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

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

This paper is posted at eScholarship@OUDIR : Okayama University Digital Information Repository.
http://escholarship.lib.okayama-u.ac.jp/industrial_engineering/60
Minimum Test Sets for Locally Exhaustive Testing of Combinational Circuits with Five Outputs

Tokumi Yokohira, Toshimi Shiozawa, 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 $2^m$ test patterns, where $m$ is the maximum number of inputs on which any output depends.

1 Introduction
Locally exhaustive testing has been proposed 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 ($1 \leq i \leq m$; $m$ is the number of outputs), $w_i$-bit exhaustive patterns are applied to them. Any minimum test set (MLTS) therefore has at least $2^m$ test patterns, where $w = \max\{w_1, w_2, \ldots, w_m\}$.

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 $2^m$ 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 $2^m$ test patterns. In Section 2, some terminologies and the concept of linear sum assignment 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 $2^m$ test patterns.

2 Preliminaries

2.1 Definitions of Terminologies
We will consider a combinational circuit under test (CUT) having $n$ inputs $x_1, x_2, \ldots, x_n$, and $m$ outputs $y_1, y_2, \ldots, y_m$. 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 $(x_1, x_2, \ldots, x_n)$ is called a test pattern. If a set $T$ of test patterns satisfies the following condition for every output $y_i$ ($1 \leq i \leq m$), then it is an LTS. Condition: If the output $y_i$ depends on $w_i$ inputs $x_1^i, x_2^i, \ldots, x_{w_i}^i$, then the projection of $T$ onto $(x_1^i, x_2^i, \ldots, x_{w_i}^i)$ 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 $j$th element of $D_C$ is 1 iff the output $y_j$ depends on the input $x_j$, otherwise it is 0.

Note that the weight of the $i$th row vector in $D_C$ is equal to $w_i$, and the maximum row weight of $D_C$ is equal to $w$, where $w = \max\{w_1, w_2, \ldots, w_m\}$.

[Definition 3] For $r \geq 1$, let $t_p$ ($1 \leq p \leq r$) be a column vector with $2^r$ elements, and assume that the $2^r \times r$ matrix constructed with $t_1, t_2, \ldots, t_r$ has all of binary $r$-dimensional row vectors. Then, $t_p$'s are called base column vectors and the set $\{t_1, t_2, \ldots, t_r\}$ 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 $k_1, k_2, \ldots, k_r \in \{0,1\}$ and $(k_1, k_2, \ldots, k_r) \neq (0, 0, \ldots, 0)$.

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

In this section, we implicitly assume, unless otherwise stated, that a base set is $T_r = \{t_1, t_2, \ldots, t_r\}$, and that linear sums are linear combinations of $t_1, t_2, \ldots, t_r$.

[Definition 5] The set of $q$ distinct linear sums $f_1, f_2, \ldots, f_q$ 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 $f_1, f_2, \ldots, f_u$, and assume that there exists such a mapping $g$ from $X = \{x_1, x_2, \ldots, x_n\}$ to $G$ that it satisfies the following condition for every output $y_i$.
Condition: Let $x_j^i$, $x_{j1}, \ldots, x_{jw_i}$ denote the inputs on which the output $y_i$ depends. If $g(x_j^i) = f_j^i$ ($1 \leq j \leq w_i$), then the set $\{f_1, f_2, \ldots, f_u\}$ is $w_i$-independent.
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 \( L (\triangleq \{ f_1, f_2, \ldots, f_q \}) \), the set of all linear combinations of \( f_1, f_2, \ldots, f_q \) is represented by \( F(L) \) or \( F(f_1, f_2, \ldots, f_q) \).

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 \( f_1 \oplus f_2, f_2 \oplus f_3, f_1 \oplus f_1, f_1 \oplus f_2 \oplus f_3 \) are represented as follows:

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

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

[Lemma 1] \( L \) is \( q \)-independent if \( |F(L)| = 2^q - 1 \).

