

CMOS LOGIC GATE GENERATION AND OPTIMIZATION BY EXPLORING PRE-DEFINED SWITCH NETWORKS

¹Anderson Santos da Silva, ²Vinicius Callegaro, ^{1,2}Renato P. Ribas, ^{1,2}André I. Reis

¹Institute of Informatics / ²PPGC - UFRGS, Porto Alegre, Brazil

{assilva, vcallegaro, rpribas, andreis}@inf.ufrgs.br

ABSTRACT

This work investigates the Moore's catalog of switch networks and its use to optimize the CMOS logic gate design. It presents an analysis of networks' properties described in the catalog, as the number of switches, the series-parallel associations in the arrangements, the longest device path, the planar profile of the network, and so on. The approach used to generate such data is shown as well as some experimental results illustrating the usefulness of this catalog. Moreover, the Moore's catalog has been verified and validated by such computational analysis.

1. INTRODUCTION

The improvement of ASIC design can be considered from different aspects and levels. One of them is the optimization of CMOS logic gates included in standard cell libraries. Algorithms for generating "optimal" transistor networks according to some criteria represent an aspect of potential improvements. Among these are the Boolean function factoring [1][2] used for series-parallel networks and methods based on graphs [3]-[5] which also cover non-series-parallel arrangements.

The Moore's catalog [12] can be used to deliver optimal transistor networks for all Boolean functions up to four inputs, because it provides networks with minimum number of switches (i.e., transistors, in this context).

2. BACKGROUND

For a better understanding of this work, a brief review of some basic concepts may be helpful.

2.1. Transistors Networks

The basic element for building a transistor network is a logic switch. It is understood as a logic switch element that can become active when applied to the logical value "1" in its control terminal.

Thus, a transistor network can be characterized by the distribution of these switches on its physical form (transistors). There are several ways to find a network to represent an arbitrary logic function, each with its

advantages and disadvantages. For a network with two terminals, four main groups can be outlined:

- **Planar networks:** Networks that have the arrangement of its transistors in a form corresponding to a planar graph. It is understood by the notion of a planar graph that has no edges crossing with each other. These networks have the property of having a topologically complementary network (dual graph [8]) of their direct network. **Non-planar networks**, in turn, are the ones that cannot be represented by a planar graph.

- **Series-parallel (SP) networks:** Networks in which all switches are connected in series and parallel associations recursively. This type of configuration always corresponds to a planar graph. Otherwise, the network is said to be **Non-series-parallel (NSP)** since it presents at least one Wheatstone bridge.

- **Self-Dual Networks:** Networks that have the property to have its dual graph identical to the corresponding original one. Thus, its complementary plan only changes the type of the transistor while maintaining the same topology. Inverter and carry-out of full-adder logic gates can be built with this property [5].

Despite being an important factor, the distribution of transistors on a network is not the only concern about their physical behavior. For instance, it is interesting to minimize the total number of contacts present in these networks, because it decreases the space required in the physical layout as well as favoring desirable electrical aspects to this type of device.

2.2. Switch Network Generation Methods

Several methods exist to generate transistor (switches) networks and these can be represented by various types of structures. Among these structures, Boolean functions can be used, because they have different modes of representation. The most common are the truth table, Boolean equation, logic gates, a list of terms (minterms or maxterms), binary decision diagram (BDD) and integer number (represented in binary form, hexadecimal form or decimal form).

A Boolean equation is a set of terms representing logical values linked by logic connectivity representing logic functions. This can be factored, i.e., rules of Boolean logic can be applied in order to eliminate redundancies in the expression denoting the function [1][2]. This approach does not always produce the

minimum number of transistors compared to methods based on graphs when an NSP solution presents less switches than the SP one [3]-[5].

Binary decision diagram (BDD) can also be used for this purpose. It is a data structure consisting of nodes, arcs and terminals representing a logic function. LBBDD, (Lower Bound BDD), proposed in [9], uses these structures to generate arrangements that meet the minimum number of switches required in series to represent a given logic function (the MDC property of a Boolean function).

2.3. Catalogs

There are catalogs that provide optimal networks in terms of the minimum switch count. These catalogs are based on particular methods, often purely mathematical, showing results comparable to current methods of generation. Moore's catalog, for functions of up to 4 inputs, is an example of this that will be explored in the next section. However, there are other catalogs, such as Ninomiya's catalog [13], offering optimal arrangements too, where the analysis of their results and comparisons is planned as future work.

3. METHODOLOGY OF ANALYSIS

The main idea of the work on the aspects of generating optimal networks is focused on the use and understanding of the Moore's catalog for functions with up to 4 variables, illustrated in Fig. 1.

