

# Fixed-Polarity OR-AND-EXOR Expressions and Their Minimization

TAKASHI HIRAYAMA,<sup>†</sup> KAZUYUKI NAGASAWA<sup>†</sup>  
and KENSUKE SHIMIZU<sup>††</sup>

We propose fixed-polarity OR-AND-EXOR expressions (FOAEs) and their minimization algorithm with  $O(4^n)$  time complexity, where  $n$  is the number of input variables. FOAEs can be used for the easily testable realization for the deterministic and/or random testing. We present experimental results of the number of terms of FOAEs and random testing for the FOAE circuits on the MCNC benchmark. The results show that the FOAE circuits have high potential testability for random testing.

## 1. Introduction

EXOR-based circuits have been studied because of the compactness<sup>5)</sup> and the easy testability<sup>3)</sup>. For the deterministic testing, many testable realizations with EXORs have been proposed<sup>2),3)</sup>. Recently, efficient properties of EXOR-based circuits for the random testing have been reported<sup>1)</sup>. The easy testability motivates the study of EXOR-based logic synthesis.

In this paper, fixed-polarity OR-AND-EXOR expressions (FOAEs) and their minimization algorithm are proposed. FOAEs are a generalization of FPRMs<sup>4),5)</sup> and SOAEs<sup>3)</sup>. Let  $n$  be the number of input variables and  $r$  be a constant such that  $1 \leq r \leq n$ . An FOAE corresponds to the three-level circuit such that the fan-ins of OR and AND gates are limited to  $r$  and  $\lceil n/r \rceil$ , respectively. The experimental results to compare the size and the testability of FOAEs with FPRMs and SOAEs are also given.

## 2. Basic Definitions and Properties

Let  $n$  be the number of input variables and  $r$  be a constant such that  $1 \leq r \leq n$ . OR-AND-EXOR expressions (OAEs) in this paper use OR terms with at most  $r$  literals. The constant  $r$  is called the fan-in parameter. The general form of an OR-AND term of OAEs is given by the following notation.

**Definition 1** The literals  $\bar{x}$  and  $x$  of a variable  $x$  may be denoted by  $x^{\{0\}}$  and  $x^{\{1\}}$ , respectively. The special literals  $x^{\{ \}}$  and  $x^{\{0,1\}}$  mean the constants 0 and 1, respectively.  $\square$

**Example 1** The OR term  $x_1 \vee \bar{x}_2 \vee x_4$  can

be written as  $x_1^{\{1\}} \vee x_2^{\{0\}} \vee x_3^{\{ \}} \vee x_4^{\{1\}}$ .  $\square$

By using the notation of Definition 1, the general form of an OR term with variables  $\{x_j, x_{j+1}, \dots, x_k\}$  ( $j \leq k$ ) is written as  $R(j, k) = x_j^{\beta_j} \vee x_{j+1}^{\beta_{j+1}} \vee \dots \vee x_k^{\beta_k}$  ( $\beta_i \subseteq \{0, 1\}$ ). By using  $R(j, k)$ , the general form of an OR-AND term with variables  $\{x_1, x_2, \dots, x_n\}$  is written as  $A(1, n) = R(1, r) \cdot R(r+1, 2r) \cdot \dots \cdot R(n-r+1, n)$ , where  $r$  is the fan-in parameter. A concrete OR-AND term  $A(1, n)$  is specified when the  $\{\beta_1, \beta_2, \dots, \beta_n\}$  is given. An EXOR combination of arbitrary OR-AND terms with fan-in parameter  $r$  is called an OAE with fan-in parameter  $r$ . Hence an OAE corresponds to an OR-AND-EXOR three-level circuit such that the fan-ins of OR and AND gates are limited to  $r$  and  $\lceil n/r \rceil$ , respectively. In some special cases, OAE circuits may result in two-level; for example AND-EXOR circuits are obtained if  $r = 1$ , and OR-EXOR circuits are obtained if  $r = n$ .

**Example 2**  $(\bar{x}_1 \vee x_2)(x_3 \vee x_4) \oplus x_1(x_3 \vee x_4) \oplus x_2 \bar{x}_4$  is an OAE with fan-in parameter  $r = 2$ .

**Definition 2** Let  $G$  be an expression. The set of all the positive and negative literals that appear in  $G$  is denoted by  $l(G)$ .  $G|_b^a$  is the expression obtained by replacing all the literals  $a$ 's in  $G$  with  $b$ 's.  $\square$

**Example 3** If  $G = \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 x_3$ ,  $l(G) = \{\bar{x}_1, x_1, \bar{x}_2, x_2, x_3\}$  and  $G|_{\bar{x}_3}^{x_3} = \bar{x}_1 x_2 \bar{x}_3 \vee x_1 \bar{x}_2 \bar{x}_3$ .

**Definition 3**  $V$  denotes an  $n$ -bit vector  $[v_1, v_2, \dots, v_n]$ . An OAE  $G$  is called a *Fixed-polarity OAE* (FOAE) with polarity vector  $V$  if  $x_i^{\{v_i\}} \notin l(G)$  for every  $i$  ( $1 \leq i \leq n$ ). Each  $v_i$  ( $1 \leq i \leq n$ ) in  $V$  is called the polarity of the variable  $x_i$ .  $\tau(G)$  represents the number of OR-AND terms connected by EXORs in an FOAE  $G$ .  $\square$

**Example 4** The OAE  $G = (\bar{x}_1 \vee x_2)(x_3 \vee$

<sup>†</sup> Ashikaga Institute of Technology

<sup>††</sup> Faculty of Engineering, Gunma University

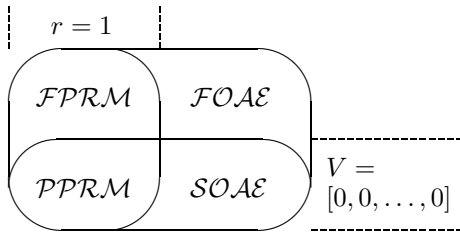


Fig. 1 Relation among classes.

$\bar{x}_4) \oplus (\bar{x}_1 \vee x_2)\bar{x}_4$  is an FOAE whose polarity vector is  $[1, 0, 0, 1]$ .  $\tau(G)$  is 2.  $\square$

FOAEs are a generalization of single-rail-input OR-AND-EXOR expressions (SOAEs)<sup>3)</sup>. SOAEs correspond to FOAEs with polarity vector  $[0, 0, \dots, 0]$ . FOAEs are also a generalization of fixed-polarity Reed-Muller expressions (FPRMs)<sup>4),5)</sup>. FPRMs correspond to FOAEs with fan-in parameter  $r = 1$ . The relation among these expressions is shown below.

**Property 1** *PPRM*, *FPRM*, *SOAE*, and *FOAE* represent the class of all PPRMs<sup>5)</sup>, FPRMs, SOAEs, and FOAEs, respectively. Then,  $PPRM \subset FPRM \subset FOAE$  and  $PPRM \subset SOAE \subset FOAE$  hold (Fig. 1).  $\square$

**Property 2** The FOAE of  $f$  is unique if the polarity vector  $V$  and the fan-in parameter  $r$  is specified.  $\square$

FOAEs can be applied to the design for testability (DFT). Since an FOAE consists of fixed polarity of literals for each variable, the easily testable realization for SOAEs<sup>3)</sup> can also be used to FOAEs. In the realization, all the single stuck-at faults in FOAE circuits are detected by fewer deterministic tests than FPRM circuits. For the random testing, the high testability of FOAE circuits is proved experimentally in this paper.

### 3. Minimization of OR-AND-EXOR Expressions

In this section, we present an algorithm to obtain minimum FOAEs.

We give a theorem which can be used to complement the polarities of an FOAE without changing the function.

**Theorem 1** Let  $G$  be an arbitrary expression. Let  $\dot{x}$  be either the positive literal  $x$  or the negative literal  $\bar{x}$  of a variable  $x$  of  $G$  and  $\bar{\dot{x}}$  be the complement of  $\dot{x}$ . Then, the following equation holds.

$$G = G|_{\dot{x}}^{\dot{x}} \oplus G|_0^{\dot{x}} \oplus G|_1^{\dot{x}} \quad (1)$$

**Proof:** Without loss of generality,  $\dot{x}$  and  $\bar{\dot{x}}$  are assumed to be the positive literal  $x$  and the

- (1) Obtain SOAE<sup>3)</sup>  $G$  of a given function  $f$ .
- (2)  $G_{min} \leftarrow G$ .
- (3) Do the following for each  $V \in \{0, 1\}^n$  in the Gray code order.
  - (3-1) Obtain  $G'$  by applying Eq. (1) to  $G$  according to the polarity change.
  - (3-2) If  $\tau(G') < \tau(G_{min})$  then  $G_{min} \leftarrow G'$ .
  - (3-3)  $G \leftarrow G'$ .
- (4) Return  $G_{min}$ .

Fig. 2 Minimization algorithm for FOAEs.

negative literal  $\bar{x}$ , respectively.

- (1) In the case of  $x = 0$ :  
 $G|_{\bar{x}}^x = G|_1^x$  holds from  $\bar{x} = 1$ . Hence  $G|_{\bar{x}}^x \oplus G|_0^x \oplus G|_1^x = G|_0^x \oplus G|_1^x = G|_1^x \oplus G|_0^x \oplus G|_1^x = G|_0^x$  holds. Since  $x = 0$ ,  $G = G|_0^x$  holds. Then we have  $G|_{\bar{x}}^x \oplus G|_0^x \oplus G|_1^x = G$ .
- (2) In the case of  $x = 1$ :  
 From  $\bar{x} = 0$ ,  $G|_{\bar{x}}^x \oplus G|_0^x \oplus G|_1^x = G|_0^x \oplus G|_0^x \oplus G|_1^x = G|_1^x = G$  is obtained in a similar way. Thus Theorem 1 is proved.  $\square$

When  $G$  is an OR-AND term,  $G|_{\bar{x}}^{\dot{x}}$ ,  $G|_0^{\dot{x}}$ , and  $G|_1^{\dot{x}}$  are also OR-AND terms.

**Example 5** For the OR-AND term  $G = (x_1 \vee x_2 \vee x_3)(x_4 \vee x_5 \vee x_6)(x_7 \vee x_8 \vee x_9)$ , complementing the polarity of the variable  $x_1$  results in  $G = G|_{\bar{x}_1}^{x_1} \oplus G|_0^{x_1} \oplus G|_1^{x_1} = (\bar{x}_1 \vee x_2 \vee x_3)(x_4 \vee x_5 \vee x_6)(x_7 \vee x_8 \vee x_9) \oplus (x_2 \vee x_3)(x_4 \vee x_5 \vee x_6)(x_7 \vee x_8 \vee x_9) \oplus (x_4 \vee x_5 \vee x_6)(x_7 \vee x_8 \vee x_9)$ .  $\square$

Generally, by complementing the polarity of a variable, a product term results in two product terms like  $\bar{x} = x \oplus 1$  while an OR-AND term results in three OR-AND terms like Example 5. Theorem 1 shows that a more complex expression also results in three expressions. So, some of synthesis techniques for FOAEs may be applied to other EXOR-based synthesis.

As a minimization algorithm for FPRMs, the Gray code method with  $O(4^n)$  is known<sup>6)</sup>. We present a similar minimization algorithm for FOAEs in Fig. 2. The computation for an FOAE of  $n = 4$  with  $r = 2$  is illustrated in the diagram of Fig. 3. From Property 2, the minimum FOAE with  $r$  is found by applying all the polarity vectors. In the diagram, polarity vectors are arranged in the Gray code order. At each polarity vector, exactly one polarity is changed from the previous vector. Therefore at most  $2^{n-1}$  OR-AND terms are expanded by Eq. (1) at each phase.

We evaluate the time complexity of this algorithm with the number of EXOR operations.

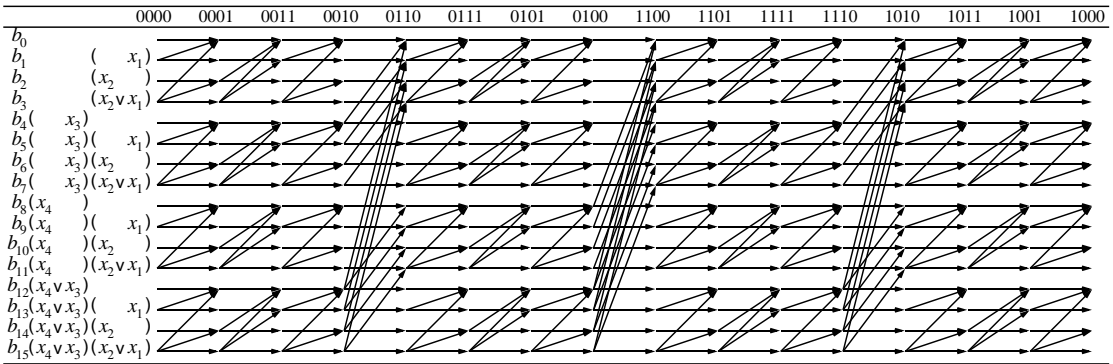


Fig. 3 Minimization of an FOAE of  $n = 4$  with fan-in parameter  $r = 2$ .

In Step (3-1) of Fig. 2, the number of EXOR operations is at most  $3 \cdot 2^{n-1}$  because, in  $G$ , there are at most  $2^{n-1}$  OR-AND terms concerned with Eq. (1) and each of them results in at most three OR-AND terms. Since the step is repeated  $2^n$  times in the Gray code order, the total number of EXOR operations is at most  $3 \cdot 2^{n-1} \cdot 2^n = (3/2)4^n$ . Thus the time complexity of the algorithm is  $O(4^n)$ . This means that the minimization of complex expressions such as FOAEs is achieved in the same time complexity of simpler expressions as FPRMs. This algorithm can be extended to multiple-output functions easily<sup>6)</sup>.

#### 4. Experimental Results

We implemented the algorithm in Lisp on SUN Ultra Sparc 60 Model 1450, and obtained minimum FOAEs for the MCNC benchmark set. In the following results, circuits with  $r = 1$  means FPRMs and circuits with  $r = 2, 3, \dots, n$  means FOAEs. **Figure 4** shows the  $\tau(G)$  of minimum FOAEs of the benchmark functions for each  $r = 1, 2, \dots, n$ . The smallest  $\tau(G)$  and its  $r$  for each benchmark function is presented at 'FO ( $r$ )' in **Table 1**. In the table, the computation time to obtain a minimum FOAE with fan-in parameter  $r$  is shown in seconds. From Fig. 4 and Table 1, FOAEs with some  $r$  consist of less terms than FPRMs whereas there is not much difference in the number of terms between FOAEs and FPRMs.

On the other hand, FOAEs have much higher random testability than FPRMs. To confirm the random testability of FOAE circuits, we applied randomly-generated test patterns to the FOAE circuits. **Figure 5** shows the results of the random testing for these circuits. The vertical axis 'Test length' represents the number of pseudo random patterns applied to the circuit.

For every circuit, the test patterns are applied until the fault coverage for the internal single stuck-at faults reaches 99.99%. Each test length is the average of 1000 experiments with different sequences of random patterns. These experimental results are plotted in approximation curves in Fig. 5. Although there are some exceptions such as 9sym and rd73, FOAE circuits ( $r \geq 2$ ) require much shorter test lengths than FPRM circuits ( $r = 1$ ). For large benchmarks such as table3 and alu4, the test lengths of the FOAE circuits can be 1/10 to 1/100 of that of FPRM circuits. In Fig. 5, the test lengths are especially short when  $r \approx \sqrt{n}$ . This is mainly due to the exclusion of OR and AND gates that have large fan-ins; those gates tend to be the bottleneck in random testing. In FOAE circuits with  $r \approx \sqrt{n}$ , fan-ins of both OR and AND gates result in relatively small ( $\approx \sqrt{n}$ ). Note that the fan-ins of OR and AND gates are  $r$  and  $\lceil n/r \rceil$ , respectively. Since the circuits in this experiment is the plain FOAE circuits without any DFT techniques, the results represent the high potential testability of FOAE circuits.

We also made experiments to compare FOAEs with SOAEs, and it was found that the test length is almost equal between them. On the other hand,  $\tau(G)$  of FOAEs can be smaller than or equal to that of SOAEs. For a function  $f$ , there exists only one SOAE if the fan-in parameter  $r$  is specified<sup>3)</sup>, whereas there exist  $2^n$  forms of FOAEs corresponding to  $2^n$  polarity vectors. Thus the minimum FOAE can be selected from them. For the value of  $r$  such that the test length is the shortest, the comparison of  $\tau(G)$  between SOAEs and FOAEs is shown at 'SO/FO ( $r$ )' in Table 1. On the average, FOAEs require 27% fewer terms than SOAEs.

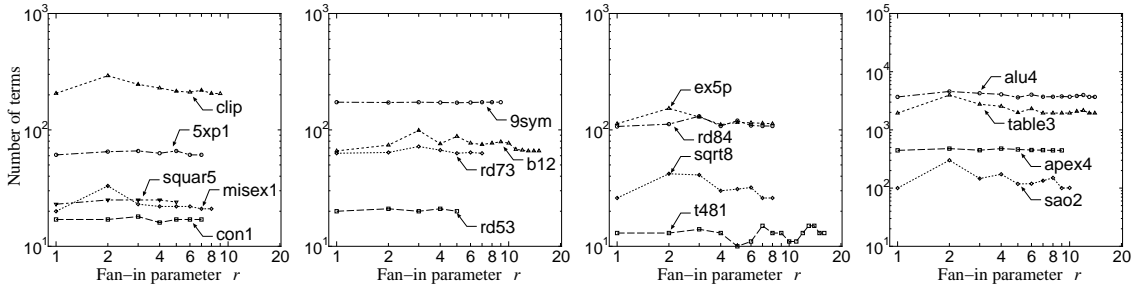


Fig. 4 Number of terms of FOAEs.

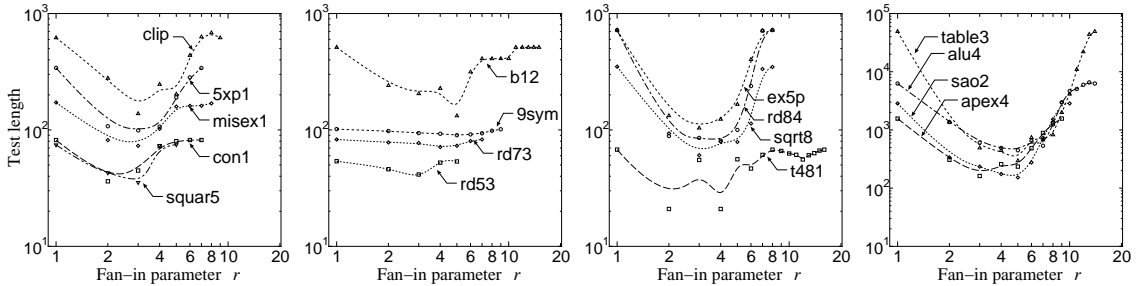


Fig. 5 Test lengths for FOAEs.

Table 1 Number of terms of the benchmarks.

Name	In/Out	FP	FO ( $r$ )	SO/FO ( $r$ )	Time
5xp1	7/10	61	61 (1,6,7)	93/66 (3)	1.93
9sym	9/1	173	171 (5)	212/171 (5)	3.66
alu4	14/8	3683	3627 (5)	5170/3627 (5)	2879
apex4	9/19	445	445 (1,8,9)	449/448 (3)	2.79
b12	15/9	66	66 (1,13-15)	120/88 (5)	983
clip	9/5	206	205 (9)	347/246 (3)	3.45
con1	7/2	17	16 (4)	26/17 (2)	60.3
ex5p	8/63	113	111 (4)	177/129 (3)	0.97
misex1	8/7	20	20 (1)	23/23 (3)	0.71
rd53	5/3	20	20 (1,3,5)	20/20 (3)	0.01
rd73	7/3	63	63 (1,5,7)	67/67 (4)	0.18
rd84	8/4	107	107 (1)	255/108 (4)	0.74
sao2	10/4	100	100 (1,9)	242/118 (5)	14.62
sqrt8	8/4	26	26 (1,7,8)	52/41 (3)	0.65
squar5	5/8	23	23 (1)	29/25 (3)	0.01
t481	16/1	13	10 (5)	41/13 (4)	306.7
table3	14/14	1945	1945 (1,7-10, 13,14)	4951/1984 (5)	3251

‘FP’=FPRM, ‘FO’=FOAE, ‘SO’=SOAE

5. Conclusions

We presented an efficient minimization algorithm for FOAEs and obtained the minimum FOAEs for the MCNC benchmark set. The results show that FOAEs require fewer terms than SOAEs and have higher potential testability for random testing than FPRMs. The highest testability is obtained by setting the parameter at  $r \approx \sqrt{n}$  while the value of  $r$  is decided by a trade-off between the costs and the testability. The realization costs for fan-in could be

different between OR and AND gates.

Our minimization algorithm is based on the Gray code method of FPRMs. As a minimization of FPRMs, a faster algorithm called the extended truth vector method<sup>4)</sup> is also known. The discussion whether the method is applicable to FOAEs is a future work.

References

- 1) Drechsler, R., et al.: Testability of 2-Level AND/EXOR Circuits, *Proc. European Design and Test Conference '97*, pp.548-553 (1997).
- 2) Fujiwara, H.: *Logic Testing and Design for Testability*, MIT Press (1985).
- 3) Hirayama, T., et al.: Easily Testable Realization Based on Single-Rail-Input OR-AND-EXOR Expressions, *IEICE Trans. Inf. & Syst.*, Vol.E82-D, No.9, pp.1278-1286 (1999).
- 4) Sasao, T. and Izuwara, F.: Exact Minimization of FPRMs Using Multi-Terminal EXOR TDDs, Sasao, T. and Fujita, M. (Eds.), *Representations of Discrete Functions*, pp.191-210, Kluwer Academic Publishers (1996).
- 5) Sasao, T.: *Switching Theory For Logic Synthesis*, Kluwer Academic Publishers (1999).
- 6) Zhang, Y.Z. and Rayner, P.J.W.: Minimization of Reed-Muller Polynomials with Fixed Polarity, *IEE Proc.*, Vol.131.Pt.E, No.5, pp.177-186 (1984).

(Received September 14, 2000)  
(Accepted February 1, 2001)