[Lemma 2] Assume that a given linear sum set \( \{ f_1, f_2, \ldots, f_{q-1} \} \) is \((q-1)\)-independent \((q \leq r)\), and construct another \( r \)-independent linear sum \( f \) that is not an element of \( F(f_1, f_2, \ldots, f_{q-1}) \). Then, \( f \) is \( q \)-independent.

[Definition 8] For two linear sums \( f (\triangleq k_1 t_1 \oplus k_2 t_2 \oplus \ldots \oplus k_5 t_5) \) and \( f' (\triangleq k'_1 t_1 \oplus k'_2 t_2 \oplus \ldots \oplus k'_5 t_5) \), if \( \sum_{p=1}^{5} k_p 2^{p-1} = \sum_{p=1}^{5} k'_p 2^{p-1} \), then \( f \) is called \( \equiv f' \).

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 \( y_i \) whose weight of the \( i \)-th row vector in the corresponding dependence matrix is equal to \( w \), and assign \( t_j \) to each input \( x_j \), \((1 \leq j \leq w = w)\).

(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 \( y_{i_1}, y_{i_2}, \ldots, y_{i_v} \), which depend on \( x_j \). Next, for each output \( y_{i_v}(1 \leq v \leq c)\), find all inputs to which linear sums have already been assigned, and construct a set of such linear sums, \( L_{i_v} \).

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

\[
S^I \triangleq F(L_{i_1}) \cup F(L_{i_2}) \cup \ldots \cup F(L_{i_v}).
\]

(A-3.3) Construct \( F(T^I) \), where \( T^I (\triangleq \{ t_{i_1}, t_{i_2}, \ldots, t_{i_v} \}) \) is a linear combination of \( t_{i_1}, t_{i_2}, \ldots, t_{i_v} \). If \( S^I \subset F(T^I) \), 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 \( S^I \) to \( x_j \).

(A-3.3.2) Assign \( t_{i_v} \) 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 \( D_L \) 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 \((w + 1)\)-th column in the fifth row is 0.

(1-2) The \((w + 1)\)-th column is \((1, 1, 1, 0, \ldots, 0, 0)^T\), where \( w^T \) represents the transpose of vector \( w \).

\[
\begin{array}{cccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0
\end{array}
\]

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

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

We consider application of Akers’ algorithm to \( D_L \) (see Figure 1). Assume that the output \( y_5 \) is selected and linear sums \( f_1, f_2, \ldots, f_w \) are assigned to the inputs \( x_1, x_2, x_3, x_4 \), respectively (see the procedure (A-2), where the set \( F^w (\triangleq \{ f_1, f_2, \ldots, f_w \}) \) 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 \( L_{w+1} \) is constructed \((1 \leq i \leq \alpha)\); note that \( L_{w+1} \) is \( i \)-independent, where \( i \triangleq |L_{w+1}| \). And then \( S_{w+1} = (F(L_{w+1}) \cup F(L_{w+2}) \cup \ldots \cup F(L_{w+\alpha})) \) is constructed in the procedure (A-3.2). Using \( F^w, L_{w+1} \) and \( S_{w+1} \), the following four lemmas hold (for the simplicity, the superscript \( w \) is removed from \( L_{w+1} \) and \( S_{w+1} \) in the discussion below).

[Lemma 3] For a given linear sum set \( \{ f_{j_1}, f_{j_2}, \ldots, f_{j_v} \} \subset F^w \), if \( f_{j_1} \oplus f_{j_2} \oplus \ldots \oplus f_{j_v} \) is an element of \( F(L_i) \), then \( q \leq i \) and \( \{ f_{j_1}, f_{j_2}, \ldots, f_{j_v} \} \subset L_i \).

The proof of Lemma 3 is trivial.

[Lemma 4] There exists such a linear sum \( f \subset F^w \), such that \( f \) is a linear combination of \( q \) linear sums \( f_{j_1}, f_{j_2}, \ldots, f_{j_q} \) \((q \leq \alpha)\), where \( f_{j_1}, f_{j_2}, \ldots, f_{j_q} \in F^w \).