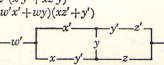
Harvard serial No.	Complement	Category class	Constituents (Elementary circuits)	Algebraic expression for series parallel Schematic for others	Notes	Number of contacts	Number of springs	Number of circuits
1	400	I	ONE CONSTITUENT 0 0/4 4/4 (1234/4 0123/4)	$w'x'y'z'$	J	4	8	16
2	396	II	TWO CONSTITUENTS 0 1 same cube 0 3 same cube 0 7 same cube 0 15 04/4 (123/4)	$w'x'y'$ $w'x'(y'z'+yz)$ $w'(x'y'+xz)(y'+z')$ $(w'+x)(wy'+x'z')(y'+z)$	J	3	6	32
3	397				J	6	10	48
4	398				J	7	11	32
5	399				J	8	12	8
6	390				III	THREE CONSTITUENTS 0 1 2 same cube 0 1 6 same cube 0 1 14 0 3 5 same cube 0 3 12 0 3 13	$w'x'(y'+z')$ $w'(x'y'+xz'y)$ $(w'x'+wy)(xz'+y')$ 	J
7	391	J	6	10				192
8	392	J	7	11				64
9	393	Q	8	13				64
10	394	W	9	14				48
11	395	X	9	14				96

Figure 1 –Part of the Moore's catalog.

This catalog brings together an optimal set of arrangements to networks of the NPN class of Boolean functions with up to 4 inputs [6]. These results were calculated using methods that professionals (designers) of the telecommunications area knew around 1958 to optimize the minimum number of contacts of circuits based on relay switches. Such study was more the result of designers' experience in the generation of these kind of circuits than from a formal mathematical method of generation. They compiled a catalog where the function

and its complementary one belong to the same NPN class and thus were related. Each line represents a function, and the minterms that are covered by it, information about its complementary function, the total number of necessary contacts and some other notes on relevant information. Each network is presented in its minimum switch count version. It is done by equation when an SP network is the best solution and by the network illustration when an NSP solution is preferred. The complementary networks are omitted because they might be obtained by generating the negated function or the dual graph.

Network whose the best arrangement is non-planar can be displayed together with its complementary one. That depends on whether the catalog also offers an optimal network for the dual graph of the network directly. If there is not a complement, the generation of the dual graph (topologically complemented) is sufficient [5].

3.1. Longest Path and MDC

A methodology for analysis of the networks present in the catalog is the concept of LDC- the Longest Decision Chain (also known the longest path). Basically, the focus is to find out the longest logical path between terminal nodes concerning transistor count.. In this way, the polarity of variables in path is considered because the presence of a variable and its negation in a path means that this one is never stimulated ('false path'). In other situation the longest path can be covered by a shorter one, and again a 'false path' is characterized. Thus, the LDC must be obtained excluding false paths. This analysis is then compared with the MDC [7], a theoretical analysis on the most essential cube needed to represent the function. It is desirable that the LDC present in the original network is equal to the same MDC.

3.2. Signatures

For a standard design of the logic gate it is necessary to obtain the pull-up and pull-down plans. Thus, determining when a gate could implement an arbitrary logic function is a common problem in logic synthesis. Moore's catalog can be used to investigate the arrangement of these plans. The strategy is to use a canonical signature for a function to find the optimal arrangement. In this correspondence, it is necessary explain the concept of 'canonical signature'. This signature is necessary to find witch network of the catalog is the correct one for some arbitrary function..

It is known that for a function F one can apply any permutation of their input variables, operation by which it is denoted by P. Additionally, it may negate inputs in any combination. Denote it by N. A negation can be added to its output and in this case it gets the complementary function.

So if F is denoted by $F(A, B, C)$, an NP application, whereas P represents a permutation of the set $\{A, B, C\}$, N a negation in any one of the variables of this set and the symbol '!' the operation of negation, this application would result in:

$F(A, B, C)$ F without permutation and negation.

$F(A, C, B)$ only permutation.

$F(!A, C, B)$ negation and permutation.

For generating a canonical signature, all permutation, input and output negation were performed in the original function in order to identify the lowest integer value, which would be the NPN signature.

3.3. NPN Matching

The algorithm chosen to generate a canonical signature was proposed by Sasao [6]. For a given function, it generates a table with all the permutations and negations of input and output (NPN) and finding a resulting signature that has the lowest value seen as a whole (less representative).

For example, the function 00001000 has as signature the function 00000001.

On this basis, the signature was generated for all functions present in the catalog. These set of signatures could be used to identify which network could be chosen to perform any boolean function up to four inputs.

3.4 Look Up Table

As Moore's catalog is a structure that is accessed often, their content was passed to the electronic media based on a 'look-up table', i.e., a table on which were inserted about all the fields present in its structure. This table consists of all functions of four inputs and guard which NPN class it belongs, the number of transistors, permutations on input and output negations.

