International Journal of Computer Theory and Engineering, Vol. 1, No. 5, December, 2009
1793-8201

Properties of Autocorrelation Coefficients for Single-Output Switching Functions

J. E. Rice, *Member, IEEE*, J. C. Muzio, *Senior Member, IEEE*, N. A. Anderson and R. Jansen

***Abstract*—A variety of mathematical transforms have traditionally been used in various logic synthesis applications. This paper investigates the use of the autocorrelation transform:**

$$C(\tau) = \sum_{v=0}^{2^n-1} f(v) \bullet f(v \oplus \tau)$$

Properties of the coefficient resulting from the application of this transform to switching functions are examined and detailed, including properties to identify symmetries and decompositions. The potential uses in logic synthesis of these properties and other observations based on the autocorrelation coefficients are explored, with emphasis on proofs as mathematical justification of the theorems relating the observed properties of the coefficients to properties of the underlying switching functions.

***Index Terms*—high level synthesis, logic synthesis, transforms, Boolean logic, digital logic, function representations.**

I. INTRODUCTION

The autocorrelation transform has been used in various areas including optimization and synthesis of combinational logic [1], variable ordering for Binary Decision Diagrams [2], and to compute the estimate *C(f)* of a function's complexity [1, 3]. Use of the autocorrelation transform, however, has been limited. This may be due to either the fact that their computation is not trivial, or that little has been known of the transform's properties. To address this first problem new computation methods have recently been introduced by Rice, Muzio and Serra [4, 5] as well as by Stankovic and Karpovsky [6]. To address the second we devote this work to an explanation of the theoretical use of the autocorrelation transform in the identification of properties that may be useful in activities such as logic synthesis of Boolean functions. Effort is made to provide proofs to both justify and explain how we propose these coefficients be used, and although this work does not yet include experimental results based on these theories, future work in this direction is currently underway.

We first present the definition and an explanation of the autocorrelation transform. We then introduce several theorems relating the values of the resulting autocorrelation coefficients to properties of the underlying switching function. A number of potential applications for these theorems are presented, and future directions for this work are discussed.

Manuscript received June 9, 2009. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

Dr. Rice is with the Dept. of Mathematics and Computer Science at the University of Lethbridge, Lethbridge, AB, Canada (phone: 403-329-2783; fax: 403-317-2882; email: j.rice@uleth.ca).

Dr. Muzio is with the Dept. of Computer Science at the University of Victoria, Victoria, BC, Canada (email: jmuzio@cs.uvic.ca).

Contributions by Mr. Anderson and Ms. Jansen were made while they were undergraduate research assistants to Dr. Rice at the University of Lethbridge.

II. BACKGROUND

The autocorrelation transform is a special case of the correlation transform, which is defined in [3] as follows:

$$B^{fg}(\tau) = \sum_{v=0}^{2^n-1} f(v) \cdot g(v \oplus \tau). \quad (1)$$

If *f* and *g* are the same function then this becomes the autocorrelation transform, also called the cross-correlation, or convolution function. The superscript is generally omitted when referring to the autocorrelation transform. By convention $B(\tau)$ is evaluated with *f* in the usual Boolean domain of {0,1}. If {+1,-1} encoding is used then the resulting autocorrelation coefficients are denoted as $C(\tau)$:

$$C(\tau) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus \tau). \quad (2)$$

It is straightforward to derive the relationship between *B* and *C*, namely:

$$C(\tau) = 2^n - 4k + 4B(\tau). \quad (3)$$

In this equation, $k=B(0)$, which is also the number of true minterms in the function. We should also point out that the $+$ operator is used to indicate the OR operator when used in logical expressions, and to indicate arithmetic addition when used in arithmetic equations such as (3) or summations.

The derivation is based on this relationship between {0,1} encoded outputs, labeled as z_i, and {+1,-1} encoded outputs, labeled as y_i: $y_i = -2z_i+1$.

Although the same information is present in both $C(\tau)$ and $B(\tau)$ there are some patterns that are more easily identifiable when using {+1,-1} encoding, and vice versa. Thus it is useful to be able to use either encoding, particularly for analysis.

It is useful to present some additional notation to aid in the understanding of this paper.

- τ and τ' indicate values ranging from 0 to 2^n-1. τ_α is used to indicate one such value. These are usually expressed as a binary expansion.

- $|\tau|$ is the weight, or the number of ones in the binary expansion of τ. If $|\tau| = j$ then $B(\tau)$ and $C(\tau)$ are said to be j^{th} order coefficients.
- The variable ordering $x_n, \ldots, x_1$ is used through-out. Thus a coefficient $B(001)$ or $C(001)$ is the first order coefficient corresponding to x_1.
- τ_i refers to a value whose binary expansion contains a 1 in the i^{th} bit, while the remaining n-1 bits are 0.
- $\tau_{i\alpha}$ refers to a set of values for which the binary expansion contains a 1 in the i^{th} bit while the remaining n-1 bits have the value $\alpha \in \{0,\ldots 2^{n-1}\}$. . $\tau_{i\alpha}$ refers to a set of values for which the binary expansion contains a 0 in the i^{th} bit while the remaining n-1 bits have the value α.
- k refers to the number of true minterms in a function.
- we use the terms *true minterm* or *positive minterm* to refer to a combination of assignments to the input values that results in a true output *e.g. f(v)* = 1, and the term *false minterm* to refer to an input assignment that results in a false output *e.g. f(v)* = 0. The term *minterm* may refer to either type of input assignment.

III. Observations

There are a number of restrictions on the values of both the {0,1} and {+1,-1} autocorrelation coefficients. These may or may not be useful in a logic synthesis context, but provide an easy test for correctness and validity, and lend some insight into the behaviour of this transform and its resulting coefficients.

The following observations are clear from the definition of the autocorrelation transform:

- $B(\tau) \in \{0,\ldots, 2^n\}$ and $C(\tau) \in \{-2^n,\ldots, 2^n\}$ $\forall\ \tau \in \{0,\ldots, 2^n\text{-}1\}$,
- both $B(\tau)$ and $C(\tau)$ are even $\forall\ \tau \neq 0$, and
- $B(\tau) \leq B(0)$ $\forall\ \tau \neq 0$ and $B(0) = k$, and $C(\tau) \leq C(0)$ $\forall\ \tau \neq 0$ and $C(0) = 2^n$.

The final observation requires further explanation:

- a function may have at most 2^{n-1} negative values for $C(\tau)$.

Let us define a function *f(X)* for which there are 2^{n-1} negative coefficients. Without loss of generality we assume that for this function every value of $C(\tau)$, $2^{n-1} \leq \tau \leq 2^n\text{-}1$, is negative. If $2^{n-1} \leq \tau \leq 2^n\text{-}1$ then in the autocorrelation equation $0 \leq v \leq 2^{n-1}\text{-}1 \Rightarrow 2^{n-1} \leq v \oplus \tau \leq 2^n\text{-}1$ and $2^{n-1} \leq v \leq 2^n\text{-}1 \Rightarrow 0 \leq v \oplus \tau \leq 2^{n-1}\text{-}1$.

