設 計 自 動 化 55－7
(1990. 12. 12)

ＡＮＤ－ＥＸＯＲ論理式について

－－その性質と最小化アルゴリズム－－

笹尾 勤

＊ 九州工業大学情報工学部 電子情報工学科

あらまし　任意の論理関数は論理積の排他的論理和(ESOP)で表現できる．ＥＳＯＰはＲＭＥ(Reed-Muller論理式) を一般化したものであり，積項数は少なくてよいが，能率のよい最小化方法は知られていない．本稿では，最小ＥＳＯＰ（ＭＥＳＯＰ）の性質を示し，その積項数の上界および下界を導く．また，いくつかの関数のクラスの積項数を調べる．次に簡単化アルゴリズムを与える．本アルゴリズムは４変数関数のＭＥＳＯＰの表を利用しており，計算機実験を行った結果，５変数乱数論理関数の約３０％に対して簡単化したＥＳＯＰの最小性を証明できた．

Exclusive-or Sum-Of-Products Expressions:

--- Their Properties and Minimization Algorithm ---

Tsutomu SASAO

Department of Computer Science and Electronics
Kyushu Institute of Technology, Iizuka 820, Japan

Abstract This paper presents properties of Exclusive-OR sum-of-products expressions (ESOPs) and their minimization algorithm. First, upper and lower bounds on the number of products in minimum ESOPs (MESOPs) are shown. Then an algorithm to simplify ESOPs is presented. The algorithm can often prove their minimality for functions of up to five variables. It utilizes the minimized results of all the 4-variable functions, and 1) finds lower bounds on the number of products in the MESOP, 2) obtains an initial solution for 5-variable functions, 3) simplifies the ESOP by iterative improvement, and 4) stops the iterative improvement when the solution is proved to be minimum.
This algorithm proved the minimality of about 30% of randomly generated functions with 5 variables. We also show the number of the products in MESOPs for some classes of functions.

I. Introduction

Recently, logic synthesis tools are used to design multi-level logic networks. Such tools often produce better solutions in a shorter time than human logic designers, and are now indispensable in the practical VLSI design. It should be noted that most tools utilize sum-of-products expression (SOP) minimizer extensively in the programs. Similarly, in order to design multi-level logic networks consisting of EXOR gates as well as AND and OR gates, a good exclusive-OR sum-of-products expression (ESOP) minimizer is necessary. As for exact minimization algorithm, only exhaustive methods are known [BIO 73, KOD 89]. As for near minimum ESOPs, several heuristic algorithms have been developed[BES 83, EVE 67, FLE 87, HEL 88, PAP 79, ROB 82, SAL 79]. Experimental results show that ESOPs require fewer products than SOPs in many cases [SAS 90a, SAS90b].

In this paper, we show methods to find upper and lower bounds on the number of products in MESOPs for the functions of n variables by using the minimum ESOPs for the functions of (n-1) variables. We also show an algorithm which simplifies ESOPs and prove their minimality for a considerable percentage of five-variable functions. Because no exact minimization algorithm is known except for an exhaustive method, this algorithm is the first one, although it can minimize up to five-variable functions. It also finds a near optimal solution for functions with more input variables, which are more important in practical designs.

II. Definitions and Basic Properties

Definition 2.1: x and $\bar{x}$ are literals of a variable x. A logical product which contains at most one literal for each variable is called a product term. Product terms combined with OR operators form a Sum-of-Products expression (SOP). Product terms combined with EXOR operators form an Exclusive-Or Sum-of-Products expression (ESOP).

Definition 2.2: A minterm is a logical product containing a literal for each variable. A minterm implying a function f is called a minterm of f.

Definition 2.3: A SOP for f is said to be a minimum SOP (or MSOP) for f if the number of the products is the minimum. An ESOP for f is said to be a minimum ESOP (or MESOP) for f if its number of products is the minimum.

Note that ESOPs are generalization of

1) Fixed polarity Reed-Muller Expressions (RMEs) (2^n ways),
2) Kronecker expressions :3^n ways),
3) Pseud-Kronecker expressions: $3^{(2^{n+1}-1)}$ ways), and
4) Generalized RMEs ($2^{n2^{n-1}}$ ways).

As for the definitions of the above expressions, read [DAV 78].

Definition 2.4: The number of products in a SOP F is denoted by $t(F)$. The number of products in an MSOP for f is denoted by $t(f)$. The number of products in an ESOP F is denoted by $\tau(F)$. The number of products in an MESOP for f is denoted by $\tau(f)$.

Lemma 2.1: If a function f can be represented as $f=g\oplus h$, then $\tau(f)\leqq\tau(g)+\tau(h)$.

(Proof) Let G and H be MESOPs for functions g and h, respectively. Because $G\oplus H$ represents the function f, we need at most $\tau(G)+\tau(H)$ products to realize f.

Lemma 2.2: $|\tau(f)-\tau(\bar{f})|\leqq 1$.

(Proof) Suppose that $g=\bar{f}$. Let MESOPs for f and g be F and G, respectively. Because $g=f\oplus 1$ and $f=g\oplus 1$, two ESOPs $F\oplus 1$ and $G\oplus 1$ represent g and f, respectively. Therefore, we have $\tau(g)\leqq\tau(F\oplus 1)\leqq\tau(f)+1$ and $\tau(f)\leqq\tau(G\oplus 1)\leqq\tau(g)+1$. From these, we have $\tau(g)-\tau(f)\leqq 1$ and $\tau(f)-\tau(g)\leqq 1$. Hence the lemma. (Q.E.D.)