[Proof] Since \( L_i \subset F^w \), there exists a linear sum \( f_{j_1} \in L_i \). If we select such an \( f_{j_1} \), for each \( i \), and create the set of the selected linear sums, then it has at most \( \alpha \) elements. Let \( \{ f_{j_1}, f_{j_2}, \ldots, f_{j_q} \} \subset F^w \), \( 1 \leq q \leq \alpha \) be such set. If a linear sum \( f_{j_1} \oplus f_{j_2} \oplus \ldots \oplus f_{j_q} \) is an element in \( F(L_i) \) for \( 3^{q+1} \), then \( \{ f_{j_1}, f_{j_2}, \ldots, f_{j_q} \} \subset L_i \) from Lemma 3. This is contradiction, because at least one ele-
Let \( f_1 \oplus f_2 \oplus \cdots \oplus f_q \) be a linear sum in \( S \). Therefore, \( f_1 \oplus f_2 \oplus \cdots \oplus f_q \) is an element of \( S \). Consequently, the linear sum is an element in \( F(\omega) - S \). Thus there exists the linear sum of \( f \).

[Lemma 5] Let \( f_1 \oplus f_2 \oplus \cdots \oplus f_q \) be a linear sum in \( S \). Then, \( f_2 \) is an element in \( S \).

(Proof) Assume that \( f_2 \notin S \). Since \( f_1 \oplus f_2 \oplus \cdots \oplus f_q \) is an element in \( S \), it follows that \( f_2 \) is an element in \( S \). 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).

\begin{align*}
& (P0) \text{ If } f_1 \in H_f, \text{ then the } j_{th} \text{ column of the } i_{th} \text{ row in } D_i^C \text{ is 0.} \\
& (P1) \text{ If } f_1 \notin H_f, \text{ then the } j_{th} \text{ column of the } i_{th} \text{ row in } D_i^C \text{ is 1.}
\end{align*}

Figure 2 The value of the \( j_{th} \) column of the \( i_{th} \) row in the case that \( f_1 \in H_f \) or \( f_1 \notin H_f \).

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

[Theorem 1] (see Figure 3) Assume that a linear sum \( f \) in Lemma 4 is equal to \( f_1 \), then the \( j_{th} \) column of the \( i_{th} \) row in \( D_i^C \) is 0 for \( i (1 \leq i \leq \alpha) \).

(Proof) From the definition of \( H_f \), \( H_f = \{ f_1 \} \). Therefore, \( F_i = \{ f_1 \} \) for \( i \) from Lemma 5. Thus, Theorem 1 holds from (P0) of Lemma 6.

[Definition 9] For two distinct linear sums \( f \) and \( f' \), \( f \) is a bitwise minimum in \( S \) if for any \( \alpha \) such that \( \exists \alpha' \in \mathbb{Z} \) and \( \alpha' > \alpha \), \( f \) is a bitwise minimum in \( S \).

[Definition 10] For a given linear sum set \( L \subset F(\omega) \), 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 \( L \).

(Proof) Assume that \( f \) is a bitwise minimum in \( L \). Then, \( f \) is a bitwise minimum in \( S \).

[Theorem 4] Assume that a linear sum \( f \) in Lemma 4 is a linear combination of at least two linear sums, that is \( f = f_1 \oplus f_2 \oplus \cdots \oplus f_q \) (\( 2 \leq q \leq \alpha \)), and \( f \) is a bitwise minimum in \( S \). Then, the followings hold.

(T1) For each \( v (1 \leq v \leq \alpha) \), there exists such a row \( R_v \) corresponding to an output \( y_v \), that the \( j_{th} \) column of the \( i_{th} \) row is 0, and the other columns among the \( j_{th} \) st, the \( j_{2nd} \), the \( j_{3rd} \) th columns of the row are 1s.

(T2) Each of \( v (\alpha - q) \) rows obtained by removing \( R_v \), \( R_{v+1} \), \( \cdots \), \( R_{\alpha} \) from a upper rows has at least one 0 among the \( j_{th} \) st, the \( j_{2nd} \), the \( j_{3rd} \) th columns.