In terms of memory, it spends only 26 bits (per function):

- 9 bits for the NPN function.
- 4 bits for the transistor count.
- 8-bit input permutation.
- 4-bit inversions in the input.
- An inversion in the output.

So, one can use the function itself as a key to access its information corresponding to the catalog described above and generate a pre-calculated table for all functions with 4-inputs (65.536 entries).

4. EXPERIMENTAL RESULTS

The validation of the networks present in Moore's catalog was made using a visualization tool based on the Karma [10] and Switchcraft [11] tools. An algorithm was developed to verify the correctness and the logic function that each function of catalog perform.

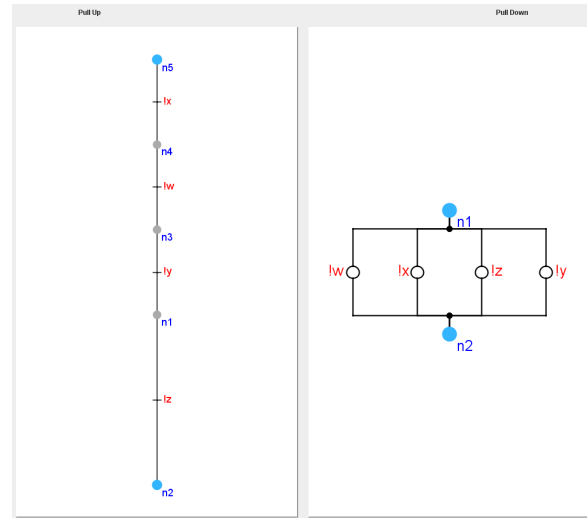


Figure 2- Network with serial number 1 of catalog.

Information of direct (complementary) plan can be viewed and a logic function simulation in both plans is made and validated.

Another great concern was the gain in the number of inverters. For each function present in the Moore's catalog was calculated a NPN transformation and chosen a permutation and negation that not only minimizes the number of transistor but also the number of inverters. The biggest gain observed was 4 inverters in the network of Fig. 2.

4.1 LDC of Network versus MDC of Function

Some functions of the catalog have the LDC greater than the theoretical limit set by the MDC. In total, 99 functions do not respect the MDC limit of 287 functions presented in this catalog. This means that in these cases there is an arrangement different than the one that optimizes the network.

For example, the function with hexa 0x8997 of Moore's catalog has 5 with LDC value and 3 with MDC value. It is presented in Fig. 3.

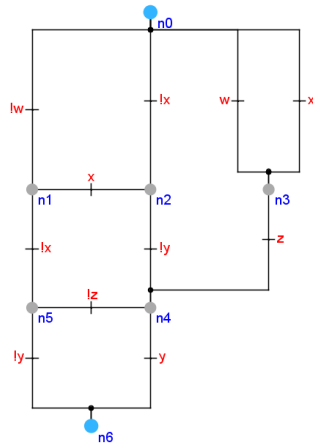


Figure 3: Switch network with hexa 0x8997.

Note that a LDC is [$!w, x, !y, !z, !y$] and this do not respects MDC. This fact can be corrected with some factorization that respects the MDC, because the catalog don't minimizes a LDC (only minimizes a number of transistors).

The largest value found in catalog for LDC was 6 without any factorization. And it occurs in function represented by hexa 0x6BBC. In Fig. 4 is presented a graphic with results of the difference between LDC and MDC (LDC minus MDC) in all functions presents in catalog. When this value is 0 (zero), network respects the MDC (optimal).

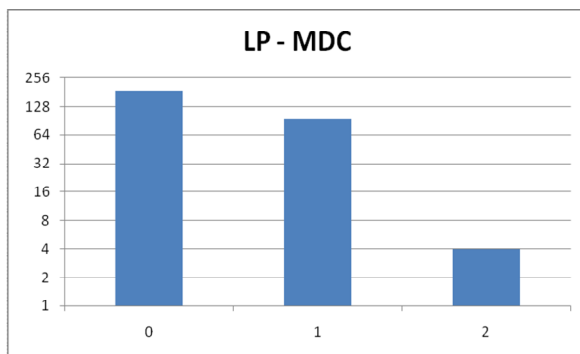


Figure 4 –Graphic in comparison LDC-MDC.

Using the factorization proposed in [1] and [2], one obtains the results that reach the limit of the MDC for networks that don't respect the MDC. This shows that the optimal result can be achieved with a factorization method in these cases. These factorizations respects the MDC value and switch count in all functions of Moore's catalog.

Results as that shown in Fig 4 demonstrate a gain in using the catalog networks. Among all the 287 networks directly presented in the Moore's catalog, the MDC value is equal to LDC value, without any factoring, in 136 functions. The MDC value of remaining 151 functions can be achieved based on factorizations proposed above.