Lemma 2.3: Suppose that $g=\bar{f}$ and $\tau(f)<\tau(g)$. If F is an MESOP for f, then $F\oplus 1$ is a MESOP for g.

(Proof) Let a MESOP of g be G. Because $g=f\oplus 1$ and by the condition of this lemma and by Lemma 2.2, we have $\tau(G)=\tau(F)+1$. This shows that $F\oplus 1$ is a MESOP for g. (Q.E.D.)

Lemma 2.4: Suppose that two functions f and g are represented as

$f=\bar{x}\cdot f0\oplus x\cdot f1\oplus 1\cdot f2$, and ------①

$g=\bar{x}\cdot g0\oplus x\cdot g1\oplus 1\cdot g2$. ------②

Then, $f=g \rightleftarrows f0\oplus g0=f1\oplus g1=f2\oplus g2$.

(Proof)

(→) Suppose that f=g. By setting x=0 to① and ②, we have

$f0\oplus f2=g0\oplus g2$. -------③

By setting x=1 to① and ②, we have

$f1\oplus f2=g1\oplus g2$. -------④

By ③ and ④, we have

$f0\oplus g0=f1\oplus g1=f2\oplus g2$.

(←) Suppose that $f0\oplus g0=f1\oplus g1=f2\oplus g2=h$. Then, $f\oplus g=\bar{x}(f0\oplus g0)\oplus x(f1\oplus g1)\oplus 1\cdot(f2\oplus g2)=h(\bar{x}\oplus x\oplus 1)=0$. Therefore, f=g. (Q.E.D.)

Lemma 2.5a: Let $\bar{x}\cdot H0\oplus x\cdot H1\oplus 1\cdot H2$ be an ESOP representing a function f. Let the ESOP $x\cdot H0\oplus\bar{x}\cdot H1\oplus 1\cdot H2$, which is obtained by interchanging the literals $\bar{x}$ and x of the original ESOP, represent a function g. Then $\tau(f)=\tau(g)$.

(Proof) Let MESOPs for f and g be $F=\bar{x}\cdot F0\oplus x\cdot F1\oplus 1\cdot F2$, and $G=\bar{x}\cdot G0\oplus x\cdot G1\oplus 1\cdot G2$, respectively.

Let the ESOPs which are obtained by interchanging the literals $\bar{x}$ and x in F and G be $FP=x\cdot F0\oplus\bar{x}\cdot F1\oplus 1\cdot F2$, and $GP=x\cdot G0\oplus\bar{x}\cdot G1\oplus 1\cdot G2$, respectively. Then, we can show that FP represents a function g.

Note that $f=\bar{x}\cdot H0\oplus x\cdot H1\oplus 1\cdot H2=\bar{x}\cdot F0\oplus x\cdot F1\oplus 1\cdot F2$.

By Lemma 2.4, $H0\oplus F0=H1\oplus F1=H2\oplus F2$. Therefore,

$g\oplus FP=(x\cdot H0\oplus\bar{x}\cdot H1\oplus 1\cdot H2)\oplus(x\cdot F0\oplus\bar{x}\cdot F1\oplus 1\cdot F2)$
$=x\cdot(H0\oplus F0)\oplus\bar{x}\cdot(H1\oplus F1)\oplus 1\cdot(H2\oplus F2)=0$

In other words, FP represents g. Similarly, GP represents f. Because F and G are MESOPs for f and g, respectively, we have $\tau(F)\leqq\tau(GP)$ and $\tau(G)\leqq\tau(FP)$. Also, it is clear that $\tau(F)=\tau(FP)$ and $\tau(G)=\tau(GP)$. Hence $\tau(F)=\tau(G)$ and $\tau(f)=\tau(g)$. (Q.E.D.)

Lemma 2.5b: Let $\bar{x}\cdot H0\oplus x\cdot H1\oplus 1\cdot H2$ be an ESOP representing a function f. Let the ESOP $1\cdot H0\oplus x\cdot H1\oplus \bar{x}\cdot H2$, which is obtained by interchanging the literals $\bar{x}$ and 1 of the original ESOP, represent a function g. Then $\tau(f)=\tau(g)$.

(Proof) Let MESOPs for f and g be $F=\bar{x}\cdot F0\oplus x\cdot F1\oplus 1\cdot F2$, and $G=\bar{x}\cdot G0\oplus x\cdot G1\oplus 1\cdot G2$, respectively.
Let the ESOPs which are obtained by interchanging the literals $\bar{x}$ and 1 in F and G be $FP=1\cdot F0\oplus x\cdot F1\oplus \bar{x}\cdot F2$, and $GP=1\cdot G0\oplus x\cdot G1\oplus \bar{x}\cdot G2$, respectively. Then, we can show that FP represents a function g.
Note that $f=\bar{x}\cdot H0\oplus x\cdot H1\oplus 1\cdot H2=\bar{x}\cdot F0\oplus x\cdot F1\oplus 1\cdot F2$.
By Lemma 2.4, $H0\oplus F0=H1\oplus F1=H2\oplus F2$. Therefore,

$$g\oplus FP=(1\cdot H0\oplus x\cdot H1\oplus \bar{x}\cdot H2)\oplus(1\cdot F0\oplus x\cdot F1\oplus \bar{x}\cdot F2)$$
$$=1\cdot(H0\oplus F0)\oplus x\cdot(H1\oplus F1)\oplus \bar{x}\cdot(H2\oplus F2)=0$$

In other words, FP represents g. Similarly, GP represents f. Because F and G are MESOPs for f and g, respectively, we have $\tau(F)\leqq\tau(GP)$ and $\tau(G)\leqq\tau(FP)$. Also, it is clear that $\tau(F)=\tau(FP)$ and $\tau(G)=\tau(GP)$. Hence $\tau(F)=\tau(G)$ and $\tau(f)=\tau(g)$. (Q.E.D.)