(Proof) From the definition of \( H_f \), \( H_f = \{ f_1 \} \). Therefore, \( F_i = \{ f_1 \} \) for \( i \) from Lemma 5. Thus, Theorem 1 holds from (P0) of Lemma 6.

\begin{align*}
& (a_1', a_2', \cdots, a_{\alpha'}', \cdots, a_q') \neq (1, 1, \cdots, 1) \\
& (y_1, y_2, \cdots, y_{\alpha'}) \neq \{0, 0, \cdots, 0, 1, 0, \cdots, 0\}
\end{align*}

Figure 4 The value of the \( j_{th} \) column of the \( i_{th} \) row in the case that \( f = f_1 \oplus f_2 \oplus \cdots \oplus f_{\alpha} \).
is 1 for $v'$ (1 $\leq v' \leq q$ ; $v' \neq v$). If we assume that $f_{j_n}$ is also an element in $L_{i_n}$, then $f \in F(L_{i_n})$. This is contradiction. Therefore, $f_{j_n} \not\in L_{i_n}$. Thus, the $j_n$th column of the $i_n$th row is 0 from (P0) of Lemma 6. A proof of (T2) is as follows: If we assume that the $j_n$th column of a row $R_i$ is 1 for $v'$ (1 $\leq v' \leq q$), then \{ $f_{j_1}, f_{j_2}, \ldots, f_{j_n}$ \} $\subseteq L_i$, consequently $f \not\in F(L_i)$. This is contradiction.

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

\[ \begin{array}{c|cc|c|c} \hline f_{j_1} & f_{j_2} & f_{j_3} & f_{j_4} \\ \hline y_1 & 0 & 1 & 1 & 1 \\ y_2 & 1 & 0 & 1 & 1 \\ y_3 & 1 & 1 & 0 & 1 \\ y_4 & 1 & 1 & 1 & 0 \\ y_5 & 1 & 1 & 1 & 1 \\ \hline \end{array} \]

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

\[ \begin{array}{c|cc|c|c} \hline f_{j_1} & f_{j_2} & f_{j_3} & f_{j_4} \\ \hline y_1 & 0 & 1 & 1 & 1 \\ y_2 & 1 & 0 & 1 & 1 \\ y_3 & 1 & 1 & 0 & 1 \\ y_4 & 1 & 1 & 1 & 0 \\ y_5 & 1 & 1 & 1 & 1 \\ \hline \end{array} \]

(b) $\alpha = 3, f = f_{j_1} \oplus f_{j_2} \oplus f_{j_3}, (a_{j_1}^3, a_{j_2}^3, a_{j_3}^3) \neq (1, 1)$

\[ \begin{array}{c|cc|c|c} \hline f_{j_1} & f_{j_2} & f_{j_3} & f_{j_4} \\ \hline y_1 & 0 & 1 & 1 & 1 \\ y_2 & 1 & 0 & 1 & 1 \\ y_3 & 1 & 1 & 0 & 1 \\ y_4 & 1 & 1 & 1 & 0 \\ y_5 & 1 & 1 & 1 & 1 \\ \hline \end{array} \]

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

\[ \begin{array}{c|cc|c|c} \hline f_{j_1} & f_{j_2} & f_{j_3} \\ \hline y_1 & 0 & 1 & 1 \\ y_2 & 1 & 0 & 1 \\ y_3 & 1 & 1 & 0 \\ y_4 & 1 & 1 & 1 \\ y_5 & 1 & 1 & 1 \\ \hline \end{array} \]

(d) $\alpha = 4, f = f_{j_1} \oplus f_{j_2} \oplus f_{j_3}, (a_{j_1}^4, a_{j_2}^4, a_{j_3}^4) \neq (1, 1), (a_{j_1}^4, a_{j_2}^4) \neq (1, 1)$

\[ \begin{array}{c|cc|c|c} \hline f_{j_1} & f_{j_2} & f_{j_3} \\ \hline y_1 & 0 & 1 & 1 \\ y_2 & 1 & 0 & 1 \\ y_3 & 1 & 1 & 0 \\ y_4 & 1 & 1 & 1 \\ y_5 & 1 & 1 & 1 \\ \hline \end{array} \]