4.2 Planar versus Non-Planar Networks

Moore's catalog often shows non-planar networks to optimal structure for a given function researched. Among all your 287 direct networks, 239 are planar and the remaining 48 are non-planar. For each of these networks, an investigation of the disposal of its series-parallel version was made and it was found that the gain and the number of transistors. Fig.5 shows the values collected of non-planar networks present on the catalog.

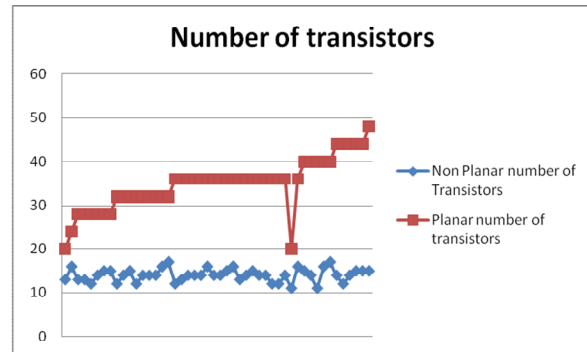


Figure 5- Number of transistor non-planar networks.

The data more below in Fig. 5 represents non-planar networks that have better results in number of transistors. Every planar network has the same number of transistors in both planes and the catalog offers a network arrangement better than a factorization in some cases, but in the case of non-planar it is not always true. Only network with serial number 103 has difference between the numbers of transistors in both planes.

5. CONCLUSION

This paper has presented aspects of the Moore's catalog and a way of minimizing the logical networks on the number of transistors and their physical arrangement. As future work, a method to generate circuits with more than four variables will be implemented using others approaches. The analysis of another catalog will made and compared with values obtained in this work to generate a common standard between them.

6. ACKNOWLEDGMENTS

Research partially funded by Nangate Inc. under a Nangate/UFRGS research agreement, by CAPES and CNPq Brazilian funding agencies, and by the European Community's Seventh Framework Programme under grant 248538 - Synaptic

7. REFERENCES

- [1] M. G. A. Martins, L. S. Rosa Jr., A. B. R. Rasmussen, R. P. Ribas, and A. I. Reis, "Boolean factoring with multiple objective goals". ICCD 2010, pp. 229-234.
- [2] V. Callegaro, L. S. Da Rosa Jr., A. I. Reis, and R. P. Ribas. "A kernel-based approach for factoring logic functions," Microeletronics Students Fórum, 2009.
- [3] J. Zhu and M. Abd-El-Barr, "On the optimization of MOS circuits," IEEE Trans. On Circuits and Systems, vol. 40, no. 6, pp. 412-422, June 1993.
- [4] D. Kagaris and T. Haniotakis, "A methodology for transistor-efficient supergate design," IEEE Trans. on VLSI Systems, vol. 15, no. 4, pp. 488-492. Apr. 2007.
- [5] L. S. Da Rosa Junior, "Automatic generation and evaluation of transistor networks in different logic styles," PhD Thesis, PGMicro / UFRGS, 2008.
- [12] E. F. Moore, "Table of four-relay contact networks", In: Logical Design of Electrical Circuits, by R. A. Higonnet and R. A. Grea, McGraw-Hill, 1958.
- [8] T. Uehara, W.M vanCleemput, "Optimal Layout of CMOS Functional Arrays", Proceeding DAC '79 Proceedings of the 16th Design Automation Conference. IEEE Press Piscataway, NJ, USA, 1979.
- [9] L.S.Rosa Jr, F.S.Marques, T.M.G.Cardoso, R.P.Ribas, S.S. Sapatnekar, A.I.Reis. 2006. Fast disjoint transistor networks from BDDs. SBCCI 2006, pp. 137-142.
- [13] Prof. I. Ninomiya, "Table of minimal switching circuits", In: Introduction to switching and automata theory, by M. A. Harrison, McGraw-Hill Book Company, 1965.
- [6] D. Debnath and T. Sasao, "Efficient computation of canonical form for Boolean matching in large libraries," Proc. Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 591-596, 2004.
- [7] Mayler G.A. Martins, Vinicius Callegaro, Renato P. Ribas, Andre I. Reis, "Efficient Method to Compute Minimum Decision Chains of Boolean Functions", In Proceedings of the 21st edition of the great lakes symposium on Great Lakes symposium on VLSI(GLSVLSI '11). ACM,New York, NY,USA, 419-422.
- [10] C.E Klock, R. P. Ribas, A.I. Reis, "Karma: um ambiente para o aprendizado de síntese de funções Booleanas", Revista Brasileira de Informática na Educação, Voluta 18, Número2, 2010.
- [11] V. Callegaro, F. de Souza Marques, C. E .Klock, L. Da Rosa Jr, R. P. Ribas, A .I. Reis, "SwitchCraft: a framework for transistor network design", SBCCI '10 Porceedings of the 23rd symposium on Integrated circuits ans system design.