.

III. METHODOLOGY

The simplification of large variables of an expression is our main concern over here, for which we are using Quine-McCluskey algorithm or the method of prime implicants which is used for minimization of Boolean functions. The other algorithms like Karnaugh method uses maps, which becomes very difficult to design as the number of input variables increases and, also the Pattern recognition of adjacent cells becomes impossible, therefore we are using an alternative method, that is Quine-McCluskey method. So, this method firstly includes the writing of code in hardware description language (HDL), this code helps us to minimize the higher variables to simple form using Quine Mc-Cluskey algorithm. Then verify the HDL code by executing it, and then implement the verified code on Field-programmable gate array kit (Spartan-6). This algorithm steps includes listing all the minterms in binary form and arrange the minterms according to number of 1's. Compare each binary number with every term in the adjacent next higher category and if they differ only by one position, put a check mark and copy the term in the next column with '-' underscore symbol in the position that they differed. Then apply the same process for the result column and continue these cycles until a single pass through, cycle yields no further elimination of literals. Then list all the prime implicants. Ultimately select the minimum number of prime implicant's which must cover all the minterms. This HDL code which is written according to the step's involved in solving the Quine-McCluskey algorithm should be executed finally and implement it on FPGA kit.

This work is licensed under a Creative Commons Attribution 4.0 International License

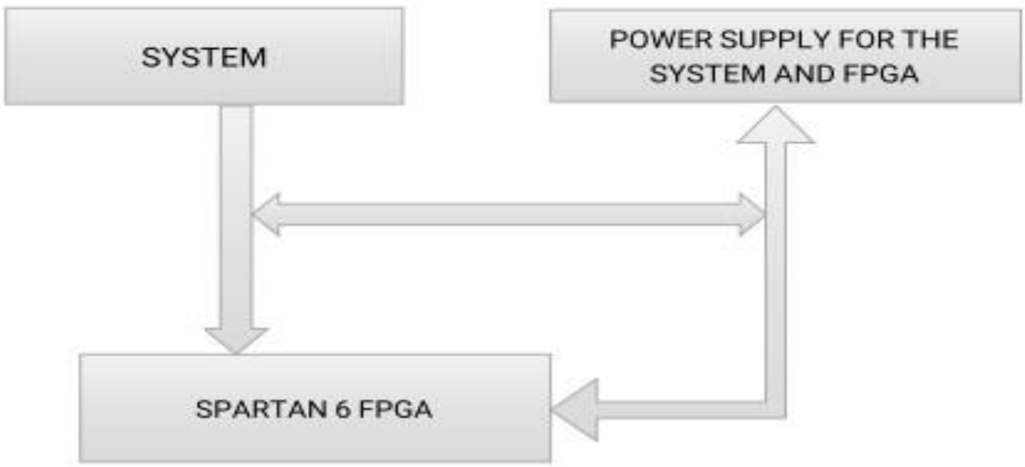
IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 5, May 2021

DOI: 10.17148/IARJSET.2021.8533



IV. CONCLUSION

The proposed system helps in avoiding the complexity and makes easy to implement on the hardware by which we get the simplified form of any given expression easily by using the quine mc-cluskey algorithm. By executed this algorithm we are just not able to get simplified expression, but we also get minimal form of Boolean functions that can be reached. By using this algorithm of quine mc-cluskey we will be able to simplify any number of variables unlike other methods as in karnaugh map where the number of variables and limited.

V. REFERENCES

[1]. Computer Simulation Codes for the Quine Mc-Cluskey Method of Logic Minimization by Sourangsu Banerji Department of ECE, RCC-Institute of Information technology.

[2]. Optimization of the quine-mccluskey method for the minimization of the boolean expressions by Tarun Kumar Jain,, Dharmender Singh Kushwaha, and Arun Kumar Misra. In Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on. IEEE, 2008.

[3]. Boolean functions simplification algorithm of O (N) complexity by BASÇÍFTÇÍ, Sirzat KAHRAMANLI Fatih Mathematical & Computational Applications 8.3 (2003): 271-278.

[4]. Safaei, Javad, and Hamid Beigy. "Quine-McCluskey Classification." Computer Systems and Applications, 2007. AICCSA'07. IEEE/ACS International Conference on IEEE, 2007.

[5].Siládi V.,&Filo T.Quine-Mccluskey algorithm on GPGPU.AWERProcedia Information Technology & ComputerScience. [Online]. 2013, 04, pp 814-820.

[6]. Manojlović, Vladislav (2013) Minimization of Switching Functions using Quine-McCluskey Method. International Journal of Computer Applications (0975 – 8887) Volume 82 – No 4, November 2013, 12–16.

[7]. Petrík, M. Quine–McCluskey method for many-valued logical functions. Soft Comput 12, 393–402 (2008).

[8]. Prasad, P., Azam Beg, and Ashutosh Kumar Singh. "Effect of Quine-McCluskey simplification on Boolean space complexity." Innovative Technologies in Intelligent Systems and Industrial Applications, 2009. CITISIA 2009. IEEE, 2009.

[9].] Programming implementation of the Quine-McCluskey method for minimization of Boolean expression by Jiangbo Huang department of Biological Sciences, Faculty of Science National University of Singapore, Singapore 117604.

[10]. Jadhav, Vitthal; Buchade, Amar (2012-03-08). "Modified Quine-McCluskey Method". arXiv:1203.2289.

This work is licensed under a Creative Commons Attribution 4.0 International License