(e) $\alpha = 4, f = f_{j_1} \oplus f_{j_2} \oplus f_{j_3} \oplus f_{j_4}, (a_{j_1}^4, a_{j_2}^4, a_{j_3}^4, a_{j_4}^4) \neq (1, 1, 1)$

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 $D^1_n$, 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 $D^1_n$ and $D^2_n$ with arbitrary number of columns, $D^1_n$ and $D^2_n$ are said to be equivalent iff the followings hold:

1. The number of rows ($m$) in $D^1_n$ is equal to that in $D^2_n$.
2. The row weight ($w_i$) of the $i$th row in $D^2_n$ is equal to that in $D^1_n$ for $1 \leq i \leq m$.
3. Linear sums can be assigned to the inputs in $D^1_n$ and $D^2_n$ so that the following conditions are satisfied for $1 \leq i \leq m$.

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

We next give an algorithm to construct a dependence matrix $D^2_n$ with $w$ columns which is equivalent to $D^1_n$ 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 $D^1_n$, respectively. As mentioned in the preceding section, we assume without loss of generality that $D^1_n$ is one of matrices of the types illustrated in Figures 3 and 5(a)-(f).

**Algorithm to construct $D^2_n$** According to the type of $D^1_n$, do one of the followings (1) $\sim$ (5) (let $a^3_{j_1}$ denote the value of the $j_1$th column of the $i_1$th row in $D^1_n$), and let $D^2_n$ be the matrix obtained from the resultant matrix by removing
the \((w + 1)\)st column. 
(1) In the case that \(D_{B}\) is of the type illustrated in Figure 3, change \(a_{ij}\) into 1 for every \(i (1 \leq i \leq \alpha)\).
(2) In the case that \(D_{B}\) is one of matrices of the types illustrated in Figure 5(a), (c) and (f), change \(a_{ij}\) into 1 for every \(i (1 \leq i \leq \alpha)\).
(3) In the case that \(D_{B}\) is of the type illustrated in Figure 5(b), do the following procedures (3.1) and (3.2).
(3.1) Change \(a_{ij}\) into 1 for every \(i (1 = 1, 2)\).
(3.2) (3.2.1) If \((a_{ij}^{1}, a_{ij}^{2}) = (0, 1)\) or \((0, 1)\), then change an \(a_{ij}^{1}\) whose value is 0 (\(v = 1\) or \(v = 2\)) into 1.
(3.2.2) If \((a_{ij}^{1}, a_{ij}^{2}) = (0, 0)\) and \((a_{ij}^{2}, a_{ij}^{2}) \neq (1, 0)\), then change \(a_{ij}^{2}\) into 1, and change \(f_{ij}\), which is assigned to \(x_{ij}\), into \(f_{ij} \oplus f_{ij}\).
(3.2.3) Otherwise, that is, \((a_{ij}^{1}, a_{ij}^{2}) = (0, 0)\) and \((a_{ij}^{2}, a_{ij}^{2}) = (1, 0)\), then change \(a_{ij}^{2}\) into 1, and change \(f_{ij}\), which is assigned to \(x_{ij}\), into \(f_{ij} \oplus f_{ij}\).
(4) In the case that \(D_{B}\) is of the type illustrated in Figure 5(d), do the following procedures (4.1) and (4.2).
(4.1) Change \(a_{ij}\) into 1 for every \(i (1 = 1, 2)\).
(4.2) (4.2.1) If \((a_{ij}^{1}, a_{ij}^{2}) = (0, 0)\) for \(v = 1\) or \(v = 2\), then select such a \(v\), and change \(a_{ij}^{1}\) and \(a_{ij}^{2}\) into 1, and change \(f_{ij}\), which is assigned to \(x_{ij}\), into \(f_{ij} \oplus f_{ij}\). 
(4.2.2) Otherwise, that is, \((a_{ij}^{1}, a_{ij}^{2}) = (0, 0)\) or \((1, 0)\) for \(v = 1\) and \(v = 2\), then change elements whose values are 0s among \(a_{ij}^{1}, a_{ij}^{2}\) and \(a_{ij}^{2}\) into 1s.
(5) In the case that \(D_{B}\) is of the type illustrated in Figure 5(e), do the following procedures (5.1) and (5.2).
(5.1) Select an \(a_{ij}\) into 1 for every \(i (1 = 1, 2, 3)\).
(5.2) Select an \(a_{ij}\) into 1, and change \(f_{ij}\), which is assigned to the input \(x_{ij}\), into \(f_{ij} \oplus f_{ij}\).

