Variable Reordering for Reversible Wave Cascades - PRELIMINARY VERSION

Dimitrios Voudouris, Marinos Sampson and George Papakonstantinou
National Technical University of Athens, Greece.

Abstract

In this paper a theoretical background for the architecture of Reversible Wave Cascades is discussed, as well as some minimization algorithms for this architecture. Moreover, heuristic algorithms are proposed for estimating the optimal (or near optimal) variable ordering of a switching function, which drastically improves the simplification results of those, previously mentioned, algorithms. The topology of Reversible Wave Cascades is useful because it has been proved to be reversible and moreover it may help in the design of quantum circuits.

1 Introduction

Logic synthesis and its related problems of Boolean algebra constitute a hot scientific area and much research in this field has taken place over these last decades. For many years, logic synthesis was based on AND, OR and NOT gates. Recently the XOR (eXclusive OR) gate is being used in place of the OR gate, because it has been observed that it can reduce the complexity of most real-life application circuits [1].

A boolean (switching) function can be expressed in many different ways. Different expressions of a Boolean function may be useful in different situations e.g. implementing a function in a specific type of FPGA (Field Programmable Gate Array). Moreover it is usually desired to detect those expressions that are optimal, regarding a certain criterium, e.g. it may be desirable that the produced expression have the least possible number of variable literals. Finding suitable and optimal expressions for a Boolean function can be a very difficult and interesting mathematical problem.

In this work we deal with a special type of expressions called ESCT (Exclusive or Sum of Complex Terms) expressions. They can be mapped to a special cellular architecture called Reversible Wave Cascades (Fig. 1). It has been proved [5] that this special architecture can be directly mapped to reversible logic gates and more specifically to Generalized Toffoli gates. A logic gate is called reversible, if it has the same number of inputs, outputs, and maps each input vector into a unique output vector and vice versa. Moreover both fan-in and fan-out are forbidden. In [7] it has been proved that all quantum logic circuits must be reversible. Therefore ESCT expressions and the Reversible Wave Cascades architecture may aid in designing quantum circuits.

In order to form ESCT expressions for a Boolean function we may use the ESCT expressions of its subfunctions through relations known as Boolean decompositions. Ashenhurst[2], Curtis[3], Roth and Carp[4] did pioneering work on the field of Boolean decomposition. An ESCT expression is a XOR sum of a special kind of terms, called complex terms.

In this work we present mathematical formulations that help us detect those ESCT expressions of a certain Boolean functions that are optimal, regarding the number of their complex terms, and even more we try to find appropriate variable orderings that produce the best possible results.

Some algorithms have been developed in the past, for mapping switching functions to cellular array architectures and then minimizing the number of the produced complex terms. In references [8], [9], [11], [12] techniques like variable reordering and cube transformation or even multi-valued logic are used, in an effort to minimize the number of complex terms, though their architectures differ from the one presented in this paper and are more complex. In references [13], [14] a systematic method was presented to produce architectures very similar to the Reversible Wave Cascade, by extending the EXORLINK [15] operation to complex terms. In Ref [5] an algorithm was proposed for mapping Reversible Wave Cascades to reversible gates (more specifically Generalized Toffoli gates), although it was not implemented. A similar reversible architecture is described in [16], with major objective to minimize the number of garbage inputs, although at that time no algorithm for the mapping and minimization of an arbitrary function to such architectures had been proposed. In Ref [10] BDDs (Binary Decision Diagrams) along with variable reordering techniques are used to produce cellular arrays. The architecture produced is similar to the Reversible Wave Cascades, although the output of each cell may be connected not only to its physical next but to others as well. In Ref [17] two algorithms were presented that produced ESCT expressions for a single-output boolean function while minimizing the number of terms in the produced expression. The first one guarantees minimality for functions up to 5 variables, while the second algorithm applied the first one on groups of cascades, inside the expression, as a term transformation operation. These algorithms have been improved in [18] with the introduction of relative terms and in [19] they have been extended for multi-output boolean functions. In Ref [20] a new algorithm is proposed which can detect minimal ESCT expressions for single-output switching functions of up to 6 input variables.

In this paper we give a brief description of the algorithms presented in [17][18] and [20]. Moreover, some algorithms are proposed for the variable reordering problem, which can drastically improve the results of these previously mentioned minimization algorithms.

2 Theoretical background

In this section we provide some background definitions.

Definition 1. *A complex Maitra term [21] (complex term or Maitra term for simplicity) is recursively defined as follows:*

1. *Constant* 0 (1) *Boolean function is a Maitra term.*
2. *A literal is a Maitra term.*

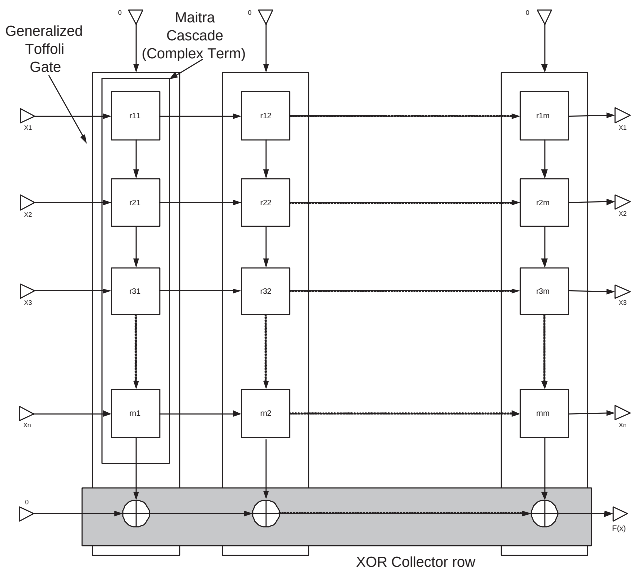

Figure 1: Reversible wave cascade CA

3. If M_i is a Maitra term, a is a literal, and G is an arbitrary two-variable Boolean function (Maitra cell), then $M_{i+1} = G(a, M_i)$ is a Maitra term.

Additionally, it is required that each variable appears in each Maitra term only once.

Definition 2. *An ESCT (Exclusive-or Sum of Complex Terms) expression (some times also called reversible wave cascade or Maitra expression) for a switching function is an exlusive-OR sum of complex terms:*

$$f = \sum_{i=1}^{m} \oplus M_i,$$

where M_i are complex terms and m is their number inside the expression. The same variable ordering is used for every M_i.

An ESOP (Exclusive or Sum Of Products) expression is a special case of ESCT expressions where the Maitra cells may not implement the logical OR and XOR functions.

Definition 3. *A minimal (or exact) expression of a switching function $f(x_1, \ldots, x_n)$ of n variables is defined as the ESCT expression which has the least number of terms comparing to every other ESCT expression for this function.*

Definition 4. *The ESCT weight $w(f)$ (or simply weight) of a switching function $f(x_1, \ldots, x_n)$ of n variables is defined as the number of complex terms in a minimal ESCT expression of f.*

Every two-variable switching function has weight equal to 1[21] . Functions with weight equal to one are called cascade-realizable.

Table 1: Cell index set

Cell index(r)	$F_r(x,y)$
1	$x+y$
2	$\overline{x}+y$
3	$\overline{x}y$
4	xy
5	$x \oplus y$
6	y

An ESCT expression can be directly mapped to a reversible wave cascade cellular architecture (Figure 1). A complex (Maitra) term is composed of Maitra cells ($r_{ij}, 1 \leq i \leq n, 1 \leq j \leq m$) which are two-input, one-output switching functions. The horizontal input to each cell is a function variable and the vertical one is the output of the previous cell in the same cascade (or the constant 0 in the case of the first cell of each cascade). The output of each cascade is connected to the XOR collector, thus obtaining the function $F(x)$.

Definition 5. *A $k * k$ generalized Toffoli gate is defined [5] as: $P_1 = A_1, P_2 = A_2, \ldots, P_{n-1} = A_{n-1}, P_n = f_{n-1}(A_1, A_2, \ldots, A_{n-1}) \oplus A_n$, where A_i are the inputs of the gate, P_i are the outputs of the gate and f_{n-1} is an arbitrary switching function of $n-1$ variables.*

In Ref [5] it was proved that a $k * k$ generalized Toffoli gate is reversible. A Maitra cascade, composed of n cells, plus the corresponding XOR collector cell, is a $(n+1) * (n+1)$ Toffoli gate (Figure 1) where: $P_1 = A_1 = X_1, \ldots, P_n = A_n = X_n, A_{n+1} = 0$ (or the output of another Toffoli gate) $, P_{n+1} = f_n(A_1, A_2, \ldots, A_n) \oplus A_{n+1}$ and $f_n(A_1, A_2, \ldots, A_n) = G_n(G_{n-1}(\ldots G_1(0, X_1), X_{n-1}), X_n)$ ($G_i, i \leq n$ are arbitrary boolean functions defined in Definition 1). It follows that a Reversible Wave Cascade is a reversible logic circuit because it is composed of reversible gates.

Is is obvious that the mapping of an ESCT expression to a reversible circuit is a trivial procedure. Algorithms that create ESCT expressions for switching functions are very attractive, since, in essence, they produce reversible circuits. Moreover, algorithms that create minimal ESCT expressions for switching functions are even more useful, because they produce smaller reversible circuits, resulting in, possibly, smaller production costs and of course less heat dissipation.

It has been proved that a Maitra cell doesn't need to implement every two-variable switching function. A set of only six functions is sufficient (complete set) [22][23] . Such a set can be seen in Table 1. The cascades using cells from this set are called Restricted Maitra Cascades and they will be used in the rest of this paper. Without loss of generality, when we mention Maitra cascades, we will refer to restricted Maitra cascades.

2.1 Representation

Definition 6. *The minterm representation (MT) of a switching function f with n variables is a bitvector of size 2^n where the i-th bit is 1 if the i-th minterm of f is 1.*

For the rest of this paper, the minterm representation of a switching function will be enclosed in brackets and will be expressed using hexadecimal notation.

A complex term is characterized by its cells, since its first input is the constant 0. Therefore, we can represent it by a series of cells (cell representation).

Definition 7. *The cell representation of a complex term is a series of numbers, corresponding to the series of indices of Maitra cells (Table 1) that belong to the complex term. The leftmost cell corresponds to the cell with the constant input 0 and the rightmost to the one closest to the XOR collector.*

For example, function $f = (x_1 \oplus x_2)x_3 + x_4 = [FF60]$ is cascade realizable. Using cells from the previously defined cell set, it can be represented as: (1541). The left-most cell is the one with one of its inputs hardwired to constant 0.

For the rest of this paper the cell representation of a complex terms will be enclosed in parenthesis.

2.2 Decompositions

Every boolean function can be expressed with the help of its subfunctions through relations, known as boolean decompositions (or expansions).

Definition 8. *Let $f(\boldsymbol{x})$ be a switching function and $\boldsymbol{x}$ the vector of its variables. Let x_i be one of the variables in the vector $\boldsymbol{x}$. Then $f(x_1, x_2, \ldots, x_i = 0, \ldots)$, $f(x_1, x_2, \ldots, x_i = 1, \ldots)$ and $\{f(x_1, x_2, \ldots, x_i = 0, \ldots) \oplus f(x_1, x_2, \ldots, x_i = 1, \ldots)\}$ are subfunctions of f, regarding variable x_i. For simplicity, in the rest of this paper, they will be referred as f_0, f_1 and f_2 respectively and x_i will be referred as x.*

A boolean function f can be expressed as (Shannon, Positive Davio and Negative Davio respectively):

- $f(\mathbf{x}) = \bar{x}f_0 \oplus xf_1$
- $f(\mathbf{x}) = xf_2 \oplus f_0$
- $f(\mathbf{x}) = \bar{x}f_2 \oplus f_1$

Additional expansions [17] for a boolean function f (in the form of ESCT expressions) are:

- $f(\mathbf{x}) = (x + f_2) \oplus (x \oplus f_1)$
- $f(\mathbf{x}) = (x + f_0) \oplus (x\bar{f}_1)$
- $f(\mathbf{x}) = (x\bar{f}_2) \oplus (x \oplus f_0)$
- $f(\mathbf{x}) = (x + \bar{f}_0) \oplus (\bar{x} + \bar{f}_1)$
- $f(\mathbf{x}) = (\bar{x} + \bar{f}_2) \oplus \bar{f}_0$
- $f(\mathbf{x}) = (x + \bar{f}_2) \oplus \bar{f}_1$
- $f(\mathbf{x}) = (\bar{x}\bar{f}_2) \oplus (x \oplus \bar{f}_1)$
- $f(\mathbf{x}) = (\bar{x}\bar{f}_0) \oplus (\bar{x} + f_1)$

- $f(\mathbf{x}) = (\bar{x} + f_2) \oplus (x \oplus \bar{f}_0)$

In Ref [18], the concept of Maitra cell class has been proposed. There are 3 different classes, the first composed by cells of index (1,3), the second by cells of index (2,4) and the third by cells of index (5,6). Using this concept, two definitions follow.

Definition 9. *Two complex terms are called relatives if their corresponding cells belong to the same Maitra cell class.*

Definition 10. *Let f be an n-variable switching function. By applying the Shannon and Davio decompositions on f, a ternary tree (ETDD-Exor Ternary Decision Diagram) is generated. This decomposition is applied until the constant 0 or 1 function is encountered or a leaf is obtained. This tree is named the generator tree.*

It has been proved [17] that the complement of a complex term M is also a complex term. Moreover if x_i is the cascade's most significant variable then $M \oplus x_i$ and $M \oplus \overline{x_i}$ are also complex terms [18].

For example, if $f(x_1, x_2, x_3, x_4) = (1233)$ (x_4 is the most significant variable), then: $f(x_1, x_2, x_3, x_4) \oplus \overline{x_4} = (1233) \oplus x_4 \oplus 1 = (1231) \oplus 1 = (2413)$.

3 Minimization theorems

The following theorems give the theoretical background for the algorithms presented in the next section.

Definition 11. *An equivalent expression (F_2) of an ESCT expression ($F_1 = \ldots \oplus P_i \oplus \ldots \oplus P_j \oplus \ldots$) for a switching function $f(x_1, \ldots x_n)$ (x_n is its most significant variable) is an expression produced by applying the following transformations to F_1:*

- $F_2 = \ldots \oplus \overline{P_i} \oplus \ldots \oplus \overline{P_j} \oplus \ldots$.
- $F_2 = \ldots \oplus (P_i \oplus x_n^*) \oplus \ldots \oplus (P_j \oplus x_n^*) \oplus \ldots$, *where* $x_n^* = x_n, \overline{x_n}$.

The above transformations can, also, be applied to pairs of descendants of f in its generator tree.

For example if $F_1 = (1234) \oplus (2343)$, then an equivalent expression of F_1 by complementing both terms is: $F_2 = \{(1234) \oplus 1\} \oplus \{(2343) \oplus 1\} = (2412) \oplus (1121)$.

Theorem 1. *At least one minimal expression of a single-output switching function $f(x_1, \ldots, x_n)$ with less than 6 variables ($n < 6$) can be obtained from the minimal expressions of f_0, f_1, f_2. This is performed by merging common complex terms between the minimal expressions of subfunctions of f.*

Proof. *The proof has been presented in [18].*

This previous theorem is based on the fact that we can produce minimal expressions for a switching function f of at most 5 variables by merging two complex terms, which are common between two minimal expressions of f's subfunctions, into one.

Theorem 2. *Let Q_1 be a minimal expression of switching function $f(x_1, \ldots, x_n)$ ($n < 6$), produced by the minimal expressions F_{i1}, F_{j1} of f_i, f_j respectively (f_i, f_j are subfunctions of f). An, equivalent to Q_1, minimal expression Q_2 of f can be obtained from two other minimal expressions F_{i2}, F_{j2} of f_i, f_j provided that these are equivalents to F_{i1}, F_{j1} respectively.*

Proof. The proof has been presented in [18].

The following corollary is a direct consequence of Theorem 2 and has been explained in [18].

Corollary 1. *At least one minimal expression of a single-output switching function $f(x_1, \ldots, x_n)$ with less than 6 variables ($n < 6$) can be obtained from the minimal expressions of f_0, f_1, f_2. This is performed by merging relative complex terms between the minimal expressions of subfunctions of f.*

Corollary 1 improves the results of Theorem 1. It denotes that we can use relative terms instead of common complex terms for merging, between the minimal expressions of f's subfunctions in order to produce a minimal expression for f.

One interesting observation, derived from Corollary 1, is that if a subfunction of a function is constant 0 or 1 then the weight of the function is equal to that of its non constant subfunctions. This is because every complex term in a minimal expression of its first non constant subfunction will be equal or complement (thus relative) to a complex term in a minimal expression of the other non constant subfunction.

Corollary 2. *Let f be a Boolean function and f_i, f_j, f_k its subfunctions. If f_i is constant (equal to 0 or 1) then $w(f) = w(f_j) = w(f_k)$.*

Proof. It is a direct consequence of Corollary 1.

Theorem 3. *A minimal expression of a switching function f of n variables with weight at most W, can be obtained by merging (xor summing) every switching function of weight at most $\lfloor W/3 \rfloor$, with the subfunctions of f.*

Proof. The proof has been presented in [19].

The following corollaries can be characterized as weaker forms of Theorem 3.

Corollary 3. *A minimal expression for a switching function f of at most 5 variables can be produced by merging every cascade realizable function with the subfunctions of f.*

Proof. It is a direct consequence of Theorem 3, since the weight of a 5 variable switching function is as most 6 (since the weight of a 4-var subfunction is at most 3 [24]).

Corollary 4. *A minimal expression for a switching function f of 6 variables can be produced by merging every switching function with weight at most 3 with the subfunctions of f.*

Proof. It is a direct consequence of Theorem 3, since the weight of a 6 variable switching function is as most 12 (since the weight of a 5-var subfunction is at most 6).

4 Variable Reordering

In ESOP expressions, which are a subset of ESCT and have been widely studied in the related literature, variable ordering of the input variables for a switching function is not important. Regardless of the variable ordering the weight of the function is the same. However, this is not true for ESCT expressions. Consider the following example:

Let f be a single-output, 3-input switching function: $f(x_1, x_2, x_3) = (x_1 \oplus x_2)x_3$. If we consider that x_3 is the most significant input variable and x_1 the least significant then f has weight 1, with minimal expression: (154). However if we consider x_1 as its most significant variable and x_2 its least significant, then the same function has weight 2, with minimal expression: $(614) \oplus (146)$.

It is obvious, from this previous example, that the ESCT-weight of a switching function is different depending on the ordering of its input variables. Therefore, it is important to be able to find the best possible ordering, the one that gives the minimum weight, before trying to find minimal expressions for a switching function.

In this paper we propose a heuristic algorithm that finds such good variable orderings. Algorithm VRORDER uses functional decomposition and is based on the idea expressed by Corollary 2. If a function has one of its subfunctions constant (0 or 1) then its weight is equal to the weight of its non-constant subfunctions and therefore the number of complex terms does not increase as we move towards the root of the generator tree (our initial function).

This previous observation shows that the more constant subfunctions there are inside the generator tree of a switching function f, the more probable is that the weight of f will be smaller. Moreover it is important that those constant subfunctions are closer to the root of the generator tree.

5 Experimental Results

All the previously mentioned algorithms (Min1, EMin1, XMin6, VRORDER) have been implemented on an AMD ATHLON XP 1.8+ processor with the LINUX operating system. In the section and when the VRORDER algorithm will be implemented, experimental results will be given of VRORDER in conjunction with actual minimization algorithms like EMin1.

References

[1] T. Sasao "Switching Theory For Logic Synthesis", Kluwer Academic Publishers, 1999.

[2] R.L. Ashenhurst: The decomposition of switching functions, Ann. Computation Lab. Harvard University, vol. 29, pp. 74-116, 1959.

[3] H.A. Curtis: A new approach to the design of switching circuits, Princeton, N.J. Van Nostrand, 1962.

[4] J.P. Roth, R.M. Karp: Minimization over Boolean graphs, IBM Journal, April 1962, pp. 227-238.

[5] A. Mishchenko, M. Perkowski, "Logic Synthesis of Reversible Wave Cascades",International Workshop on Logic And Synthesis 2002, New Orleans, Louisiana, June 4-7, 2002.

[6] C. Bennet "Logical Reversibility of Computation", IBM Journal of Research and Development, 17, 1973, pp. 525-532.

[7] J. Preskill "Lecture notes in quantum computing", http://www.Theory.caltech.edu/ preskill/ph229

[8] A. Sarabi, N. Song, "A comprehensive approach to logic synthesis and physical design for two-dimensional logic arrays", DAC 1994, 321-326.

[9] I. Schaefer, M. Perkowski, H. Wu "Multilevel logic synthesis for cellular FPGAs based on orthogonal expansions", Proc, IFIP WG 10.5 Workshop on Applications of the Reed-Muller Expansion in Circuit Design, Sept. 1993, Hamburg, Germany, pp. 42-51.

[10] G. Lee "Logic synthesis for celullar architecture FPGA using BDD", ASP-DAC 97, pp 253-258 Jan 1997.

[11] G. Lee, R. Drechsler "ETDD-based Synthesis of term-based FPGAs for incompletely specified boolean functions", ASP-DAC 1998.

[12] P. Lindgren, R. Drechsler, B. Becker "Look-up table FPGA synthesis from minimized multi-valued pseudo kronecker expressions",ISMVL 98.

[13] N. Song, M. Perkowski "Minimization of exclusive sums of multi-valued complex terms for logic cell arrays", ISMVL 98, p 32.

[14] N. Song, M. Perkowski "A new approach to and/or/exor factorization for regular arrays", Proc. 1998 Euromicro, pp. 269-276, Vasteras, Sweden, August 25-27, 1998.

[15] M. Helliwell, M. Perkowski "A fast algorithm to minimized multi-output mized polarity generalized reed-muller forms", Proc 25th ACM IEEE Conference on Design Automation, Atlantic city, New Jersey, USA, pp. 427-432, 1988.

[16] D. Maslov, G. Dueck "Garbage in Reversible Designs of Multiple Output Functions", Proc RM2003, March 10/11 2003, Germany.

[17] D. Voudouris, S. Stergiou, G. Papakonstantinou "Minimization of reversible wave cascades", IEICE Trans. on Fund., Vol E88-A, No. 4, pp. 1015-1023, 2005/04.

[18] D. Voudouris, G. Papakonstantinou "Maitra Cascade Minimization", 6th IWSBP, 2005, Freiberg (Sachsen), Germany.

[19] D. Voudouris, M. Kalathas, G. Papakonstantinou "Decomposition of Multi-output Boolean Functions", HERCMA 2005, Athens, Hellas.

[20] D. Voudouris, M. Sampson, G. Papakonstantinou "Exact ESCT Minimization for functions of up to six input variables", accepted for publication to Integration, The VLSI Journal, Elsevier.

[21] K.K. Maitra "Cascaded switching networks of two-input flexible cells" IRE Trans. Electron. Comput., pp, 136-143, 1962.

[22] R. C. Minnick, "Cutpoint cellular logic" IEEE Trans. Electron. Comput., vol. EC-13. pp. 685-698, Dec, 1964.

[23] G. Papakonstantinou "Cascade Transformation", IEEE Transactions on computers, Jan 1976.

[24] G. Papakonstantinou, F. Gritzali "Modulo-2 expressions of switching functions", Electronic Letters, 13(1977).

[25] G. Papakonstantinou "Synthesis of cutpoing cellular arrays with exclusive-OR collector row", Electronic Letters, 13(1977).

[26] T. Hirayama, T. Sato, Y. Nishitani "Minimizing and-exor expressions of some benchmark functions", 6th International Workshop on Applications of Reed Muller Expansion in Circuit Design, March 2003.