Lemma 2.5c: Let $\bar{x}\cdot H0\oplus x\cdot H1\oplus 1\cdot H2$ be an ESOP representing a function f. Let the ESOP $\bar{x}\cdot H0\oplus 1\cdot H1\oplus x\cdot H2$, which is obtained by interchanging the literals x and 1 in the original ESOP, represents a function g. Then $\tau(f)=\tau(g)$.

(Proof) Similar to Lemma 2.5a. (Q.E.D.)

Theorem 2.1: Let an ESOP for a function f be

$$\sum_{\oplus (S_1, S_2, \cdots, S_n)} x_1^{S_1} x_2^{S_2} \cdots x_n^{S_n}$$

Consider an ESOP which is obtained by interchanging some of literals $(1, x_i, \bar{x}_i)$, and let g be the function represented by the ESOP. Then $\tau(f)=\tau(g)$.

(Proof) By using Lemmas 2.3 to 2.5, iteratively. (Q.E.D.)

Theorem 2.2: Let a function f be represented as $f=x\cdot g$ or $f=\bar{x}\cdot g$, where g is a function independent of the variable x, then $\tau(f)=\tau(g)$.

(Proof) When f is represented as $f=x\cdot g$:
Replace x with 1, and apply Theorem 2.1.
When f is represented as $f=\bar{x}\cdot g$:
Replace $\bar{x}$ with 1, and apply Theorem 2.1. (Q.E.D.)

Theorem 2.3: Let a function f be represented as $f=xp\oplus g$, $f=\bar{x}p\oplus g$, $f=p\oplus xg$, or $f=p\oplus \bar{x}g$, where p is a product term, g is a function and, both p and g do not depend on the variable x.
Then $\tau(f)=\min\{\tau(g), \tau(p\oplus g)\}+1$.

(Proof) When f is represented as $f=xp\oplus g$:
Note that f can be represented as $f=xp\oplus g=\bar{x}p\oplus(p\oplus g)$.
From Lemma 2.1, we have $\tau(f)\leqq 1+\tau(g)$ and $\tau(f)\leqq 1+\tau(p\oplus g)$. Therefore,

$\tau(f)\leqq\min\{\tau(g), \tau(p\oplus g)\}+1$ ----①

Next, let F be an MESOP for f. Because f depends on the variable x, F contains at least one product containing the literal x or $\bar{x}$.

A) When the product term contains the literal x.
In F, if we set x to 0, then that product term becomes 0, and it can be deleted from F. Because the resulting ESOP represents the function g,
we have $\tau(g)\leqq\tau(F)-1$ ----②.

B) When the product term contains the literal $\bar{x}$.
In F, if we set $\bar{x}$ to 0, then that product term becomes 0, and it can be deleted from F. Because the resulting ESOP represents the function $p\oplus g$,
we have $(p\oplus g)\leqq\tau(F)-1$ ----③
From ② and ③, we have
$\tau(F)\geqq\min\{\tau(g), \tau(p\oplus g)\}+1$ ----④
From ① and ④, we have $\tau(f)=\min\{\tau(g), \tau(p\oplus g)\}+1$.
For other cases: We can show the same result by Theorem 2.1 and the above result. (Q.E.D.)

Theorem 2.4: $\tau(f)\leqq\min\{|f|, 2^n-|f|+1\}$, where $|f|$ denotes the number of minterms in f.

(Proof) Let the minterm expansion of the function f be $f=m_1 \vee m_2 \vee\cdots\vee m_k$. Because $m_i\cdot m_j=0(i\neq j)$, f is also represented as $f=m_1\oplus m_2\oplus\cdots\oplus m_k$. Therefore, we have $\tau(f)\leqq|f|$. Since $f=\bar{f}\oplus 1$, we have $\tau(f)\leqq\tau(\bar{f})+1$ by Lemma 2.1. Also note that $\tau(\bar{f})\leqq|\bar{f}|=2^n-|f|$. From these we have $\tau(f)\leqq 2^n-|f|+1$. (Q.E.D.)

Lemma 2.6: If a function f can be represented as an ESOP without a minterm, then f contains an even number of minterms.

(Proof) Suppose that the function f is represented by an ESOP without a minterm. Now consider the map for f and loops created by the ESOP. The number of cells covered by a loop of the ESOP is either 2, 4, 8, …, or 2^k. Note that these numbers are all even. So, the total number of cells covered by the loops of the ESOP is even, if we count the cells with repetition.
Let n_i be the number of cells which the loops cover exactly i times. Where $i=1,2,\cdots,k$. Assume that k is an odd number, and n_k may be 0. When we count the number of cells with repetition, total number of cells covered by the loops of the map is

$$A=n_1+2n_2+3n_3+\cdots+kn_k,$$
$$=(n_1+n_3+n_5+\cdots+n_k)+(2n_2+2n_3+4n_4+4n_5+\cdots+(k-1)n_{k-1})$$

Because k is an odd number, the sum of the latter part is even number.
Because A is even number, $n_1+n_3+n_5+\cdots+n_k$ is also even.
In the ESOP, the minterms of f are cells covered by odd number of loops. So the total number of minterms in f is $n_1+n_3+n_5+\cdots+n_k$.
Hence, the number of minterms of f is even. (Q.E.D.)

Theorem 2.5: If the number of the minterms of f is odd, then any ESOP for f contains a minterm.

(Proof) Suppose that an ESOP for f does not contain a minterm. By Lemma 2.6, the number of minterms of f must be even number. This contradicts the assumption of the theorem. Hence the theorem. (Q.E.D.)

Example 2.1: Consider the 2-variable function $f=x_1\vee x_2$.
Note that f has three minterms. ESOPs for this function are:
$x_1\oplus x_2\oplus x_1x_2$, $x_1\oplus\bar{x}_1x_2$, $x_2\oplus x_1\bar{x}_2$, and $1\oplus\bar{x}_1\bar{x}_2$.
Note that each ESOP contains a minterm. However, the last ESOP contains a minterm which is not a minterm of f. (End of Example)

Theorem 2.6: If the number of minterms of f is odd, then

$\tau(f)=\min_{\underline{a}\in B^n}\{\tau(f\oplus m)\}+1$, where $m=x_1^{a_1}\cdot x_2^{a_2}\cdot\dots\cdot x_n^{a_n}$ and $\underline{a}=(a_1,a_2,\dots,a_n)$.

(Proof) By Theorem 2.5, an MESOP for f contains a minterm. Let it be m_0. The function other than the minterm can be represented by $f\oplus m_0$. Hence the theorem. (Q.E.D.)

III. Upper and Lower Bounds on the Number of Products

Theorem 3.1: $\tau(f)\geqq A$,
where $A=\max_i[\max\{\tau(fi0),\tau(fi1),\tau(fi2)\}]$,
$f=\bar{x}ifi0\oplus xifi1$, $fi2=fi0\oplus fi1$, and $i=1,2,\dots,n$.

(Proof) Let an MESOP of f be
$Fm=\bar{x}iFa\oplus xiFb\oplus Fc$ ------①
From ①, we have
$\tau(f)=\tau(Fa)+\tau(Fb)+\tau(Fc)$ ------②
By setting xi to 0 in ①, we have
$\tau(fi0)\leqq\tau(Fa)+\tau(Fc)\leqq\tau(f)$ -----③
because $Fm(xi=0)=Fa\oplus Fc$ represents the function fi0.
Similarly, by setting xi to 1 in ①, we have
$\tau(fi1)\leqq\tau(Fb)+\tau(Fc)\leqq\tau(f)$ -----④
because $Fm(xi=1)=Fb\oplus Fc$ represents the function fi1.
Next, consider the ESOP which is obtained by interchanging the literals xi and 1:
$Gm=\bar{x}iFa\oplus 1\cdot Fb\oplus xiFc$. ------⑤
Let the function represented by Gm be g. Then by Theorem 2.1, Gm is a MESOP for g. By setting xi to 0 in ⑤, we have
$\tau(fi2)\leqq\tau(Fa)+\tau(Fb)\leqq\tau(g)$ -----⑥
because $Gm(xi=0)=Fa\oplus Fb$ represents the function fi2.
From ③〜⑥, and $\tau(f)=\tau(g)$, we have
$\max\{\tau(fi0),\tau(fi1),\tau(fi2)\}\leqq\tau(f)$
This relation holds for all possible i, so we have the theorem. (Q.E.D.)

Theorem 3.2: $\tau(f)\leqq B$, where $B=\min[B_i]$, where $B_i=\tau(fi0)+\tau(fi1)+\tau(fi2)-\max_i\{\tau(fi0),\tau(fi1),\tau(fi2)\}$,
$f=\bar{x}ifi0\oplus xifi1$, $fi2=fi0\oplus fi1$, and $i=1,2,\dots,n$.

(Proof) Because f can be represented as
$f=fi0\oplus xifi2=fi1\oplus\bar{x}ifi2$
$=\bar{x}ifi0\oplus xifi1$, we have
$\tau(f)\leqq\tau(fi0)+\tau(fi2)$, $\tau(f)\leqq\tau(fi1)+\tau(fi2)$,
and $\tau(f)\leqq\tau(fi0)+\tau(fi1)$. Therefore, $\tau(f)\leqq B_i$.
Because these relations hold for all possible i, we have the theorem. (Q.E.D.)

Lemma 3.1: Suppose that f is represented as $f=\bar{x}f0\oplus xf1$. If $\tau(f)=\tau(f1)$, then an MESOP for f has a form $xFa\oplus Fb$, where Fa and Fb are ESOPs not containing the variable x.

(Proof) Suppose that an MESOP for f is represented in the form:
$xFa\oplus Fb\oplus\bar{x}Fc$, -----①
where $Fc\neq 0$. We have
$\tau(f)=\tau(Fa)+\tau(Fb)+\tau(Fc)$. -----②
Because $\tau(Fc)>0$, from ② we have
$\tau(f)>\tau(Fa)+\tau(Fb)$. -----③
Let x be 1. Because ① represent f1, we have
$\tau(f1)\leqq\tau(Fa)+\tau(Fb)$. -----④
From ③ and ④ we have $\tau(f1)<\tau(f)$. However, this contradicts the assumption of the lemma. In other words, if we assume that the MESOP has the form ①, then we have the contradiction. (Q.E.D.)

Lemma 3.2: For a given function f, let A and B be the values defined in Theorems 3.1 and 3.2.
Then $\tau(f)=A \rightleftarrows A=B$.

(Proof for →) When $\tau(f)=A$, without loss of generality, we can assume that $\tau(fi1)=A$.
By Lemma 3.1, an MESOP for f has the form
$xiFa\oplus Fb$ ---①.
Because f can be represented as $f=xifi2\oplus fi0$,
Fa represents a function fi2, and Fb represents a function fi1. Also note that both Fa and Fb are MESOPs for fi2 and fi0, respectively. Because if not, ① is not a MESOP. Therefore,
$\tau(Fa)=\tau(fi2)$ and $\tau(Fb)=\tau(fi0)$.
So, we have $\tau(f)=\tau(Fa)+\tau(Fb)=\tau(fi2)+\tau(fi0)$.
By the definition of B, we have $\tau(fi2)+\tau(fi0)\geqq B$.
Therefore, $\tau(f)\geqq B$. On the other hand, by Theorem 3.2 we have $\tau(f)\leqq B$. So $\tau(f)=B$. Hence A=B.

(Proof for ←) When A=B. By Theorem 3.2 we have $\tau(f)\geqq A$. By Theorem 3.2, we have $\tau(f)\leqq B$.
Hence $\tau(f)=A$. (Q.E.D.)

Theorem 3.3: For a given function f, let A and B be values defined by Theorem 3.1 and 3.2. If $A\neq B$ then $\tau(f)\geqq A+1$.

(Proof) By Lemma 3.2, if $A\neq B$ then $\tau(f)\neq A$. By Theorem 3.1, we have $A\leqq\tau(f)$. Therefore $A<\tau(f)$.
Hence the theorem. (Q.E.D.)

Example 3.1: Suppose that the table of MESOPs for 3-variable functions is available. Let's prove that the following ESOP is minimum:
$f=\bar{x}1\bar{x}4\oplus\bar{x}1x2x3\oplus x1x2\bar{x}3$.
f can be represented as
$f=\bar{x}4(\bar{x}1\oplus\bar{x}1x2x3\oplus x1x2\bar{x}3)\oplus x4(\bar{x}1x2x3\oplus x1x2\bar{x}3)$
$=\bar{x}4f0\oplus x4f1$.
Note that $f2=f1\oplus f0=\bar{x}1$.
From the table of MESOPs, we have $\tau(f0)=3$, $\tau(f1)=2$ and $\tau(f2)=1$. So, we have $\max\{\tau(f0),\tau(f1),\tau(f2)\}=3$.
By Theorem 3.1, we have $\tau(f)\geqq 3$. Therefore, the given ESOP is minimum. (End of Example)

Example 3.2: Suppose that the Table of MESOPs for 3-variable functions is available. Let prove that the following ESOP is minimum:
$f=\bar{x}1\bar{x}3\bar{x}4\oplus x2x3\bar{x}4\oplus x1x3x4\oplus x1\bar{x}2\bar{x}3$.
This function can be represented as
$f=\bar{x}4(\bar{x}1\bar{x}3\oplus x2x3\oplus x1\bar{x}2\bar{x}3)\oplus x4(x1x3\oplus x1\bar{x}2\bar{x}3)$
$=\bar{x}4f0\oplus x4f1$
Note that $f2=f1\oplus f0=\bar{x}1\bar{x}3\oplus x1x3\oplus x2x3$
By the table of MESOPs, we have $\tau(f0)=3$, $\tau(f1)=2$, and $\tau(f2)=3$. In a similar way we have
$A=\max_i[\max\{\tau(fi0),\tau(fi1),\tau(fi2)\}]=3$ and
$B=\tau(fi0)+\tau(fi1)+\tau(fi2)-\max_i\{\tau(fi0),\tau(fi1),\tau(fi2)\}$
$=5$.

By Theorem 3.3, we have $\tau(f) \geqq 4$ because $A \neq B$.
Hence, the given ESOP is minimum. (End of Example)

Example 3.3: Suppose that the table of MESOPs for all the 4-variable functions is available. Let prove that the following ESOP is minimum:

$f=\bar{x}1\bar{x}2\bar{x}3x4\bar{x}5 \oplus \bar{x}1x2x3\bar{x}4\bar{x}5 \oplus x1\bar{x}2\bar{x}3\bar{x}4x5 \oplus x1x2x3x4x5.$

f can be expanded as

$f=\bar{x}1(\bar{x}2\bar{x}3x4\bar{x}5 \oplus x2x3\bar{x}4\bar{x}5) \oplus x1(\bar{x}2\bar{x}3\bar{x}4x5 \oplus x2x3x4x5)$
$=\bar{x}1f0 \oplus x1f1 .$

Also note that

$f2=f1 \oplus f0=\bar{x}2\bar{x}3x4\bar{x}5 \oplus x2x3\bar{x}4\bar{x}5 \oplus \bar{x}2\bar{x}3\bar{x}4x5 \oplus x2x3x4x5$.

From the table of MESOPs, we have $\tau(f0)=\tau(f1)=2$, and $\tau(f2)=4$. Note that $Max\{\tau(f0), \tau(f1), \tau(f2)\}=4$.
By Theorem 3.3, we have, $\tau(f) \geqq 4$. This shows that the given ESOP is minimum. (End of Example)

IV. Complexity of Some Classes of Functions

Experimental results show that ESOPs require fewer products than SOPs to represent symmetric functions and randomly generated functions [SAS 90a]. Also, an ESOP requires only n products to represent a parity function of n variables while the SOP requires 2^{n-1} products. However, this is not always the case. There is a 2n variable function whose MESOP requires $2^{n-1}-1$ products while the MSOP requires n products.

Conjecture 4.1:
$\tau(f_n)=2^n$, where $f_n=(x_1 \oplus y_1)\cdot(x_2 \oplus y_2)\cdots\cdots(x_n \oplus y_n)$.

Lemma 4.1: $\tau(f_n)=\tau(g_n)$,
where $g_n=(1 \oplus x_1y_1)\cdot(1 \oplus x_2y_2)\cdots\cdot(1 \oplus x_ny_n)\cdot$

(Proof) Replacing the literals as $x_i \rightleftarrows 1$ in g_n, and we have the function f_n. By Theorem 2.1, $\tau(f_n)=\tau(g_n)$.
(Q.E.D.)

Lemma 4.2: $h_n=g_n \oplus 1$, where $h_n= x_1y_1 \vee x_2y_2 \vee \cdots \vee x_ny_n$.

(Proof) $g_n = (1 \oplus x_1y_1)\cdot(1 \oplus x_2y_2)\cdots\cdot(1 \oplus x_ny_n)$
$= \overline{(x_1y_1)}\cdot \overline{(x_2y_2)} \cdots\cdots \overline{(x_ny_n)}$.

$g_n \oplus 1 = \bar{g}_n = x_1y_1 \vee x_2y_2 \vee \cdots \vee x_ny_n = h_n$. (Q.E.D.)

Lemma 4.3: $\tau(h_n) \leqq 2^n-1$.

(Proof) Let H_n be the expression defined as follows:
$H_n= H_{n-1} \oplus x_ny_n(H_{n-1} \oplus 1)$, where $H_1= x_1y_1$.
By applying distributive law repeatedly, we have the ESOP with for H_n with $2^n -1$ products. Also note that

$1 \oplus H_n = H_{n-1} \oplus x_ny_n \cdot H_{n-1} \oplus x_ny_n \oplus 1$
$= (1 \oplus H_{n-1})\cdot(1 \oplus x_ny_n)$.

By replacing $1 \oplus H_{n-1}$ recursively, we have
$1 \oplus H_n =(1 \oplus x_1y_1)\cdot(1 \oplus x_2y_2)\cdots\cdots(1 \oplus x_ny_n)$.
Therefore, $1 \oplus H_n$ represents the function g_n.
By Lemma 4.2, H_n represents the function h_n.
(Q.E.D.)

Conjecture 4.2: $\tau(h_n) \geqq 2^n-1$.

(Explanation supporting the conjecture)
By Conjecture 4.1, Lemmas 4.1 and 4.2, we have $\tau(\bar{h}_n)=2^n$. By Lemma 2.3, $|\tau(h_n)-\tau(\bar{h}_n)| \leqq 1$.
Hence, $\tau(h_n) \geqq 2^n-1$. (End of explanation)

Note that an ESOP for h_n requires 2^n-1 products, while SOP's require only n products.

Definition 4.1: E(n,k) function is defined by an ESOP which consists of product terms with exactly k true variables out of n inputs:

$$E(n,k)= \sum_{\oplus\; i<j<\cdots<m} x_ix_j\cdots x_m$$.

Note that the above expression has at most $\binom{n}{k}$ products. Because MESOPs for E(n,k) are not so easy to obtain, they are used as benchmark functions for ESOP minimization algorithms [SAS 90a].

Theorem 4.1: $T(n,k)=T(n,n-k)$, where $T(n,k)=\tau(E(n,k))$.

(Proof) Consider the ESOP

$$E(n,k)= \sum_{\oplus\; i<j<\cdots<m} x_ix_j\cdots x_m$$

Apply the transformations $x_q \rightleftarrows 1$ $(q=1,2,\cdots,n)$, and we have the ESOP

$$\Sigma_{\oplus}\, x_1x_2 \cdots x_{i-1}1\, x_{i+1}\cdots x_{j-1}1\, x_{j+1}\cdots x_{m-1}1\, x_{m+1}\cdots x_n.$$

It is easy to see that this denotes the E(n,n-k) function.

Example 4.1: Let $E(4,1)=x1 \oplus x2 \oplus x3 \oplus x4$. Apply the transformation $1 \rightleftarrows xi$ $(i=1,2,3,4)$, and we have $x2x3x4 \oplus x1x3x4 \oplus x1x2x4 \oplus x1x2x=E(4,3)$.
By Theorem 4.1, we have $T(4,1)=T(4,3)$. (End of Example)

V. Minimization Algorithm

An algorithm for MESOPs of up to 5 variables

1. Obtain the lower bound on the number of products in ESOPs for the given function. Let it be LB.
2. Expand the given function into one of the following forms:
   $f=\bar{x}ifi0 \oplus xifi1$, $f=xifi2 \oplus fi0$, or $f=\bar{x}ifi2 \oplus fi1$,
   where $fi2=fi0 \oplus fi1$.
3. Obtain the MESOPs of subfunctions for fi0, fi1, and fi2 from the table of MESOPs for 4-variable functions.
4. Obtain an ESOP for f by combining two MESOPs.
5. Simplify the ESOP by EXMIN [SAS 90b]. Let the number of products be τa.
6. If LB is equal to τa, then the simplified ESOP is the minimum. So stop the algorithm. Otherwise go to step 2, and try another expansion. If all the combinations are exhausted (15 possible expansions) and still LB $\neq \tau a$, then the minimality is not proved. Let the solution be the ESOP with minimum number of products.

```
/* Algorithm for Lower Bound */
for (all the variables ) {
   f=x̄ifi0 ⊕ xifi1,
   τi0 =τ(fi0)
   τi1 =τ(fi1)        /* Table Look up of MESOPs */
   τi2 =τ(fi2)
}
A = max { max(τi0, τi1, τi2)}
     i
B = min {τi0+τi1+τi2 - max(τi0, τi1, τi2)}
     i
if(A==B)
   LB=A
else
   LB=A+1
endif
```

VI. Experimental Results

6.1 Minimization of 4-variable functions [KOD 89]

We obtained MESOPs for all the 4-variable functions by an exhaustive method. There are 2^{16} =65536 functions of 4 variables. First, we made a table of 65536 entries each of which corresponds to a unique function of 4 variables. Then, we marked the entry for the function represented without a product, that is the constant zero function. Next, we marked the entries for the functions represented by one product. There are 3^4=81 such functions. Next, we marked the entries for the functions represented by two products. We repeated the similar procedure for up to six products. Because an arbitrary 4-variable function can be represented by an ESOP with at most 6 products, we can obtain the MESOP by this procedure.

Table 6.1 compares the numbers of functions requiring given numbers of products in MESOPs and MSOPs for n=4, where MSOPs were obtained by QM algorithm [SAS 84]. Note that in SOPs, 8 products are necessary to realize an arbitrary function, while in ESOPs only 6 products are necessary. This result also shows that ESOPs require fewer products than SOPs in many cases.

6.2 Minimization of 5-variable functions [KOD 90]

In the case of 5 variable functions, we cannot use the exhaustive method because the number of the combinations to consider is too large.
The set of the n-variable functions can be partitioned into 2^n+1 classes according to the number of minterms, where the k-th class consists of the functions with k minterms. (k=0,1,..., 2^n). For each class, we generated 1000 functions by using a random function generator. We simplified the ESOPs and obtained the number of products in ESOPs. Also, we obtained the lower bounds on the number of products in MESOPs. Table 6.2 shows the results. From this table, we can see the following:

1. When the number of minterms is less than 7 or more than 27, the algorithm proved the minimality of the ESOPs for more than 99% of the functions.
2. For the functions with 19 minterms, the algorithm proved the minimality for more than 27% of the functions.

VII. Conclusion

In this paper, we showed several properties of MESOPs. Also, we derived upper and lower bounds on the numbers of products in MESOPs for n-variable functions when the numbers of products in MESOPs for (n-1)-variable functions are available. We developed a minimization algorithm for ESOPs for 5-variable functions, which uses the table of MESOPs for 4-variable functions. The features of the algorithm are 1)to obtain a lower bound and 2) to stop the algorithm when the solutions is proved to be minimum. So the solutions are more reliable than ones obtained by the existing heuristic algorithms. This algorithm simplified given ESOPs and proved their minimality for about 30 percents of randomly generated functions of 5 variables.

Although various minimization algorithms for logical expressions have been developed, no algorithm used the minimized results of all the functions with fewer variables. By using the properties of MESOPs, we proved that MESOPs require only 16 products to realize an arbitrary function of 6-variables, while SOPs require 32 products [KOD 91]. Theorem 2.1 can be extended for the functions with multiple-valued inputs [SAS 91] to derive MESOPs and near minimum ESOPs [BRA 90]. We conjecture that ESOPs require fewer products than SOPs in most cases[SAS 90a]. However, thexe exist a 2n variable function whose MESOP requires 2^n-1 products while the MSOP requires only n products. Logic minimization programs have been an indispensable tools for the design and analysis of VLSI circuits. A good ESOP minimization algorithm will help the design of compact and easily testable VLSI circuits.

VIII. Acknowledgements

This work was supported in part by Grant in Aid for Scientific Research of the Ministry of Education, Science and Culture of Japan, and a research adjunct professorship at the Naval Postgraduate School, Monterey, California, U.S.A. Prof. Jon T.Butler arranged the stay in Monterey and constantly helped the author. Dr. Daniel. Brand improved the proof of Theorem 2.1. Mr.N.Koda developed the computer programs and did experiments. Prof. M.Davio noticed me very important results [DAV 78]. Profs. J.Muzio and M.R. Mukerjee sent me their latest papers([LUI 90] and [MUK 90]).

IX. References

[BES 83] Ph. W. Besslich,"Efficient computer method for ExOR logic design," IEE Proc., vol.130, Part E, pp.203-206, 1983.

[BES 85] Ph. W. Besslich,"Spectral processing of switching functions using signal-flow transformations, " in M.Karpovsky (Ed.), Spectral Techniques and Fault Detection, Orlando, FL:Academic Press,1985, pp.91-141.

[BIO 73] G.Bioul, M.Davio and J.P.Deschamps: "Minimization of ring-sum expansions of Boolean functions," Philips Res.Rpts., vol.28, pp. 17-36, 1973.

[DAV 78] M.Davio, J-P Deschamps, and A. Thayse, "Discrete and switching functions," McGraw-Hill International, 1978.

[EVE 67] S.Even, I.Kohavi and A.Paz "On minimal modulo-2 sum of products for switching functions," IEEE Trans. on Electron Computers, Vol. EC-16,pp.671-674, Oct. 1967.

[FLE 87] H.Fleisher, M.Tarvel, and J.Yeager, "A computer algorithm for minimizing Reed-Muller canonical forms, "IEEE Trans. on Computers Vol.C-36, No.2 Feb. 1987.

[GRE 76] D.H.Green and I.S.Taylor,"Multiple-valued switching circuit design by means of generalized Reed-Muller expansions," Digital Processes,vol.2, pp.63-81, 1976.

[HEL 88] M.Helliwell and M.Perkowski,"A fast algorithm to minimize multi-output mixed-polarity generalized Reed-Muller forms," Proc.of the 25th Design Automation Conference, pp.427-432, June 1988.

[KOD 89] N.Koda and T.Sasao, "Four-variable AND-EXOR minimum expressions and their properties"(in Japanese) Technical Paper of IEICE Japan, FTS89-25 , Oct 24, 1989.

[KOD 90] N.Koda and T.Sasao, "On the minimization of 5-variable AND-EXOR expressions"(in Japanese), National Convention of IEICE Japan, SA-3-3, October 2, 1990.

[KOD 91] N.Koda and T.Sasao, (in preparation).

[LIU 90] P.K.Lui and J.Muzio, "Boolean matrix transforms for the parity spectrum and the minimization of modulo-2 canonial expansions," University of Victoria, DCS-135-IR, July 1990.

[MUK 70] A. Mukhopadhyay and G.Schmitz,"Minimization of Exclusive OR and logical Equivalence of switching circuits," IEEE Trans. Comput., C-19,pp.132-140, 1970.

[MUL 54] D.E.Muller,"Application of Boolean algebra to switching circuit design and to error detection," IRE Trans. Electron. Comput., EC-3, pp.6-12, 1954.

[MUK 90] M.R.Mukerjee, "Minimization of ring-sum expansion of mixed polarity ", AMSE Sympo. on Modelling and Simulation, Greensboro, Oct. 1990.

[PAP 79] G.Papakonstantinou,"Minimization of modulo-2 sum of products," IEEE Trans. Comput., C-28, pp.163-167, 1979.

[PER 89] M.Perkowski, M.Helliwell, and P.Wu,"Minimization of multiple-valued input multi-output mixed-radix exclusive sum of products for incompletely specified Boolean functions," Proc.of the 19 th International Symposium on Multiple-valued Logic, pp.256-263, May 1989.

[RED 72] S.M.Reddy, "Easily testable realization for logic functions," IEEE Trans. Comput., C-21, pp.1083-1088, 1972.

[REE 54] I.S.Reed,"A class of multiple-error-correcting codes and the decoding scheme," IRE Trans. Information Theory PGIT-4. pp.38-49, 1954.

[ROB 82] J.P.Robinson and Chia-Lung Yeh, "A method for modulo-2 minimization", IEEE Trans. Comput., C-31,pp. 800-801, 1982.

[SAL 79] K.K.Saluja and E.H.Ong, "Minimization of Reed-Muller canonic expansion," IEEE Trans. Comput., C-28, pp.535-537, 1979.

[SAS 81] T.Sasao, "Multiple-valued decomposition of generalized Boolean functions and the complexity of programmable logic arrays," IEEE Trans. on Comput. vol.C-30, No.9, pp.635-643, Sept.1981.

[SAS 84] T.Sasao,"Input variable assignment and output phase optimization of PLA's," IEEE Trans. on Comput. vol C-33, No.10,pp.879-894, Oct. 1984.

[SAS 89]T.Sasao and M.Higashida "On a design algorithm for AND-EXOR PLA's with input decoders"(in Japanese), Technical paper of IEICE Japan, VLD89-84, Dec.15,1989.

[SAS 90a] T.Sasao and P.Besslich,"On the complexity of MOD-2 Sum PLA's,"IEEE Trans. on Comput. Vol.32. No.2, pp.262-266, Feb.1990.

[SAS 90b]T.Sasao, "EXMIN: A simplification algorithm for Exclusive-OR-Sum-of-Products Expressions for multiple-Valued input two-valued output functions," ISMVL-90, May 1990, pp.128-135.

[SAS 90c] D.Brand and T.Sasao,"On the minimization of AND-EXOR expressions"IEICE Technical Report, VLD90-, December 1990 (to be published).

[SAS 91]T.Sasao, "A transformation of multiple-valued input two-valued output functions and its application to simplification of exclusive-or sum-of-products expressions" (in preparation).

Table 6.1 Number of functions requiring a certain number of products in minimum expressions (n=4).

# of products	# of functions	
	MESOP	MSOP
0	1	1
1	81	81
2	2268	1804
3	21774	13472
4	37530	28904
5	3888	17032
6	24	3704
7	0	512
8	0	26

Table 6.2 Average number of products, lower bounds, and number of functions whose minimality are proved.

#mt	#pt	LB	Proof	#mt	#pt	LB	Proof
2	1.847	1.847	1000	17	6.366	5.536	282
3	2.561	2.561	1000	18	6.310	5.568	349
4	3.173	3.173	1000	19	6.427	5.588	270
5	3.692	3.690	998	20	6.381	5.592	300
6	4.145	4.138	993	21	6.385	5.589	286
7	4.565	4.529	964	22	6.240	5.553	380
8	4.873	4.756	883	23	6.150	5.542	425
9	5.179	4.992	814	24	5.868	5.406	561
10	5.339	5.016	688	25	5.549	5.263	717
11	5.655	5.210	571	26	5.131	5.020	889
12	5.758	5.248	512	27	4.714	4.664	950
13	5.972	5.387	464	28	4.161	4.161	1000
14	5.998	5.322	391	29	3.556	3.556	1000
15	6.198	5.445	353	30	2.831	2.831	1000
16	6.217	5.457	315	31	2.000	2.000	1000

#mt = number of minterms
#pt = average # of products in simplified ESOP's
LB = average lower bounds
Proof = number of functions whose minimality are proved(1000 functions for each #mt)