Before proving that \(D_{C}\) is equivalent to \(D_{B}\), we give the following lemma.

[Lemma 7] For a set \(F_{a} \subseteq F^{w}\), assume that \(F_{a} \supseteq \{f_{ij}, f_{ij}, \cdots, f_{ij}\} (w \leq 2)\). And let \(F_{a}^{e} = \{f_{ij} \ominus f_{ij}\} \cup \{f_{ij} \ominus f_{ij} \oplus f_{ij}\} (1 \leq w \leq w)\). Then \(F(F_{a}^{e}) = F(F_{a})\).

The proof is trivial. Note that since \(F_{a}\) is \(q\)-indepenent, \(F_{a}^{e}\) is also \(q\)-independent from Lemmas 1 and 7, where \(q \neq [F_{a}]\).

[Theorem 3] \(D_{C}\) is equivalent to \(D_{B}\).

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

According to the cases (1) (5) in the algorithm above, it can be proved that \(F(K_{f}) = F(K_{f})\). We show a proof only in the case (3) due to space limitation. Let \(X_{d} (1 \leq i \leq 5; k = 1, 2)\) be a set of the inputs on which the output \(y_{d}\) depends in \(D_{B}\).
(A) In the case that (3.2.1) in the algorithm is executed.
A proof for \((a_{ij}^{1}, a_{ij}^{2}) = (0, 1)\) is as follows: \(X_{d}\) does not contain \(x_{ij}\), and contains \(x_{ij}\) and \(x_{w+1}\). The algorithm changes \(a_{ij}^{1}\) into 1 and removes the \((w + 1)\)st columns. Therefore, \(X_{d} \ominus \{x_{w+1}\} = X_{d} \ominus \{x_{ij}\}\). On the other hand, \(x_{ij}\) and \(x_{ij} \oplus x_{ij}\) are assigned to \(x_{ij}\) and \(x_{w+1}\), respectively. Therefore, \(K_{f} \ominus \{x_{ij} \ominus x_{ij}\} = K_{f} \ominus \{x_{ij}\}\) and \(K_{f} \ominus \{x_{ij} \ominus x_{ij}\}\) is assigned to \(x_{ij}\). Therefore, \(F(K_{f}) = F(K_{f})\) fromLemma 7. Similarly, we can prove that \(F(K_{f}) = F(K_{f})\) for \(i = 2, 3\). For \(i = 4, 5\), \(a_{ij}^{1}\) is 0, and the algorithm does not change the first \(w\) columns. Therefore \(F(K_{f}) = F(K_{f})\).
(B) In the case that (3.2.2) in the algorithm is executed.
\(X_{d}\) does not contain \(x_{ij}\) and contains \(x_{ij}\) and \(x_{w+1}\) to which \(f_{ij}\) and \(f_{ij} \oplus f_{ij}\) are assigned in \(D_{C}\), respectively. From the algorithm \(X_{d} \ominus \{x_{ij} \ominus x_{ij}\}\) contains \(x_{ij}\) and \(x_{ij}\) to which \(f_{ij} \ominus f_{ij}\) and \(f_{ij} \oplus f_{ij}\) are assigned in \(D_{C}\), respectively, and does not contain \(x_{w+1}\). