Engineering

Industrial & Management Engineering fields

Okayama University *Year* 1994

Minimum test sets for locally exhaustive testing of combinational circuits with five outputs

Tokumi Yokohira, Okayama University
Toshimi Shimizu, Okayama University
Hiroyuki Michinishi, Okayama University
Yuji Sugiyama, Okayama University
Takuji Okamoto, Okayama University

Minimum Test Sets for Locally Exhaustive Testing of Combinational Circuits with Five Outputs

Tokumi Yokohira, Toshimi Shimizu, Hiroyuki Michinishi, Yuji Sugiyama and Takuji Okamoto

Faculty of Engineering
Okayama University
3-1-1, Tsushima-naka, Okayama-shi, 700 Japan

Abstract

In this paper, features of dependence matrices of combinational circuits with five outputs are discussed, and it is shown that a minimum test set for locally exhaustive testing of such circuits always has 2^w test patterns, where w is the maximum number of inputs on which any output depends.

1 Introduction

Locally exhaustive testing has been proposed[1-5] as a method to decrease the number of test patterns while retaining the advantages of exhaustive testing in built-in self-test of multiple output combinational circuits (CUTs). In this testing, if an output y_i depends on w_i inputs ($1 \leqq i \leqq m$; m is the number of outputs), w_i-bit exhaustive patterns are applied to them. Any minimum test set (MLTS) therefore has at least 2^w test patterns, where $w \triangleq max\{w_1, w_2, \cdots, w_m\}$.

There has been few researches on the number of elements in an MLTS except the papers [6-8], in which it is clarified that every CUT with up to four outputs has an MLTS with 2^w elements. On the other hand, it can be easily shown that every CUT with more than five outputs does not have such an MLTS. It has not been however known whether every CUT with five outputs has such an MLTS or not.

In this paper, we show that every CUT with five outputs has an MLTS with 2^w test patterns. In Section 2, some terminologies and the concept of linear sum assignment[1] are described as preliminaries for the succeeding sections. In Section 3, features of dependence matrices of CUTs with $(w+1)$ inputs and five outputs are clarified. In Section 4, a theorem is established from the features that there exists a $5 \times w$ dependence matrix which is equivalent to each of the above matrices with respect to linear sum assignment. In Section 5, it is clarified from the theorem that every CUT with five outputs has an MLTS with 2^w test patterns.

2 Preliminaries[8]

2.1 Definitions of Terminologies

We will consider a combinational circuit under test (CUT) having n inputs $x_1, x_2, \cdots, x_n$, and m outputs $y_1, y_2, \cdots, y_m$. It is assumed that the CUT remains combinational even if any fault occurs. *A locally exhaustive test set* (LTS) for the CUT is defined as follows.

[Definition 1] An n-dimensional vector $(x_1, x_2, \cdots, x_n)$ is called *a test pattern.* If a set T of test patterns satisfies the following condition for every output y_i ($1 \leqq i \leqq m$), then it is an LTS.

Condition: If the output y_i depends on w_i inputs $x^i_1, x^i_2, \cdots, x^i_{w_i}$, then the projection of T onto $(x^i_1, x^i_2, \cdots, x^i_{w_i})$ subspace contains all of 2^{w_i} distinct binary patterns. ■

[Definition 2] *The dependence matrix D_C* for a CUT has m row vectors and n column vectors. The ijth element of D_C is 1 iff the output y_i depends on the input x_j, otherwise it is 0. ■

Note that the weight of the ith row vector in D_C is equal to w_i, and the maximum row weight of D_C is equal to w, where $w \triangleq max\{w_1, w_2, \cdots, w_m\}$.

[Definition 3] For $^\forall r$ ($r \geqq 1$), let t_p ($1 \leqq p \leqq r$) be a column vector with 2^r elements, and assume that the $2^r \times r$ matrix constructed with $t_1, t_2, \cdots, t_r$ has all of binary r-dimensional row vectors. Then, t_ps are called *base column vectors* and the set $\{t_1, t_2, \cdots, t_r\}$ is called *a base set.* ■

[Definition 4] A linear combination of the base column vectors, $k_1t_1 \oplus k_2t_2 \oplus \cdots \oplus k_rt_r$, is called *a linear sum*, where $k_1, k_2, \cdots, k_r \in \{0, 1\}$ and $(k_1, k_2, \cdots, k_r) \neq (0, 0, \cdots, 0)$. ■

Note that there exits $2^r - 1$ linear sums.

In this section, we implicitly assume, unless otherwise stated, that a base set is T^r ($\triangleq \{t_1, t_2, \cdots, t_r\}$), and that linear sums are linear combinations of $t_1, t_2, \cdots, t_r$.

[Definition 5] The set of q distinct linear sums $f_1, f_2, \cdots, f_q$ is called *q-independent* if the $2^r \times q$ matrix constructed with these linear sums has all of binary q-dimensional row vectors. ■

[Definition 6] Let G be a set of u linear sums $f_1, f_2, \cdots, f_u$, and assume that there exists such a mapping g from X ($\triangleq \{x_1, x_2, \cdots, x_n\}$) to G that it satisfies the following condition for every output y_i.

Condition: Let $x^i_1, x^i_2, \cdots, x^i_{w_i}$ denote the inputs on which the output y_i depends. If $g(x^i_j) = f^i_j$ ($1 \leqq j \leqq w_i$), then the set $\{f^i_1, f^i_2, \cdots, f^i_{w_i}\}$ is w_i-independent.

Then the CUT or the corresponding dependence matrix D_C is called *r-assignable*, and if $f_{u_1} = g(x_j)$, then it is called that *the linear sum f_{u_1} is assigned to the input x_j.* ■

0-8186-6690-0/94 $4.00 © 1994 IEEE

Note that, if a CUT is r-assignable, then an LTS with 2^r test patterns can be easily obtained.

[Definition 7] For a given linear sum set L ($\triangleq \{f_1, f_2, \cdots, f_q\}$), the set of all linear combinations of $f_1, f_2, \cdots, f_q$ is represented by $F(L)$ or $F(f_1, f_2, \cdots, f_q)$. ∎

For example, let $f_1 \triangleq t_1 \oplus t_2$, $f_2 \triangleq t_2 \oplus t_3$, $f_3 \triangleq t_3$ and $L \triangleq \{f_1, f_2, f_3\}$, then $f_1 \oplus f_2$, $f_2 \oplus f_3$, $f_3 \oplus f_1$, $f_1 \oplus f_2 \oplus f_3$ are represented as follows:

$$f_1 \oplus f_2 = t_1 \oplus t_3, \quad f_2 \oplus f_3 = t_2,$$
$$f_3 \oplus f_1 = t_1 \oplus t_2 \oplus t_3, \quad f_1 \oplus f_2 \oplus f_3 = t_1.$$

Therefore, $F(f_1, f_2, f_3) = F(L) = \{t_1, t_2, t_3, t_1 \oplus t_2, t_2 \oplus t_3, t_3 \oplus t_1, t_1 \oplus t_2 \oplus t_3\}$.

[Lemma 1] A given linear sum set L ($\triangleq \{f_1, f_2, \cdots, f_q\}$) is q-independent iff $|F(L)| = 2^q - 1$. ∎

[Lemma 2] Assume that, a given linear sum set $\{f_1, f_2, \cdots, f_{q-1}\}$ is $(q-1)$-independent ($q \leqq r$), and a linear sum f is not an element of $F(f_1, f_2, \cdots, f_{q-1})$. Then, the linear sum set $\{f_1, f_2, \cdots, f_{q-1}, f\}$ is q-independent. ∎

[Definition 8] For two linear sums f ($\triangleq k_1 t_1 \oplus k_2 t_2 \oplus \cdots \oplus k_r t_r$) and f' ($\triangleq k'_1 t_1 \oplus k'_2 t_2 \oplus \cdots \oplus k'_r t_r$), if $\sum_{p=1}^{r} k_p 2^{p-1} < \sum_{p=1}^{r} k'_p 2^{p-1}$, then it is called that f *is smaller than* f'. ∎

2.2 Linear Sum Assignment Algorithm

An LTS for an arbitrary CUT can be obtained using Akers' algorithm below.

(A-1) $r = w$.

(A-2) Select such an arbitrary output y_i that the weight of the ith row vector in the corresponding dependence matrix is equal to w, and assign t_j to each input x_j^i ($1 \leqq j \leqq w_i = w$).

(A-3) Repeat the following procedures (A-3.1) ~ (A-3.3) until a linear sum is assigned to every input.

(A-3.1) Select an arbitrary input x_j to which a linear sum has not been assigned, and find all output $y_{i_1}^j$, $y_{i_2}^j$, $\cdots$, $y_{i_c}^j$ which depend on x_j. Next, for each output $y_{i_v}^j$ ($1 \leqq v \leqq c$), find all inputs to which linear sums have been already assigned, and construct a set of such linear sums, $L_{i_v}^j$.

(A-3.2) Construct an set S^j according to the following equation.

$$S^j \triangleq F(L_{i_1}^j) \cup F(L_{i_2}^j) \cup \cdots \cup F(L_{i_c}^j).$$

(A-3.3) Construct $F(T^r)$, where $T^r \triangleq \{t_1, t_2, \cdots, t_r\}$. If $S^j \subset F(T^r)$, then execute the following procedure (A-3.3.1), otherwise execute the following procedure (A-3.3.2).

(A-3.3.1) Assign the smallest linear sum in the set $\overline{S^j}$ ($= F(T^r) - S^j$) to x_j.

(A-3.3.2) Assign t_{r+1} to x_j, and increase the value of r by 1.

(A-4) Construct the matrix with n linear sums which are assigned to the inputs, and consider it as a matrix representation of an LTS. ∎

In the succeeding sections, we will prove using the concept of Akers' algorithm that every CUT with five outputs has a minimum locally exhaustive test set (MLTS) with 2^w test patterns.

3 Features of Dependence Matrix with $m = 5$ and $w = n - 1$

In this section, unless otherwise stated, we implicitly assume the followings.

(1) A dependence matrix D_C^1 corresponding to a CUT with n inputs and five outputs is given, and $w = n - 1$. And, the followings are satisfied (see Figure 1).

(1-1) The first w columns in the fifth row are 1s, and the $(w+1)$st column in the fifth row is 0.

(1-2) The $(w+1)$st column is $(\underbrace{1, \cdots, 1}_{\alpha}, \underbrace{0, \cdots, 0}_{4-\alpha}, 0)^T$, where v^T represents the transpose of vector v.

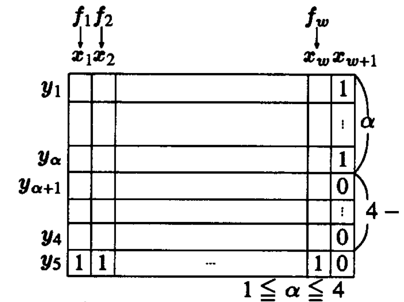


Figure 1 General form of dependence matrix with $(w+1)$ columns and five rows.

(2) A base set is T^w ($\triangleq \{t_1, t_2, \cdots, t_w\}$), and a linear sum is a linear combination of $t_1, t_2, \cdots, t_w$.

We consider application of Akers' algorithm to D_C^1 (see Figure 1). Assume that the output y_5 is selected and linear sums $f_1, f_2, \cdots, f_w$ are assigned to the inputs $x_1, x_2, \cdots, x_w$ respectively in the procedure (A-2), where the set F^w ($\triangleq \{f_1, f_2, \cdots, f_w\}$) is w-independent (note that $F(F^w) = F(T^w)$). In the procedure (A-3.1), x_{w+1} is selected as x_j and L_i^{w+1} are constructed ($1 \leqq i \leqq \alpha$; note that L_i^{w+1} is l_i-independent, where $l_i \triangleq |L_i^{w+1}|$). And then S^{w+1} ($= F(L_1^{w+1}) \cup F(L_2^{w+1}) \cup \cdots \cup F(L_\alpha^{w+1})$) is constructed in the procedure (A-3.2). Using F^w, L_i^{w+1} and S^{w+1}, the following four lemmas hold (for the simplicity, the superscript w+1 is removed from L_i^{w+1} and S^{w+1} in the discussion below).

[Lemma 3] For a given linear sum set $\{f_{j_1}, f_{j_2}, \cdots, f_{j_q}\}$ ($\subset F^w$), if $f_{j_1} \oplus f_{j_2} \oplus \cdots \oplus f_{j_q}$ is an element in $F(L_i)$, then $q \leqq l_i$ and $\{f_{j_1}, f_{j_2}, \cdots, f_{j_q}\} \subseteq L_i$. ∎

The proof of Lemma 3 is trivial.

[Lemma 4] There exists such a linear sum f in $F(F^w) - S$ that it is a linear combination of q linear sums $f_{j_1}, f_{j_2}, \cdots, f_{j_q}$ ($q \leqq \alpha$), where $f_{j_1}, f_{j_2}, \cdots, f_{j_q} \in F^w$. ∎

(Proof) Since $L_i \subset F^w$, there exists a linear sum $f_{j^*}^i$ in $\overline{L_i}$($= F^w - L_i$). If we select such an $f_{j^*}^i$ for each i, and create the set of the selected linear sums, then it has at most α elements. Let $\{f_{j^*}^{i_1}, f_{j^*}^{i_2}, \cdots, f_{j^*}^{i_q}\}$ ($\subset F^w$; $1 \leqq q \leqq \alpha$) be such set. If a linear sum $f_{j^*}^{i_1} \oplus f_{j^*}^{i_2} \oplus \cdots \oplus f_{j^*}^{i_q}$ is an element in $F(L_i)$ for $^\exists i$, then $\{f_{j^*}^{i_1}, f_{j^*}^{i_2}, \cdots, f_{j^*}^{i_q}\} \subseteq L_i$ from Lemma 3. This is contradiction, because at least one element in $\{f_{j_*}^{i_1}, f_{j_*}^{i_2}, \cdots, f_{j_*}^{i_q}\}$ is an element in $\overline{L_i}$. Therefore the linear sum is not an element in S, consequently the linear sum is an element in $F(F^w) - S$. Thus there exists the linear sum as f. ■

[Lemma 5] Let $f_{j_1} \oplus f_{j_2} \oplus \cdots \oplus f_{j_q}$ $(1 \leqq q \leqq \alpha)$ be a linear sum f in Lemma 4, and define H_f and F_i $(1 \leqq i \leqq \alpha)$ as follows:

$$H_f \triangleq \{f_{j_1}, f_{j_2}, \cdots, f_{j_q}\}, \quad F_i \triangleq \overline{L_i} \cap H_f.$$

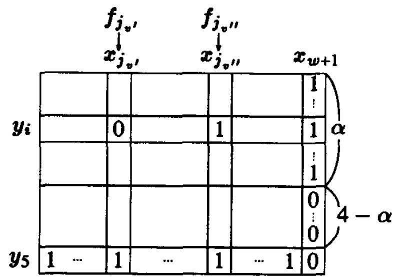
Then, F_i is not empty for $^\forall i$. ■

(Proof) Assume that $F_i = \phi$ for $^\exists i$. Since $F^w = L_i \cup \overline{L_i}$ and $H_f \subset F^w$, $H_f \subseteq L_i$. f is therefore an element in $F(L_i)$. This is contradiction. ■

[Lemma 6] Let f, H_f and F_i be the same definitions as those in Lemma 5, then the followings hold (see Figure 2).

(P0) If $f_{j_v} (\in H_f)$ is an element of F_i $(1 \leqq i \leqq \alpha)$, then the j_v th column of the ith row in D_C^1 is 0.

(P1) If $f_{j_v} (\in H_f)$ is not an element of F_i $(1 \leqq i \leqq \alpha)$, then the j_v th column of the ith row in D_C^1 is 1. ■



$f_{j_{v'}} \in F_i$, $f_{j_{v''}} \notin F_i$

Figure 2 The value of the j_vth column of the ith row in the case that $f_{j_v} \in F_i$ or $f_{j_v} \notin F_i$.

The proof of Lemma 6 is trivial from the definitions of L_i, f, H_f and F_i.

From Lemmas 4 ~ 6, the following two theorems can be obtained.

[Theorem 1](see Figure 3) Assume that a linear sum f in Lemma 4 is equal to f_{j_1}, then the j_1st column of the ith row in D_C^1 is 0 for $^\forall i$ $(1 \leqq i \leqq \alpha)$. ■

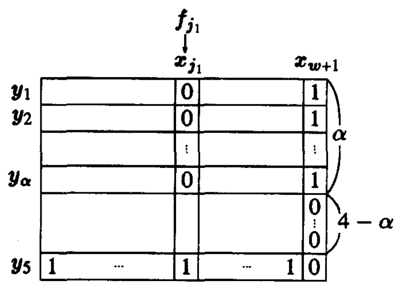


Figure 3 Values of the jth columns in the case that $f = f_{j_1}$.

(Proof) From the definition of H_f, $H_f = \{f_{j_1}\}$. Therefore, $F_i = \{f_{j_1}\}$ for $^\forall i$ from Lemma 5. Thus, Theorem 1 holds from (P0) of Lemma 6. ■

[Definition 9] For two distinct linear sums f $(\triangleq k_1 f_1 \oplus k_2 f_2 \oplus \cdots \oplus k_w f_w)$ and f' $(\triangleq k'_1 f_1 \oplus k'_2 f_2 \oplus \cdots \oplus k'_w f_w)$ which are elements in $F(F^w)$, if $k_p \leqq k'_p$ for $^\forall p$ $(1 \leqq p \leqq w)$, then it is called that f *is bitwise smaller than* f'. ■

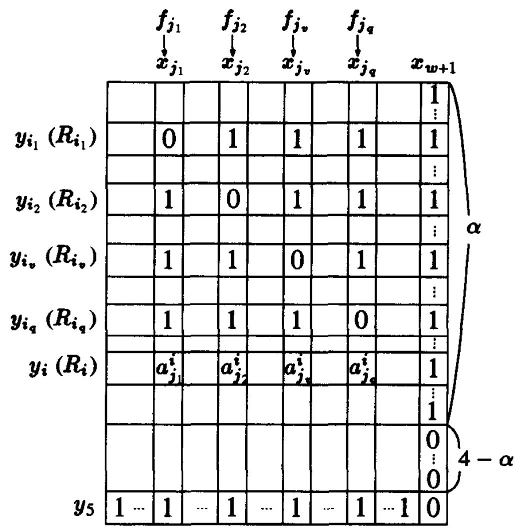
[Definition 10] For a given linear sum set L $(\subseteq F(F^w))$ and f $(\in L)$, if there does not exist such a linear sum in L that it is bitwise smaller than f, then it is called that f *is a bitwise minimum in* L. ■

For example, let $L \triangleq \{f_1 \oplus f_2, f_2 \oplus f_3, f_1 \oplus f_2 \oplus f_3\}$, then each of $f_1 \oplus f_2$ and $f_2 \oplus f_3$ is a bitwise minimum in L.

[Theorem 2](see Figure 4) Assume that a linear sum f in Lemma 4 is a linear combination of at least two linear sums, that is, $f = f_{j_1} \oplus f_{j_2} \oplus \cdots \oplus f_{j_v} \oplus \cdots \oplus f_{j_q}$ $(2 \leqq q \leqq \alpha)$, and f is a bitwise minimum in $\overline{S}$ $(= F(F^w) - S)$. Then, the followings hold.

(T1) For each v $(1 \leqq v \leqq q)$, there exists such a row R_{i_v} corresponding to an output y_{i_v} that, the j_vth column of the row is 0, and the other columns among the j_1st, the j_2nd, $\cdots$, the j_qth columns of the row are 1s.

(T2) Each of $(\alpha - q)$ rows obtained by removing $R_{i_1}, R_{i_2}, \cdots, R_{i_q}$ from α upper rows has at least one 0 among the j_1st, the j_2nd, $\cdots$, the j_qth columns. ■



$(a^i_{j_1}, a^i_{j_2}, \cdots, a^i_{j_v}, \cdots, a^i_{j_q}) \neq (1, 1, \cdots, 1)$

Figure 4 The value of the j_vth column of the ith row in the case that $f = f_{j_1} \oplus f_{j_2} \oplus \cdots \oplus f_{j_v} \oplus \cdots \oplus f_{j_q}$ $(2 \leqq q \leqq \alpha)$.

(Proof) A proof of (T1) is as follows: Since f is a bitwise minimum in $\overline{S}$, a linear sum f' $(\triangleq f_{j_1} \oplus f_{j_2} \oplus \cdots \oplus f_{j_{v-1}} \oplus f_{j_{v+1}} \oplus \cdots \oplus f_{j_q})$ which is constructed by removing f_{j_v} from f is an element in S. Assume that $f' \in F(L_{i_v})$ $(1 \leqq i_v \leqq \alpha)$. The set $\{f_{j_1}, f_{j_2}, \cdots, f_{j_{v-1}}, f_{j_{v+1}}, \cdots, f_{j_q}\}$ is a subset of L_{i_v} from Lemma 3. Therefore, each of $f_{j_1}, f_{j_2}, \cdots, f_{j_{v-1}}, f_{j_{v+1}}, \cdots, f_{j_q}$ is not an element in F_{i_v}. From (P1) of Lemma 6, the $j_{v'}$th column in the i_vth row of D_C^1

is 1 for $^\forall v'$ $(1 \leqq v' \leqq q\,;\, v' \neq v)$. If we assume that f_{j_v} is also an element in L_{i_v}, then $f \in F(L_{i_v})$. This is contradiction. Therefore, $f_{j_v} \notin L_{i_v}$. Thus, the j_vth column of the i_vth row is 0 from (P0) of Lemma 6. A proof of (T2) is as follows: If we assume that the $j_{v'}$th column of a row R_i is 1 for $^\forall v'$ $(1 \leqq v' \leqq q)$, then $\{ f_{j_1}, f_{j_2}, \cdots, f_{j_q} \} \subseteq L_i$, consequently $f \in F(L_i)$. This is contradiction. ■

Figure 5 summarizes the results given by Theorem 2, where $j_1 \sim j_q$ and $i_1 \sim i_q$ in Theorem 2 are assumed without loss of generality that $j_1 < j_2 < \cdots < j_q$ and $i_v = v (1 \leqq v \leqq q)$, respectively.

		f_{j_1} ↓ x_{j_1}		f_{j_2} ↓ x_{j_2}		x_{w+1}
y_1		0		1		1
y_2		1		0		1
y_3						0
y_4						0
y_5	1 ···	1	···	1	··· 1	0

(a) $\alpha = 2,\ f = f_{j_1} \oplus f_{j_2}$

		f_{j_1} ↓ x_{j_1}		f_{j_2} ↓ x_{j_2}		x_{w+1}
y_1		0		1		1
y_2		1		0		1
y_3		$a^3_{j_1}$		$a^3_{j_2}$		1
y_4		$a^4_{j_1}$		$a^4_{j_2}$		0
y_5	1 ···	1	···	1	··· 1	0

(b) $\alpha = 3,\ f = f_{j_1} \oplus f_{j_2},\ (a^3_{j_1}, a^3_{j_2}) \neq (1,1)$

		f_{j_1} ↓ x_{j_1}		f_{j_2} ↓ x_{j_2}		f_{j_3} ↓ x_{j_3}		x_{w+1}
y_1		0		1		1		1
y_2		1		0		1		1
y_3		1		1		0		1
y_4								0
y_5	1 ···	1	···	1	···	1	··· 1	0

(c) $\alpha = 3,\ f = f_{j_1} \oplus f_{j_2} \oplus f_{j_3}$

		f_{j_1} ↓ x_{j_1}		f_{j_2} ↓ x_{j_2}		x_{w+1}
y_1		0		1		1
y_2		1		0		1
y_3		$a^3_{j_1}$		$a^3_{j_2}$		1
y_4		$a^4_{j_1}$		$a^4_{j_2}$		1
y_5	1 ···	1	···	1	··· 1	0

(d) $\alpha = 4,\ f = f_{j_1} \oplus f_{j_2},\ (a^3_{j_1}, a^3_{j_2}) \neq (1,1),$
$(a^4_{j_1}, a^4_{j_2}) \neq (1,1)$

		f_{j_1} ↓ x_{j_1}		f_{j_2} ↓ x_{j_2}		f_{j_3} ↓ x_{j_3}		x_{w+1}
y_1		0		1		1		1
y_2		1		0		1		1
y_3		1		1		0		1
y_4		$a^4_{j_1}$		$a^4_{j_2}$		$a^4_{j_3}$		1
y_5	1 ···	1	···	1	···	1	··· 1	0

(e) $\alpha = 4,\ f = f_{j_1} \oplus f_{j_2} \oplus f_{j_3},\ (a^4_{j_1}, a^4_{j_2}, a^4_{j_3}) \neq (1,1,1)$

		f_{j_1} ↓ x_{j_1}		f_{j_2} ↓ x_{j_2}		f_{j_3} ↓ x_{j_3}		f_{j_4} ↓ x_{j_4}		x_{w+1}
y_1		0		1		1		1		1
y_2		1		0		1		1		1
y_3		1		1		0		1		1
y_4		1		1		1		0		1
y_5	1 ···	1	···	1	···	1	···	1	··· 1	0

(f) $\alpha = 4,\ f = f_{j_1} \oplus f_{j_2} \oplus f_{j_3} \oplus f_{j_4}$

Figure 5 Summary of Theorem 2.

4 Equivalent Dependence Matrix

In this section, we show that there exists such a dependence matrix with w columns that it is equivalent to D^1_C described in the preceding section if a bitwise minimum f in $F(F^w) - S$ is assigned to the input x_{w+1}.

First, we define equivalence relation between dependence matrices as follows:

[Definition 11] For two dependence matrices D^1_C and D^2_C with arbitrary number of columns, D^1_C and D^2_C are said to be equivalent iff the followings hold.

(E1) The number of rows (m) in D^1_C is equal to that in D^2_C.

(E2) The row weight (w_i) of the ith row in D^1_C is equal to that in D^2_C for $^\forall i$ $(1 \leqq i \leqq m)$.

(E3) Linear sums can be assigned to the inputs in D^1_C and D^2_C so that the following conditions are satisfied for $^\forall i$ $(1 \leqq i \leqq m)$.

Condition: Let K^1_i and K^2_i $(|K^1_i| = |K^2_i| = w_i)$ be sets of linear sums which are assigned to the inputs on which the outputs y_is depend in D^1_C and D^2_C, respectively. Then both K^1_i and K^2_i are w_i-independent, and $F(K^1_i) = F(K^2_i)$. ■

We next give an algorithm to construct a dependence matrix D^2_C with w columns which is equivalent to D^1_C described in the preceding section under the condition that linear sums $f_1 \sim f_w$ and a bitwise minimum f in $F(F^w) - S$ are assigned to the inputs $x_1 \sim x_w$ and x_{w+1} in D^1_C, respectively. As mentioned in the preceding section, we assume without loss of generality that D^1_C is one of matrices of the types illustrated in Figures 3 and 5(a)-(f).

[Algorithm to construct D^2_C] According to the type of D^1_C, do one of the followings (1) ~ (5) (let a^i_j denote the value of the jth column of the ith row in D^1_C), and let D^2_C be the matrix obtained from the resultant matrix by removing the $(w+1)$st column.

(1) In the case that D_C^1 is of the type illustrated in Figure 3, change $a_{j_1}^i$ into 1 for every i $(1 \leqq i \leqq \alpha)$.

(2) In the case that D_C^1 is one of matrices of the types illustrated in Figure 5(a), (c) and (f), change $a_{j_i}^i$ into 1 for every i $(1 \leqq i \leqq \alpha)$.

(3) In the case that D_C^1 is of the type illustrated in Figure 5(b), do the following procedures (3.1) and (3.2).

(3.1) Change $a_{j_i}^i$ into 1 for every i $(i = 1, 2)$.

(3.2) (3.2.1) If $(a_{j_1}^3, a_{j_2}^3) = (0, 1)$ or $(1, 0)$, then change an $a_{j_v}^3$ whose value is $0 (v = 1$ or $2)$ into 1.

(3.2.2) If $(a_{j_1}^3, a_{j_2}^3) = (0, 0)$ and $(a_{j_1}^4, a_{j_2}^4) \neq (1, 0)$, then change $a_{j_1}^3$ into 1, and change f_{j_1} which is assigned to x_{j_1} into $f_{j_1} \oplus f_{j_2}$.

(3.2.3) Otherwise, that is, $(a_{j_1}^3, a_{j_2}^3) = (0, 0)$ and $(a_{j_1}^4, a_{j_2}^4) = (1, 0)$, then change $a_{j_2}^3$ into 1, and change f_{j_2} which is assigned to x_{j_2} into $f_{j_1} \oplus f_{j_2}$.

(4) In the case that D_C^1 is of the type illustrated in Figure 5(d), do the following procedures (4.1) and (4.2).

(4.1) Change $a_{j_i}^i$ into 1 for every i $(i = 1, 2)$.

(4.2) (4.2.1) If $(a_{j_v}^3, a_{j_v}^4) = (0, 0)$ for $v = 1$ or 2, then select such a v, and change $a_{j_v}^3$ and $a_{j_v}^4$ into 1, and change f_{j_v} which is assigned to x_{j_v} into $f_{j_1} \oplus f_{j_2}$.

(4.2.2) Otherwise, that is, $(a_{j_v}^3, a_{j_v}^4) = (0, 1)$ or $(1, 0)$ for $v = 1$ and 2, then change elements whose values are 0s among $a_{j_1}^3$, $a_{j_2}^3$, $a_{j_1}^4$ and $a_{j_2}^4$ into 1s.

(5) In the case that D_C^1 is of the type illustrated in Figure 5(e), do the following procedures (5.1) and (5.2).

(5.1) Change $a_{j_i}^i$ into 1 for every i $(i = 1, 2, 3)$.

(5.2) Select an $a_{j_v}^4$ $(1 \leqq v \leqq 3)$ whose value is 0, and change $a_{j_v}^4$ into 1, and change f_{j_v} which assigned to the input x_{j_v} into $f_{j_1} \oplus f_{j_2} \oplus f_{j_3}$. ■

Before proving that D_C^2 is equivalent to D_C^1, we give the following lemma.

[Lemma 7] For a set F_a $(\subseteq F^w)$, assume that $F_a \supseteq \{f_{j_1}, f_{j_2}, \cdots, f_{j_u}\}$ $(u \geqq 2)$. And let F_b be $(F_a - \{f_{j_v}\}) \cup \{f_{j_1} \oplus f_{j_2} \oplus \cdots \oplus f_{j_u}\}$ $(1 \leqq v \leqq u)$. Then $F(F_b) = F(F_a)$. ■

The proof is trivial. Note that since F_a is q-independent, F_b is also q-independent from Lemmas 1 and 7, where $q \triangleq |F_a|$.

[Theorem 3] D_C^2 is equivalent to D_C^1. ■

(Proof) It is trivial that the conditions (E1) and (E2) are satisfied. Since a bitwise minimum f is assigned to x_{w+1}, K_i^1 $(1 \leqq i \leqq 5)$ is w_i-independent from Lemma 2. Therefore, from Lemma 1, if $F(K_i^1) = F(K_i^2)$, then K_i^2 is also w_i-independent.

According to the cases (1) ~ (5) in the algorithm above, it can be proved that $F(K_i^1) = F(K_i^2)$. We show a proof only in the case (3) due to space limitation. Let X_i^k $(1 \leqq i \leqq 5; k = 1, 2)$ be a set of the inputs on which the output y_i depends in D_C^k.

(A) In the case that (3.2.1) in the algorithm is executed.

A proof for $(a_{j_1}^3, a_{j_2}^3) = (0, 1)$ is as follows: X_1^1 does not contain x_{j_1}, and contains x_{j_2} and x_{w+1}. The algorithm changes $a_{j_1}^1$ into 1 and removes the $(w+1)$st columns. Therefore, $X_1^1 - \{x_{w+1}\} = X_1^2 - \{x_{j_1}\}$. On the other hand, f_{j_1} and $f_{j_1} \oplus f_{j_2}$ are assigned to x_{j_1} and x_{w+1}, respectively. Therefore, $K_1^1 - \{f_{j_1} \oplus f_{j_2}\} = K_1^2 - \{f_{j_1}\}$, consequently, $F(K_1^1) = F(K_1^2)$ from Lemma 7. Similarly, we can prove that $F(K_i^1) = F(K_i^2)$ for $i = 2$ and 3. For $i = 4$ or 5, a_{w+1}^i is 0, and the algorithm does not change the first w columns. Therefore $F(K_i^1) = F(K_i^2)$.

Similarly, we can prove for $(a_{j_1}^3, a_{j_2}^3) = (1, 0)$.

(B) In the case that (3.2.2) in the algorithm is executed.

X_1^1 does not contain x_{j_1}, and contains x_{j_2} and x_{w+1} to which f_{j_2} and $f_{j_1} \oplus f_{j_2}$ are assigned in D_C^1, respectively. From the algorithm, X_1^2 contains x_{j_1} and x_{j_2} to which $f_{j_1} \oplus f_{j_2}$ and f_{j_2} are assigned in D_C^2, respectively, and does not contain x_{w+1}. Therefore, $K_1^1 = K_1^2$, consequently $F(K_1^1) = F(K_1^2)$.

For the second row, we can similarly obtain that $K_2^1 - \{f_{j_1}, f_{j_1} \oplus f_{j_2}\} = K_2^2 - \{f_{j_1} \oplus f_{j_2}, f_{j_2}\}$. Let $K \triangleq (K_2^1 - \{f_{j_1} \oplus f_{j_2}\}) \cup \{f_{j_2}\}$, then the following equations hold.

$$K - \{f_{j_2}\} = K_2^1 - \{f_{j_1} \oplus f_{j_2}\}.$$
$$K - \{f_{j_1}\} = K_2^2 - \{f_{j_1} \oplus f_{j_2}\}.$$

From Lemma 7, therefore, $F(K_2^1) = F(K_2^2)$.

For the third row, X_3^1 does not contain x_{j_1} and contains x_{w+1} to which $f_{j_1} \oplus f_{j_2}$ is assigned in D_C^1. From the algorithm, X_3^2 contains x_{j_1} to which $f_{j_1} \oplus f_{j_2}$ is assigned in D_C^2. Therefore, $K_3^1 = K_3^2$, consequently $F(K_3^1) = F(K_3^2)$.

For the fourth row, if $(a_{j_1}^4, a_{j_2}^4) = (0, 0)$ or $(0, 1)$, then it is trivial that $K_4^1 = K_4^2$, consequently $F(K_4^1) = F(K_4^2)$. If $(a_{j_1}^4, a_{j_2}^4) = (1, 1)$, then f_{j_1} assigned to x_{j_1} in D_C^1 is changed into $f_{j_1} \oplus f_{j_2}$ in D_C^2. Therefore, $K_4^1 - \{f_{j_1}\} = K_4^2 - \{f_{j_1} \oplus f_{j_2}\}$. From Lemma 7, $F(K_4^1) = F(K_4^2)$. The same argument holds for the fifth row.

(C) In the case that (3.2.3) in the algorithm is executed.

The argument in (B) similarly holds. ■

5 The Number of Elements in Minimum Test Set

In this section, we prove that every CUT with five outputs has an MLTS with 2^w test patterns.

Without loss of generality, we assume that the first w columns of the fifth row in a given dependence matrix D_C are 1s. Then, the following theorem is derived from Theorem 3.

[Theorem 4] If a linear sum which is a linear combination of $t_1, t_2, \cdots, t_w$ is assigned to each input in D_C using the following algorithm, then a set of linear sums assigned to the inputs on which the output y_i depends is w_i-independent ($1 \leqq i \leqq 5$).

(1) Assign base column vectors $t_1, t_2, \cdots, t_w$ to $x_1, x_2, \cdots, x_w$ in D_C, respectively, and create a dependence matrix $D_C^{(0)}$ by copying the first w columns in D_C keeping the assignment.

(2) $j = 1$.

(3) Repeat the following procedures (3.1) $\sim$ (3.5) until a linear sum is assigned to every input in D_C.

(3.1) If the $(w + j)$th column in D_C is not the form of $(\underbrace{1, \cdots, 1}_{\alpha}, \underbrace{0, \cdots, 0}_{4-\alpha}, 0)^T$, then rearranging four rows except for the fifth one in this form and rearrange the corresponding rows of $D_C^{(j-1)}$ in the same interchanges as D_C. If the $(w + j)$th column in D_C is the form of $(\underbrace{1, \cdots, 1}_{\alpha}, \underbrace{0, \cdots, 0}_{4-\alpha}, 0)^T$, then keep both D_C and $D_C^{(j-1)}$ unchanged, and go to the procedure (3.2).

(3.2) Concatenate $D_C^{(j-1)}$ with the $(w+j)$th column in D_C. And considering the concatenated matrix and linear sums which are assigned to the inputs corresponding to the first w columns in the concatenated matrix as D_C^1 and $f_1, f_2, \cdots, f_w$ in Sections 3, respectively, create $F(F^w)$ and S.

(3.3) Assign a bitwise minimum f in $F(F^w) - S$ to the input x_{w+j} in D_C.

(3.4) Assign f to the input x_{w+j} in D_C^1 (the input corresponding to the $(w + 1)$st column in D_C^1), and create D_C^2 using the algorithm in Section 4.

(3.5) $D_C^{(j)} = D_C^2$, and increase the value of j by 1. ■

(Proof) If $n = w$, then the proof is trivial. We therefore assume that $n \geqq w + 1$. Let $M^{(j-1)}$ ($1 \leqq j \leqq n - w + 1$) be the matrix constructed with the first $(w + j - 1)$st columns in D_C.

(1) The argument below holds in the first visit of the procedure (3).

Since $D_C^{(0)} = M^{(0)}$ and a base column vector t_{j_1} ($1 \leqq j_1 \leqq w$) is assigned to the input x_{j_1}, the set of $f_1, f_2, \cdots, f_w$ in the procedure (3.2) is w-independent. Therefore, the procedures (3.3) and (3.4) are executable and $D_C^{(0)}$ and $M^{(0)}$ are equivalent.

Since $D_C^{(0)}$ and $M^{(0)}$ are equivalent and a bitwise minimum f is assigned to the input x_{w+1} in D_C by the procedure (3.3) and the input x_{w+1} in D_C^1 by the procedure (3.4), D_C^1 and $M^{(1)}$ are equivalent. On the other hand, from Theorem 3, $D_C^2 (= D_C^{(1)})$ and D_C^1 are equivalent. Therefore, $D_C^{(1)}$ and $M^{(1)}$ are equivalent.

(2) If we assume that $D_C^{(j-1)}$ and $M^{(j-1)}$ ($2 \leqq j \leqq n - w$) are equivalent, then the argument below holds in the jth visit of the procedure (3).

Since $D_C^{(j-1)}$ and $M^{(j-1)}$ are equivalent, from the condition (E3) in the definition 11, K_5^1 in D_C^1 of the procedure (3.2) is w_5-independent ($w_5 = w$). Therefore, the set of $f_1, f_2, \cdots, f_w$ in the procedure (3.2) is w-independent. Therefore, the procedures (3.3) and (3.4) are executable.

Since $D_C^{(j-1)}$ and $M^{(j-1)}$ are equivalent and a bitwise minimum f is assigned to the input x_{w+j} in D_C by the procedure (3.3) and the input x_{w+j} in D_C^1 by the procedure (3.4), D_C^1 and $M^{(j)}$ are equivalent. On the other hand, from Theorem 3, $D_C^2 (= D_C^{(j)})$ and D_C^1 are equivalent. Therefore, $D_C^{(j)}$ and $M^{(j)}$ are equivalent.

(3) From (1) and (2), by induction, $D_C^{(n-w)}$ and $M^{(n-w)}$ (D_C itself) are equivalent. Therefore, from the condition (E3) in the definition 11, a set of linear sums assigned to the inputs on which the output y_i depends in D_C is w_i-independent. ■

From the definition 6 and Theorem 4, every CUT with five outputs is w-assignable. Therefore, we can conclude that every CUT with five outputs has an MLTS with 2^w test patterns.

6 Conclusion

In this paper, we showed that every CUT with five outputs has an MLTS with 2^w test patterns. From the result, it can be concluded that while every CUT with more than five outputs does not have such an MLTS, every CUT with up to five outputs has such an MLTS.

References

[1] S. B. Akers, "On The Use of Linear Sums in Exhaustive Testing," *Digest of Papers 15th Symp. on Flt.Tolerant Comp.*, pp. 148-153, June 1985.

[2] H. Hiraishi, K. Kawahara and S. Yajima, "Locally Exhaustive Testing of Combinational Circuits Using Linear Logic Circuits," *Journal of Information Processing*, VOL. 11, No. 3, pp. 191-198, Nov. 1988.

[3] D. T. Tang and L. S. Woo, "Exhaustive Test Pattern Generation with Constant Weight Vector," *IEEE Trans. Comput.*, VOL. C-32, No. 12, pp. 1145-1150, Dec. 1983.

[4] L. T. Wang and E. J. McCluskey, "Linear Feedback Shift Register Design Using Cyclic Code," *IEEE Trans. Comput.*, VOL. C-37, No. 10, pp. 1302-1306, Oct. 1988.

[5] E. J. McCluskey, "Verification Testing — A Pseudoexhaustive Test Technique," *IEEE Trans. Comput.*, VOL. C-33, No. 6, pp. 541-546, June 1984.

[6] H. Michinishi, T. Yokohira and T. Okamoto, "Minimum Verification Test Set for Combinational Circuit," *Proc. ATS '92*, pp. 14-19, Nov. 1992.

[7] H. Michinishi, T. Yokohira and T. Okamoto, "Minimum Test Set for Locally Exhaustive Testing of Multiple Output Combinational Circuits," *IEICE Trans.* Inf. & Syst., VOL. E76-D, No. 7, pp. 791-799, July 1993.

[8] H. Michinishi, T. Yokohira and T. Okamoto, "Proof that Akers' Algorithm for Locally Exhaustive Testing Gives Minimum Test Sets of Combinational Circuits with up to Four Outputs," *Proc. ATS '93*, pp. 14-19, Nov. 1993.