In other words, in computing each of the negative coefficients we are matching a minterm from the top half of the function with one from the bottom half of the function, assuming that minterms are ordered numerically from $x_n x_{n-1}\ldots x_1 = 00\ldots0$ to $x_n x_{n-1}\ldots x_1 = 11\ldots1$. For any one of the designated coefficients to be negative, there must be $2^{n-2}+1$ of the values $0 \leq v \leq 2^{n-1}\text{-}1$ negative if the values in $2^{n-1} \leq v \leq 2^n\text{-}1$ are positive, or vice versa. However, this results in the remaining 2^{n-1} coefficients having positive values. Thus there can be at most 2^{n-1} negative autocorrelation coefficients.

Theorems 1 and 2 provide two further observations about the values of the autocorrelation coefficients.

Theorem 1

$$\sum_{\tau=0}^{2^n-1} B(\tau) = k^2. \tag{4}$$

The proof of this theorem relies on the following Lemma:

Lemma 1

$$\sum_{\tau=1}^{2^n-1} B(\tau) = 2\binom{k}{2}. \tag{5}$$

$\binom{k}{2}$ is the number of pairings of the minterms as computed in the summation of the autocorrelation coefficients. This is then multiplied by two to produce all possible pairings in the form *i, j* and *j, i*.

Proof: Using Lemma 1 the sum of all of the {0,1} autocorrelation coefficients is as follows:

$$\begin{aligned}\sum_{\tau=0}^{2^n-1} B(\tau) &= B(0) + 2\binom{k}{2} \\ &= k + 2\frac{k(k-1)}{2} \\ &= k^2.\end{aligned}$$

n

Theorem 2 $C(\tau) = 2^n - 4m$ $\forall\ \tau$ (6)

where $m \in k, k\text{-}2, \ldots, 0$ for even values of k and $m \in k, k\text{-}2, \ldots, 1$ for odd values of k, k being the number of true minterms in the function or the number of false minterms in the function, whichever is fewer.

Proof: The largest possible number of mismatch pairs, that is, negative contributions to the total coefficient value is $-2k$. The remaining pairs, which of necessity result in positive contributions to the coefficient value is $2(2^{n-2}\text{-}k)$. Thus the total value for the coefficient is

$$\begin{aligned}C(\tau) &= -2k + 2(2^{n-1} - k) \\ &= -2k + 2^n - 2k \\ &= 2^n - 4k.\end{aligned}$$

n

However, this assumes that all positive minterms will pair with false minterms and vice versa. This is not the case; for some coefficients a subset of false minterms may pair with other false minterms. Each time a false minterm is paired with another false minterm the number of negative contributions is reduced by 2, leading to the equation in Theorem 2.

IV. General Properties

This section introduces theorems that relate particular patterns in the autocorrelation coefficients to underlying properties of the switching function. We propose in future work to utilize these patterns in identifying properties in the switching functions that may be useful in logic synthesis. For example, identification of variables of which the function is independent may reduce the problem size to something more manageable, while determining the possible existence of symmetries is known to be a useful technique in logic synthesis [7].

A. Trivial Functions

A trivial function is one in which all output values of the function are 1, or all output values of the function are 0, assuming {0,1} encoding.

Theorem 3 *$C(\tau) = C(\tau') \; \forall \; \tau$ and $\tau' \in \{0,\ldots, 2^n\text{-}1\}$ and if and only if $f(X)=1$ or $f(X)=0$.*

Proof: If all the coefficients are equal, they must all have the value 2^n as the coefficient $C(0)$ always has this value. Based on this, if all of the coefficients have equal value, then this implies that the function matches itself at every value of τ. This can only occur if the function consists entirely of true minterms, or entirely of false minterms. **n**

B. Identifying Redundant Variables

The following two theorems may be applied to identify redundant variables in a function. Theorem 4 describes the situation that occurs when a function does not depend on one of the input variables. Theorem 5 describes the situation when a function may depend on only one of the input variables.

Theorem 4 *A function $f(X)$ is independent of input variable x_i if and only if $C(\tau_i)=2^n$.*

Proof: Without loss of generality let us define a function $f(X)$ that is independent of x_n. By definition, $f(0, x_{n-1},\ldots, x_1) = f(1, x_{n-1},\ldots, x_1)$.. Then

$$\begin{aligned} C(\tau_n) &= \sum_{v=0}^{2^n-1} f(v)\times f(v\oplus\tau_n) \\ &= \sum_{v=0}^{2^{n-1}-1} f(v)\times f(v\oplus\tau_n) \\ &\quad + \sum_{v=2^{n-1}}^{2^n-1} f(v)\times f(v\oplus\tau_n) \end{aligned}$$

Let us define the range 0 to 2^{n-1}-1 as A and 2^{n-1} to 2^n-1 as B. Then $v\in A \Rightarrow v\oplus\tau_n \in B$ and $v\in B \Rightarrow v\oplus\tau_n \in A$. Since the function is defined to have $f(A) = f(B)$ then

$$\begin{aligned} C(\tau_n) &= \sum_{v=0}^{2^{n-1}-1} f(v)\times f(v\oplus\tau_n) \\ &\quad + \sum_{v=2^{n-1}}^{2^n-1} f(v)\times f(v\oplus\tau_n) \\ &= \sum_{v=0}^{2^{n-1}-1} 1 + \sum_{v=2^{n-1}}^{2^n-1} 1 \\ &= 2^n. \end{aligned}$$

To prove the second part of the theorem we define (without loss of generality) a function $f(X)$ for which $C(\tau_n)=2^n$. This is only possible if $f(v) = f(v\oplus\tau_n) \; \forall \; v$. This implies that $f(1, x_{n-1},\ldots, x_1) = f(0, x_{n-1},\ldots, x_1)$, indicating that $f(X)$ is not dependent on x_n. **n**

Theorem 5 *A function $f(X)$ has 2^{n-1} autocorrelation coefficients $C(\tau)=2^n$ (including $C(0)$) and the remaining 2^{n-1} coefficients $C(\tau')=-2^n$ if and only if the function is dependent on only one of its input variables OR is related to such a function through the application of one or more invariance operations.*

A function that is dependent on only one of its input variables has exactly 2^{n-1} true minterms. However, this is not the only situation where a function can have exactly 2^{n-1} true minterms. As discussed in [8] any function that is related to this type of function through the application of one of four invariance operations will have 2^{n-1} true minterms. The theorem above and the proof following refer to any of these types of functions.

Proof: Without loss of generality let us define $f(X)=x_1$ where x_1 is the lowest order bit of the input X. Then if τ is an odd number the binary expansion of τ contains a 1 in the lowest order bit, and then by definition $f(v) = \overline{f(v\oplus\tau)}$ where the bar indicates the Boolean not operator. Then

$$C(\tau) = \sum_{v=0}^{2^n-1} 1\times -1 = -2^n.$$

Similarly if τ' is an even number, then the binary expansion contains a 0 in the lowest order bit and by definition $f(v) = f(v \oplus \tau')$. Then

$$C(\tau') = \sum_{v=0}^{2^n-1} (-)1\times(-)1 = 2^n.$$

Given autocorrelation coefficients of the pattern described above the function must be dependent on only one of the input variables (or related to such a function). Without loss of generality we assume that $C(\tau')=2^n$ where τ' is even and $C(\tau)=-2^n$ where τ is odd. $C(\tau')=2^n$ where τ' is even indicates that the function matches up two false or two true minterms for every product in the summation. Additionally every product being computed is comparing two inputs for which x_1 remains unchanged. Moreover, $C(\tau)=-2^n$ where τ is odd indicates that the function matches a false minterm with a true minterm for every product in the summation, and that every product is matching a pair of inputs for which x_1 varies. Based on this we can determine that the function must be dependent only on x_1. **n**

C. Dissimilar Minterms

The following are three theorems that allow a designer to identify a sparse (or the inverse) function from the values of the function's autocorrelation coefficients. A sparse function is one in which a majority of the input values result in a particular output, *e.g.* 0, and the remaining minority (possibly only one) result in the other possible output value. The first two theorems detail two specific cases: functions that possess one and only one true minterm (or the inverse) and functions that possess only two true minterms (or the inverse).

Theorem 6 *A function $f(X)$ has exactly one dissimilar minterm if and only if $C(\tau)=2^n\text{-}4 \; \forall \; \tau \neq 0$.*

The proof is given in the Appendix.

The corollary for the {0,1} encoding can be shown by applying (3) to the theorem above. The general result is as follows:

Corollary 1 *A function $f(X)$ has exactly one dissimilar minterm if and only if $B(\tau)=k\text{-}1$.*

It should be pointed out that this general $B(\tau)$ are quite limited. This is because for a function to have exactly one dissimilar minterm then either $k=2^n-1$, in which case $B(\tau)=2^n-2\ \forall\ \tau \neq 0$, or $k=1$, which results in $B(\tau)=0\ \forall\ \tau \neq 0$.

This type of analysis of the division of true and false minterms in the function can be extended to the situation with d dissimilar minterms. Initially we look at the case of $d=2$ before giving the general result.

Theorem 7 *A function f(X) has exactly two dissimilar minterms if and only if*

$C(0) = 2^n$,

$C(\tau_\alpha)=2^n$, *and*

$C(\tau) = 2^n-8\ \forall\ \tau, \tau_\alpha \neq 0$ *and* $\tau \neq \tau_\alpha$.

The proof is given in the Appendix.

Corollary 2 *A function f(X) has exactly two dissimilar minterms if and only if*

$B(0) = B(\tau_\alpha) = k$ *and*

$B(\tau) = k-2\ \forall\ \tau, \tau_\alpha \neq 0$ *and* $\tau \neq \tau_\alpha$.

Again, although Corollary 2 states a general result, in practice the values are limited to the following:

- $B(0) = B(\tau_\alpha) = 2$ and $B(\tau) = 0$, or
- $B(0) = B(\tau_\alpha) = 2^n-2$ and $B(\tau) = 2^n-4$.

It should also be noted that this pattern of coefficients indicates that the function is either itself degenerate or is related through the application of the autocorrelation invariance operators [8] to a degenerate function.

Theorem 8 *A function f(X) has d dissimilar minterms if and only if the autocorrelation coefficients have the following properties:*

- $C(0) = 2^n$,
- for $\binom{d}{p}$ $p \in 2,4,6,\ldots,d$ (or $2,4,6,d-1$ if d is odd) $C(\tau) = 2^n - 4d + 4p$, and
- for the remaining coefficients, $C(\tau) = 2^n - 4d$.

Again, the proof is given in the Appendix.

D. Identification of Exclusive-OR Logic

In some approaches to logic synthesis it is useful to identify a decomposition of the function that utilizes the Boolean $\oplus$ (exclusive-or) operator [9].

Theorem 9 $C(\tau_i) = -2^n$ *if and only if the function f(X) has a decomposition* $f(X) = h(X) \oplus x_i$ *where h(X) is independent of* x_i.

If no first order coefficients meet the requirements for the presence of this decomposition, we then go on to include the second order coefficients in the examination. This is described in Theorem 10.

Theorem 10 $C(\tau_i) = C(\tau_j) = C(\tau_{ij}) = 0$, $i \neq j$ *if and only if the function f(X) can be decomposed into* $h(X) \oplus g(X)$ *where* $g(X) = x_i * x_j$, $* \in \{\wedge, \vee\}$, *and h(X) is independent of both* x_i *and* x_j.

Theorem 10 can be further extended to functions where $g(X)=\vee(x_i \ldots x_{i+m})$ or $g(X)=\wedge(x_i \ldots x_{i+m})$, $i \in \{1 \ldots n\}$ and $i + m \leq n$:

Theorem 11 *If* $f(X)$ *can be decomposed into* $h(X1) \oplus g(X2)$ *where* $X1 \cup X2 = X$ *and* $X1 \cap X2 = \varnothing$ *then* $C^{ff}(\tau_{X2}) = C^{gg}(\tau_{X2})$.

This theorem describes a situation in which a function $f(X)$ is known to have a decomposition of the format $h(X) \oplus g(X)$ where $h(X)$ is independent of all the variables in g. Thus $\phi == \{\varnothing\}$. In this case autocorrelation coefficients that are related to g's variables for both functions f and g will then be equivalent. Since it is possible to construct examples such that $f(X) = h(X1) \oplus g(X2)$, $C^{ff}(\tau_{X2}) = C^{gg}(\tau_{X2})$, and $X1 \cap X2 \neq \{\varnothing\}$, the presence of such a pattern $C^{ff}(\tau_{X2}) = C^{gg}(\tau_{X2})$ is not strong enough to uniquely identify all exclusive-or based decompositions of this type. However, given two functions $f(X)$ and $g(X)$ it is always possible to construct $h(X)$ such that $g(X) \oplus h(X)$ since $\oplus$ is reversible and $h(X)$ can be determined by finding $f(X) \oplus g(X)$. Thus using a known library it would be possible to perform a fast determination, using the autocorrelation coefficients, of whether a mutually exclusive decomposition was likely, and then use the above to construct the second function of the decomposition. Proofs for each of these theorems are given in the Appendix.

V. USES

The above properties have been put to a variety of uses, including determining three-level decompositions [5], variable ordering and optimization of binary decision diagrams (BDDs) [2, 10], and in classifying Boolean logic functions [8]. Further work is progressing on additional uses, such as developing heuristics for Kronecker decision diagram (KDD) decomposition selections and for determining whether a BDD, KDD or functional decision diagram (FDD) is a better representation for a function [11]. Below we investigate the use of autocorrelation coefficients in identifying properties such as symmetries, linearity and self-duality of functions.

A. Totally Symmetric Functions

There are a number of different types of symmetries. We begin with the most restrictive symmetry. A Boolean function is said to be totally symmetric if the output is unchanged by any permutation of the inputs to the function. For example, $f = x_1+x_2+x_3$ is totally symmetric, as is the majority function $f = x_1x_2+x_2x_3+x_1x_3$. We also discuss a recently introduced type of symmetry termed *antisymmetries* [12]. An antisymmetry occurs when permuting all or a subset of variables results in the exact inverse of the original function.

Theorem 12 *If a function f(X) is totally (anti)symmetric then all {+1,-1} autocorrelation coefficients for any given order will be equal within the order. This may be written as* $C(\tau) = C(\tau')\ \forall\ \tau, \tau'$ *such that* $|\tau|=|\tau'|$.

Proof: Work in [8] showed that permuting any two variables j and k results in exchanging the values of the coefficients $C(\tau_{j\alpha})$ and $C(\tau_{k\alpha})$. Since a function symmetric in two variables j and k by definition will not change if j and k are permuted then the autocorrelation coefficients will also not change — the function remains the same. Thus for $C(\tau_{j\alpha})$ and $C(\tau_{k\alpha})$ to be exchanged and yet no change to occur, we must have $C(\tau_{j\alpha}) = C(\tau_{k\alpha})$. A function that is totally symmetric will not change for any permutation of its

variables, so $C(\tau_{1\alpha}) = C(\tau_{2\alpha}) = ... = C(\tau_{n\alpha})$, assuming the variables are numbered from 1 to n. We can express this as $C(\tau) = C(\tau')$ where $|\tau|=|\tau'|$ since α can take on any value and the property still holds. **n**

We note that this implies that there are non-totally-symmetric functions with coefficients of this pattern. An example of this is given below in Subsection V-B. However, such a function must be related through the application of one or more invariance operators to some totally symmetric function. We expand upon this in Section VI. We note also that the above theorem includes totally antisymmetric functions, since the negation of a function does not affect the {+1,-1} autocorrelation coefficients.

Symmetry	Definition
$E\{x_{n-1}, x_n\}$	$f(x_1,..., x_{n-2}, 0,0) = f(x_1,..., x_{n-2}, 1,1)$
$N\{x_{n-1}, x_n\}$	$f(x_1,..., x_{n-2}, 0,1) = f(x_1,..., x_{n-2}, 1,0)$
$S\{x_n \mid x_{n-1}\}$	$f(x_1,..., x_{n-2}, 1,0) = f(x_1,..., x_{n-2}, 1,1)$
$S\{x_n \mid x'_{n-1}\}$	$f(x_1,..., x_{n-2}, 0,0) = f(x_1,..., x_{n-2}, 0,1)$

B. Partially Symmetric Functions

A slightly less restrictive form of symmetry is that of partial symmetry. A Boolean function is said to be partially symmetric, or possess a partial symmetry if the output is unchanged by any permutation of some subset of the inputs to the function.

Theorem 13 *If a function f(X) is partially symmetric in a subset of its input variables x_{i_1} ,...,x_{i_m} then the autocorrelation coefficients $C(\tau_{j\alpha})$ will have equal values for all $j \in \{i_1,...,i_m\}$.*

The same reasoning as used above for Theorem 12 can be used here. Permuting any m variables x_{i_1} through x_{i_m} results in exchanging the values of the coefficients $C(\tau_{i_{1_\alpha}\alpha})$ through $C(\tau_{i_{m_\alpha}\alpha})$. However, the function does not change, by definition, and so coefficients $C(\tau_{i_{1_\alpha}\alpha})$ through $C(\tau_{i_{m_\alpha}\alpha})$ must be equal. For example, the function $f(X) = \overline{x}_1\overline{x}_2\overline{x}_3 + \overline{x}_3x_4 + \overline{x}_1x_4 + x_2x_3 + x_1x_2 + x_1x_3$ is partially symmetric in $x_1x_2x_3$. Although it appears that the product $\overline{x}_2x_4$ is missing it is unnecessary as it is covered by the other products. The autocorrelation coefficients for this function are given in Table I. Of note are the sets of values (for τ) 1000, 0100 and 0010, and 1001, 0101 and 0011 which illustrate the theorem above.

C. Functions with Symmetries of Degree Two

A third type of symmetry is a symmetry of degree two. This is a partial symmetry in which two sub-functions of the original function are identical and also are independent of two of the function's variables. Symmetries of degree two are identified by finding patterns where $f(x_1,..., a,..., b,..., x_n) = f(x_1,..., c,..., d,..., x_n)$, $a,b,c,d \in \{0,1\}$. Equivalence (E), non-equivalence (N) and single-variable (S) symmetries as defined by Hurst, Miller and Muzio are all types of symmetries of degree two, and are defined in Table II [13]. Without loss of generality these definitions label the two variables of interest as n and n-1. x' refers to the the inverse of x.

Antisymmetries can also be extended to the symmetries of degree two. For instance, an anti-equivalence symmetry is usually denoted $\overline{E}\{x_{n-1}, x_n\}$.

Theorem 14 *A function f(X) with some type of (anti)symmetry of degree two will have autocorrelation coefficient values as follows:*

$E\{x_i, x_j\}$ *or* $N\{x_i, x_j\} \rightarrow C(\tau_{i\alpha}) = C(\tau_{j\alpha})$,

$S\{x_j \mid x_i\}$ or $S\{x_j \mid \overline{x}_i\} \rightarrow C(\tau_{i\alpha}) = C(\tau_{ij\alpha})$, and

$S\{x_i \mid x_j\}$ or $S\{x_i \mid \overline{x}_j\} \rightarrow C(\tau_{j\alpha}) = C(\tau_{ij\alpha})$.

Proofs for these are given in [14].

τ	$C(\tau)$	τ	$C(\tau)$	τ	$C(\tau)$	τ	$C(\tau)$
0000	16	0100	4	1000	4	1100	12
0001	4	0101	4	1001	4	1101	4
0010	4	0110	12	1010	12	1110	4
0011	4	0111	4	1011	4	1111	4

TABLE I: {+1,-1}AC COEFFICIENTS FOR THE PARTIALLY SYMMETRIC FUNCTION $f(X) = \overline{x}_1\overline{x}_2\overline{x}_3 + \overline{x}_3x_4 + \overline{x}_1x_4 + x_2x_3 + x_1x_2 + x_1x_3$.

TABLE II: DEFINITIONS AND NOTATION FOR EQUIVALENCE, NON-EQUIVALENCE, AND SINGLE-VARIABLE SYMMETRIES.

D. Is it Possible to Determine Symmetries from the Autocorrelation Coefficients?

Hurst, Miller and Muzio provide tests based on a function's spectral coefficients that will ascertain whether or not the function possesses a particular symmetry [13]. However, as indicated by the example in Table I, the autocorrelation coefficients cannot be used in the same way. This can be explained by examining the spectral symmetry tests, as described in Table III. The notation used in this table is as follows:

- S^0 includes all spectral coefficients that involve neither of x_i or x_j,
- S^1 includes all spectral coefficients that involve x_i but not x_j,
- S^2 includes all spectral coefficients that involve x_j but not x_i, and
- S^3 includes all spectral coefficients that involve both x_i and x_j.

The spectral coefficients are computed using

$$T^n \cdot Y = S. \tag{7}$$

For example, for a n=3 Boolean function,

$$T^n = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix},$$

Y is the output vector of the function, for example

$$Y = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \end{bmatrix} \begin{matrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7, \end{matrix}$$

and *S* is the resulting spectral coefficients. Using the sample function from above, the coefficients would be

$$S = \begin{bmatrix} 2 \\ 2 \\ 2 \\ 2 \\ -2 \\ -2 \\ -2 \\ 6 \end{bmatrix} \begin{matrix} s_0 \\ s_1 \\ s_2 \\ s_{12} \\ s_3 \\ s_{13} \\ s_{23} \\ s_{123}. \end{matrix}$$

Examination of the spectral symmetry tests for three variables illustrates that if $\begin{bmatrix} s_2 \\ s_{12} \end{bmatrix} = \begin{bmatrix} s_3 \\ s_{13} \end{bmatrix}$ then the function must possess $N\{x_2, x_3\}$. Similarly, if $\begin{bmatrix} s_2 \\ s_{12} \end{bmatrix} = \begin{bmatrix} -s_3 \\ -s_{13} \end{bmatrix}$ then the function must possess $E\{x_2, x_3\}$. The notation used here for labeling of coefficients is as illustrated in the example above.

In the autocorrelation coefficients, this distinction is lost. This brings to question the following situation. If $\begin{bmatrix} s_2 \\ s_{12} \end{bmatrix} = \begin{bmatrix} -s_3 \\ s_{13} \end{bmatrix}$ then the autocorrelation coefficients will still be equal; however, the symmetries do not exist. The same holds true if $s_2 = s_3$ and $s_{12} = -s_{13}$. Therefore it is not possible to determine if a function has a particular equivalence, nonequivalence or single variable symmetry solely by examining the autocorrelation coefficients. The same holds true for totally and partially symmetric functions.

E. Self-Dual & Self-Anti-Dual Functions

Definition 5.1 *The dual of a function $f(x_1, x_2, \ldots, x_n)$ is $\overline{f}(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n)$ and is denoted by f^d [9].*

f^d is obtained first by replacing each literal x_i with $\overline{x}_i$ and then by complementing the function. A self-dual function is a function such that $f = f^d$. There are $2^{2^{n-1}}$ self-dual functions of *n* variables. A self-anti-dual function is a function such that $f = f(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_n)$.

Theorem 15 *A function f(X) will have $C(2^n\text{-}1) = -2^n$ if and only if it is a self-dual function. Similarly, a function will have $C(2^n\text{-}1) = 2^n$ if and only if it is a self-anti-dual function.*

Proof: If a function is self-dual, then by definition $f(X) = \overline{f}(\overline{X})$, which can be rewritten as $f(X) = \overline{f}(X \oplus 2^n - 1)$. Using {+1,-1} notation $f(X) \bullet \overline{f}(X) = -1$. Then by definition

$$C(2^n - 1) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus 2^n - 1)$$

and thus

$$= \sum_{v=0}^{2^n-1} f(v) \cdot \overline{f}(v)$$

$$= -2^n.$$

TABLE III: SPECTRAL SYMMETRY TESTS FOR SYMMETRIES IN $\{X_{N-1}, X_N\}$. X_I' REFERS TO THE INVERSE OF X_I.

Symmetry	Test
$S\{x_{n-1} / x'_n\}$	$S^1 + S^3 = 0$
$S\{x_n / x'_{n-1}\}$	$S^2 + S^3 = 0$
$E\{x_n, x_{n-1}\}$	$S^1 + S^2 = 0$
$N\{x_n, x_{n-1}\}$	$S^1 - S^2 = 0$
$S\{x_n / x_{n-1}\}$	$S^2 - S^3 = 0$
$S\{x_{n-1} / x_n\}$	$S^1 - S^3 = 0$

Similarly, for self-anti-dual functions, by definition $f(X) = f(\overline{X})$, which can be rewritten $f(X) = f(X \oplus 2^n - 1)$ and so again, by definition

$$C(2^n - 1) = \sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus 2^n - 1)$$

and thus

$$= \sum_{v=0}^{2^n-1} f(v) \cdot f(v)$$

$$= 2^n.$$

If $C(2^n\text{-}1) = -2^n$ then every pair of minterms $f(v)$ and $f(v \oplus 2^n\text{-}1)$ in the summation $\sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus 2^n - 1)$ must result in a -1 when multiplied and thus must have inverse values of each other. So $f(v) = \overline{f}(v \oplus 2^n - 1)$, or, $f(v) = \overline{f}(\overline{v})$, which is the definition of a self-dual function. Similarly, if $C(2^n\text{-}1) = 2^n$ then every pair of minterms $f(v)$ and $f(v \oplus 2^n\text{-}1)$ in the summation $\sum_{v=0}^{2^n-1} f(v) \cdot f(v \oplus 2^n - 1)$ must result in a 1 when multiplied and thus must have identical values. So $f(v) = f(v \oplus 2^n - 1)$, or, $f(v) = \overline{f}(v)$, which is the definition of a self-anti-dual function. **n**

F. Linear Functions

Definition 5.2 *If a logic function f is represented as $f = a_0 \oplus a_1x_1 \oplus a_2x_2 \oplus \ldots \oplus a_nx_n$ where $a_i=0$ or 1 then f is said to be a linear function* [9].

It is interesting to note that a linear function is either a self-dual or self-anti-dual function. The proof is given in [9]. There are 2^{n+1} linear functions of n variables, and a linear function that is obtained by assigning a linear function to an arbitrary variable of a linear function is also a linear function.

Theorem 16 *A function f(X) is linear if and only if all of its coefficients $C(\tau)=-2^n$, such that the weight of τ $|\tau|=1$.*

This theorem follows directly from Theorem 9. We extend this theorem further to specify that if ALL values of τ with a single one in the binary expansion result in $C(\tau) = -2^n$ then the function must be decomposable for all variables in the fashion described above.

VI. Discussion and Future Work

As noted in Sections V-A and V-B, identifying where a function has equal coefficients within a given order, or in a subset of that order, is not sufficient to identify a symmetric function. However, other work has identified that a function that does not have a symmetry but whose autocorrelation coefficients reflect this property must be in the same autocorrelation class as some totally/partially symmetric function [8]. Thus it may be possible to identify the necessary operations to apply in order to transform the subject function into a symmetric function, thus making it possible to leverage the advantages inherent in symmetries. Future work will address tools to make this determination.

A comment on the suitability of the autocorrelation transform as an analysis tool is appropriate; the authors have found that properties defined on the outputs of a function are better suited to analysis with autocorrelation coefficients than are properties defined based on the structure of a function. For instance, the properties of self-duality and self-anti-duality lend themselves very nicely to identification through autocorrelation coefficients, while on the other hand monotone functions are much more difficult to identify.

VII. Conclusion

There are many existing techniques for the identification of properties such as symmetries, including [15, 16] and [17]. Rather than competing with these, this paper concentrates instead on the theoretical aspects of the autocorrelation transform as an analysis tool. We can conclude from this work that the autocorrelation transform can identify if a function does *not* possess a symmetry, but that the autocorrelation coefficients resulting from the transform do not provide a sufficient condition for the existence of symmetries. Ongoing work in this area includes implementation of our technique in order that we may compare it with existing techniques, as well as the various directions described in Section 6. An extension of the analysis led to necessary and sufficient conditions for the identification of self-dual/self-anti-dual and linear functions. Future work will include implementations for these properties as well.

This paper presents an exploration of the properties inherent to the autocorrelation transform as applied to single-output completely specified boolean functions. Various uses have been suggested in other publications. The ultimate goal of this work is to develop a preprocessing tool which will be used to aid chip designers in making choices prior to or during the design process. For instance, if one can quickly identify that a function cannot result in a non-exponential BDD then much optimization time will be saved by beginning work with a KDD representation. The properties described here are being used in the development of such a tool.

Other avenues for future work include extending this research to the incompletely specified and multiple-output cases.

Appendix -- Proofs

Theorem 6: *A function f(X) has exactly one dissimilar minterm if and only if $C(\tau) = 2^n-4$ $\forall$ $\tau \neq 0$.*

Proof: Without loss of generality let us define a function f such that $f(v)=1$ when $v \in 0,\ldots,2^n-2$ and $f(v)=-1$ when $v=2^n-1$. Then

$$\begin{aligned} C(\tau) &= \sum_{v=0}^{2^n-1} f(v) \times f(v \oplus \tau) \\ &= \Big(\sum_{v=0}^{2^n-2} f(v) \times f(v \oplus \tau)\Big) \\ &\quad + f(2^n-1) \times f(2^n-1 \oplus \tau) \\ &= \Big(\sum_{v=0}^{2^n-2} 1 \times f(v \oplus \tau)\Big) + (-1) \times 1 \\ &= (2^n-2-1)-1 \\ &= 2^n - 4 \forall \tau \neq 0. \end{aligned}$$

Thus if $f(X)$ has exactly one true minterm then all of the coefficients $C(\tau) = 2^n-4$, $\tau \neq 0$.

For the second part of this proof, if all that is known of the function is the coefficients of this pattern, then it can be shown as follows that the function must have either exactly one true or exactly one false minterm.

For a coefficient $C(\tau)$ let us define q as the number of positive pairs in the summation, and r as the number of negative pairs in the summation. A pair in this case is a combination of two minterms i, j, and a positive pair results when both minterms are true or when both are false. It should be noted that in the summation for the autocorrelation equation each pair is encountered twice. Then $2q-2r = 2^n-4$ and $2q+2r = 2^n$

These equations can be solved to show that $r=1$. If there is only one negative pair in the summation then there is only one pair combining a true and a false minterm; all other pairs must combine either two true minterms or two false minterms. If there is only one coefficient $C(\tau)$ for which this

holds, then there can be any number of combinations of true and false minterms to meet these requirements. However, there are 2^n-1 coefficients that have only one negative pair; therefore there can be only one dissimilar minterm in the function. **n**

Theorem 7: *A function has exactly two dissimilar minterms if and only if* $C(0) = 2^n$, $C(\tau_\alpha)=2^n$, *and* $C(\tau) = 2^n$-8 $\forall\ \tau, \tau_\alpha \neq 0$ *and* $\tau \neq \tau_\alpha$.

Proof: We approach this proof by first demonstrating that if there is one coefficient $C(\tau_\alpha)=2^n$, $\tau\neq 0$ and the remaining 2^n-2 coefficients $C(\tau) = 2^n$-8, then the function has exactly two dissimilar minterms. Let us define a function f such that $f(v) = 1$ when $v \in 0,\ldots, 2^n, v \neq i, j$ and $f(v) = 1$ when $v = i, j$. Without loss of generality let i=0 and j=1. Then

$$\begin{aligned} C(\tau) &= \sum_{v=0}^{2^n-1} f(v)\times f(v\oplus\tau) \\ &= f(i)\times f(i\oplus\tau) + f(j)\times f(j\oplus\tau) \\ &\quad + \sum_{v=2}^{2^n-1} f(v)\times f(v\oplus\tau) \\ &= (-1)\times f(0\oplus\tau) + (-1)\times f(1\oplus\tau) \\ &\quad + \sum_{v=2}^{2^n-1} 1\times f(v\oplus\tau) \end{aligned}$$

Then if $i \oplus \tau = j$ and $j \oplus \tau = i$, $C(\tau) = 2^n$. Otherwise $C(\tau) = -2 + (2^n\text{-}4) -2 = 2^n - 8$. Because of the nature of the $\oplus$ operator, $i \oplus \tau = j \Leftrightarrow j \oplus \tau = i$, and so there is only one assignment of τ for which this can occur.

A similar process to that shown in the proof of Theorem 6 can be used to prove that this pattern of coefficients can only result in a function with exactly two dissimilar minterms. **n**

Theorem 8: A function has d dissimilar minterms if and only if the autocorrelation coefficients have the following properties:

- $C(0) = 2^n$,
- for $\binom{d}{p}$ $p \in 2,4,6,\ldots, d$ (or 2,4,6, d-1 if d is odd) $C(\tau) = 2^n$ -4d + 4p, and
- for the remaining coefficients, $C(\tau) = 2^n - 4d$.

The proof is similar to those for Theorems 6 and 7.

Proof: Let us define a function $f(X)$ for which there are d dissimilar minterms. Without loss of generality we assume that $f(v) = -1$ when $v \in 0,\ldots, d$-1 and $f(v) = 1$ when $v \in d,\ldots, 2^n$-1. Then there are d-p mod 2 ways (resulting in $\binom{d}{2}+\binom{d}{4}+\ldots+\binom{d}{d-1}$ or $\binom{d}{2}+\binom{d}{4}+\ldots+\binom{d}{d}$ coefficients) in which pairs of dissimilar minterms may match up, resulting in

$$\begin{aligned} C(\tau) &= 2p - 2(d-p) + \sum_{d}^{2^n-1-d} f(v)\times f(v\oplus\tau) \\ &= 2p - 2(d-p) + 2^n - 2d \\ &= 4p - 4d + 2^n \end{aligned}$$

where the first term $2p$ is the result of the sum of the matching dissimilar minterms, the second term $2(d-p)$ is the result of the sum of the non-matching dissimilar minterms, and the final term is the sum of the remaining minterms which are all similar.

There are also coefficients resulting from the situation in which none of the dissimilar coefficients match in the summation:

$$\begin{aligned} C(\tau) &= -2d + \sum_{d}^{2^n-1-d} f(v)\times f(v\oplus\tau) \\ &= 2^n - 4d. \end{aligned}$$

Again, using a similar technique to that shown in the proof of Theorem 6, if q is the number of positive pairs and r is the number of negative pairs then $2q+2r = 2^n$ and $2q-2r = 2^n$-4d which results in $r = d$. **n**

Theorem 9: $C(\tau_i) = -2^n$ *if and only if the function f(X) has a decomposition* $f(X) = f^*(X) \oplus x_i$ *where* $f^*(X)$ *is independent of* x_i.

Proof: We first determine that a function with the decomposition $f(X) = f^*(X) \oplus x_i$ has a first order autocorrelation coefficient $C(\tau_i) = -2^n$. Without loss of generality let $i = n$. Then

$$\begin{aligned} C(\tau_n) &= \sum_{v=0}^{2^n-1} f(v)\times f(v\oplus\tau_n) \\ &= \sum_{v=0}^{2^n-1} ([f^*(v)\oplus x_n]\times[f^*(v\oplus\tau_n)\oplus(x_n\oplus\tau_n)]) \\ &= \sum_{v=0}^{2^{n-1}-1} (f^*(v)\oplus 0)\times(f^*(v\oplus\tau_n)\oplus(0\oplus\tau_n)) \\ &\quad + \sum_{v=2^{n-1}}^{2^n-1} (f^*(v)\oplus 1)\times(f^*(v\oplus\tau_n)\oplus(1\oplus\tau_n)) \\ &= \sum_{v=0}^{2^{n-1}-1} (f^*(v)\oplus 0)\times(f^*(v\oplus\tau_n)\oplus(1)) \\ &\quad + \sum_{v=2^{n-1}}^{2^n-1} (f^*(v)\oplus 1)\times(f^*(v\oplus\tau_n)\oplus(0)) \\ &= \sum_{v=0}^{2^{n-1}-1} f^*(v)\times(-f^*(v\oplus\tau_n)) \\ &\quad + \sum_{v=2^{n-1}}^{2^n-1} (-f^*(v))\times f^*(v\oplus\tau_n) \\ &= -\sum_{v=0}^{2^n-1} f^*(v)\times f^*(v\oplus\tau_n) \\ &= -2^n. \end{aligned}$$

since by definition $f^*(X)$ is independent of x_n.

We next determine that a first order {+1,-1} autocorrelation coefficient with the value -2^n implies that the function $f(X)$ can be decomposed into $f^*(X) \oplus x_i$. If $C(\tau_i) = -2^n$ then the equation

$$C(\tau_i) = \sum_{v=0}^{2^n-1} f(v)\times f(v\oplus\tau_i)$$

implies that $f(v) = -f(v\oplus\tau_i)\ \forall\ v$. This means that half of the

function is the inverse of the other half, which can be achieved by defining a function $f(X)$ as $f(X) = f^*(X) \oplus x_i$. ■

Theorem 10: *$C(\tau_i) = C(\tau_j) = C(\tau_{ij}) = 0$, $i \neq j$ if and only if the function $f(X)$ can be decomposed into $h(X) \oplus g(X)$ where $g(X) = x_i * x_j$, $* \in \{\wedge, \vee\}$, and $h(X)$ is independent of both x_i and x_j.*

Proof: Let us define a function $f(X) = f^*(X) \oplus g(X)$ where $g(X) = x_i \wedge x_j$ and $f^*(X)$ is independent of x_i and x_j, and let us assume without loss of generality that $i = n$ and $j = n$-1. Then

$$\begin{aligned} C(\tau) &= \sum_{v=0}^{2^n-1} f(v) \times f(v \oplus \tau) \\ &= \sum_A f(v_1) \times f(v_1 \oplus \tau) + \sum_B f(v_2) \times f(v_2 \oplus \tau) \\ &\quad + \sum_C f(v_3) \times f(v_3 \oplus \tau) + \sum_D f(v_4) \times f(v_4 \oplus \tau) \end{aligned}$$

where

- A: $v_1 = 0$ to 2^{n-2}-1 (0000 ... 0011),
- B: $v_2 = 2^{n-2}$ to 2^{n-1}-1 (0100 ... 0111),
- C: $v_3 = 2^{n-1}$ to 2^n-2^{n-2}-1 (1000 ... 1011), and
- D: $v_4 = 2^n$-2^{n-2} to 2^n-1 (1100 ... 1111).[1]

Then

$$\begin{aligned} C(\tau_{n-1}) &= \sum_A [(f^*(v_1) \oplus x_n \wedge x_{n-1}) \\ &\quad \times (f^*(v_1 \oplus \tau_{n-1}) \oplus (x_n \wedge x_{n-1} \oplus \tau_{n-1}))] \\ &\quad + \sum_B [(f^*(v_2) \oplus x_n \wedge x_{n-1}) \\ &\quad \times (f^*(v_2 \oplus \tau_{n-1}) \oplus (x_n \wedge x_{n-1} \oplus \tau_{n-1}))] \\ &\quad + \sum_C [(f^*(v_3) \oplus x_n \wedge x_{n-1}) \\ &\quad \times (f^*(v_3 \oplus \tau_{n-1}) \oplus (x_n \wedge x_{n-1} \oplus \tau_{n-1}))] \\ &\quad + \sum_D [(f^*(v_4) \oplus x_n \wedge x_{n-1}) \\ &\quad \times (f^*(v_4 \oplus \tau_{n-1}) \oplus (x_n \wedge x_{n-1} \oplus \tau_{n-1}))] \\ &= \sum_A [(f^*(v_1) \oplus 0 \wedge 0) \\ &\quad \times (f^*(v_1 \oplus \tau_{n-1}) \oplus (0 \wedge 0 \oplus \tau_{n-1}))] \\ &\quad + \sum_B [(f^*(v_2) \oplus 0 \wedge 1) \\ &\quad \times (f^*(v_2 \oplus \tau_{n-1}) \oplus (0 \wedge 1 \oplus \tau_{n-1}))] \\ &\quad + \sum_C [(f^*(v_3) \oplus 1 \wedge 0) \\ &\quad \times (f^*(v_3 \oplus \tau_{n-1}) \oplus (1 \wedge 0 \oplus \tau_{n-1}))] \\ &\quad + \sum_D [(f^*(v_4) \oplus 1 \wedge 1) \\ &\quad \times (f^*(v_4 \oplus \tau_{n-1}) \oplus (1 \wedge 1 \oplus \tau_{n-1}))] \\ &= \sum_A f^*(v_1) \times f^*(v_1 \oplus \tau_{n-1}) \\ &\quad + \sum_B f^*(v_2) \times f^*(v_2 \oplus \tau_{n-1}) \\ &\quad + \sum_C f^*(v_3) \times (-f^*(v_3 \oplus \tau_{n-1})) \\ &\quad + \sum_D (-f^*(v_4)) \times f^*(v_4 \oplus \tau_{n-1}) \\ &= 2^{n-2} + 2^{n-2} + (-2^{n-2}) + (-2^{n-2}) \\ &= 0 \end{aligned}$$

and similarly for $C(\tau_n)$ and $C(\tau_{n\,n-1})$.

If $C(\tau_n) = C(\tau_{n-1}) = C(\tau_{n\,n-1}) = 0$ then each of the summations may be broken down into $C(\tau) = 2^{n-2} + 2^{n-2} - 2^{n-2} - 2^{n-2}$. Let us assume there exists some variable ordering such that

$$\sum_A f(v_1) \times f(v_1 \oplus \tau_{n-1}) = 2^{n-2} \text{ and}$$
$$\sum_B f(v_2) \times f(v_2 \oplus \tau_{n-1}) = 2^{n-2} \text{ and}$$
$$\sum_C f(v_3) \times f(v_3 \oplus \tau_{n-1}) = -2^{n-2} \text{ and}$$
$$\sum_D f(v_4) \times f(v_4 \oplus \tau_{n-1}) = -2^{n-2}.$$

Then the first two summations tell us that for part of the function $f(v)$ is independent of variable x_{n-1} and the second two indicate that for part of the function $f(v)$ contains $\oplus x_{n-1}$. This indicates that the solution must be of the form $f(X) = f^*(X) \oplus g(X)$ where $f^*(X)$ is independent of x_{n-1} and $g(X)$ contains x_{n-1} . The same process is then applied to the other known coefficients, $C(\tau_{n\,n-1}) = C(\tau_n) = 0$. There are two possible solutions:

Solution 1

	τ_{n-1}	τ_n	$\tau_{n\,n-1}$
$\sum_A f(v_1) \times f(v_1 \oplus \tau)$	$= 2^{n-2}$	$= 2^{n-2}$	$= -2^{n-2}$
$\sum_B f(v_2) \times f(v_2 \oplus \tau)$	$= 2^{n-2}$	$= -2^{n-2}$	$= 2^{n-2}$
$\sum_C f(v_3) \times f(v_3 \oplus \tau)$	$= -2^{n-2}$	$= 2^{n-2}$	$= 2^{n-2}$
$\sum_D f(v_4) \times f(v_4 \oplus \tau)$	$= -2^{n-2}$	$= -2^{n-2}$	$= -2^{n-2}$

The above is obtained for $g(X) = x_1 \wedge x_2$.

Solution 2

	τ_{n-1}	τ_n	$\tau_{n\,n-1}$
$\sum_A f(v_1) \times f(v_1 \oplus \tau)$	$= -2^{n-2}$	$= -2^{n-2}$	$= -2^{n-2}$
$\sum_B f(v_2) \times f(v_2 \oplus \tau)$	$= -2^{n-2}$	$= 2^{n-2}$	$= 2^{n-2}$
$\sum_C f(v_3) \times f(v_3 \oplus \tau)$	$= 2^{n-2}$	$= -2^{n-2}$	$= 2^{n-2}$
$\sum_D f(v_4) \times f(v_4 \oplus \tau)$	$= 2^{n-2}$	$= 2^{n-2}$	$= 2^{n-2}$

The above is obtained for $g(X) = x_1 \vee x_2$. ■

The proof is easily extended to any number of variables in

[1] Four variable expansions are given for the sake of clarity only. This does not limit the proof to four variables.

g since there is always only one input combination for which $g(X) = x_i \wedge x_{i+1} \wedge \ldots \wedge x_{i+m} = 1$ and similarly where $g(X) = x_i \vee x_{i+1} \vee \ldots \vee x_{i+m} = 0$.

Theorem 11: *If f(X) can be decomposed into h(X1)⊕g(X2) where X1 ∪ X2=X and X1 ∩ X2=∅ then* $C^{ff}(\tau_{X2}) = C^{gg}(\tau_{X2})$.

Proof: Without loss of generality, let us assume that *X2* consists of some combination of the variables $x_1, x_2, \ldots, x_{m-1}$ and *X1* consists of the remaining variables from X. Then given a function *f(X)* that is decomposable into *h(X1)* ⊕ *g(X2)* as described in the theorem above, by definition we have $h(X) = h(X \oplus \tau_{X2})$; in other words, *h* is not affected if any of the variables in *X*2 are changed. Then, noting that in {+1,-1} notation performing an ⊕ operation is the same as the mathematical multiplication operation, the following holds:

$$
\begin{aligned}
C^{ff}(\tau_{X2}) &= \sum_{v=0}^{2^n-1} f(v) \times f(v \oplus \tau_{X2}) \\
&= \sum_{v=0}^{2^n-1} [h(v) \oplus g(v)] \times [h(v \oplus \tau_{X2}) \oplus g(v \oplus \tau_{X2})] \\
&= \sum_{v=0}^{2^n-1} h(v) \times h(v \oplus \tau_{X2}) \times g(v) \times g(v \oplus \tau_{X2}) \\
&= \sum_{v=0}^{2^n-1} (h(v))^2 \times g(v) \times g(v \oplus \tau_{X2}) \\
&= \sum_{v=0}^{2^n-1} 1 \times g(v) \times g(v \oplus \tau_{X2}) \\
&= C^{gg}(\tau_{X2})
\end{aligned}
$$

n

REFERENCES

[1] R. Tomczuk, Autocorrelation and decomposition methods in combinational logic design, Ph.D. thesis, University of Victoria (1996).

[2] J. E. Rice, J. C. Muzio, M. Serra, The use of autocorrelation coefficients for variable ordering for ROBDDs, in: Proceedings of the 4th International Workshop on Applications of the Reed-Müller Expansion in Circuit Design, 1999, pp. 185--196.

[3] M. Karpovsky, Finite Orthogonal Series in the Design of Digital Devices, John Wiley & Sons, 1976.

[4] J. E. Rice, J. C. Muzio, Methods for calculating autocorrelation coefficients, in: Proceedings of the 4th International Workshop on Boolean Problems, (IWSBP2000), 2000, pp. 69--76.

[5] J. E. Rice, On the use of autocorrelation coefficients in the identification of three-level decompositions, in: Proceedings of the International Workshop on Logic Synthesis (IWLS), 2003, pp. 187--191.

[6] R. S. Stankovic, M. G. Karpovsky, Remarks on calculation of autocorrelation on finite dyadic groups by local transformations of decision diagrams, in: R. Moreno-Daz, F. Pichler, A. Quesada-Arencibia (Eds.), EUROCAST, Vol. 3643 of Lecture Notes in Computer Science, Springer, 2005, pp. 301--310.

[7] S. L. Hurst, The Logical Processing of Digital Signals, Crane Russak & Company, Inc., 1978.

[8] J. E. Rice, J. C. Muzio, Use of the autocorrelation function in the classification of switching functions, in: Proceedings of the Euromicro Symposium on Digital System Design: Architectures, Methods and Tools (DSD), 2002, pp. 244--251.

[9] T. Sasao, Switching Theory for Logic Synthesis, Kluwer Academic Publishers, 1999.

[10] Y. Kolotov, I. Levin, V. Ostrovsky, M. G. Karpovsky, Software tool for BDD optimizing by using autocorrelation functions, in: Proceedings of the 23rd IEEE Convention of EEEI, 2004, pp. 129--132.

[11] [11] J. E. Rice, Making a choice between FDDs and BDDs, in: Proceedings of the International Workshop on Logic Synthesis (IWLS), 2005, pp. 46--50.

[12] J. E. Rice, J. C. Muzio, Antisymmetries in the realization of boolean functions, in: Proceedings of the International Symposium on Circuits and Systems (ISCAS), 2002, CD ROM paper number 2666.

[13] S. L. Hurst, D. M. Miller, J. C. Muzio, Spectral Techniques in Digital Logic, Academic Press, Inc., Orlando, Florida, 1985.

[14] J. E. Rice, Autocorrelation coefficients in the representation and classification of switching functions, Ph.D. thesis, University of Victoria (2003).

[15] D. Möller, J. Mohnke, M. Weber, Detection of symmetry of Boolean functions represented by ROBDDs, in: Proceedings of the International Conference on Computer-Aided Design (ICCAD), 1993, pp. 680--684.

[16] S. Kannurao, B. J. Falkowski, Identification of complement single variable symmetry in Boolean functions through Walsh transform, in: Proceedings of the International Symposium on Circuits and Systems (ISCAS), 2002, pp. 745--748.

[17] S. Panda, F. Somenzi, B. Plessier, Symmetry detection and dynamic variable ordering of decision diagrams, in: Proceedings of International Conference on Computer-Aided Design (ICCAD), 1994, pp. 628--631.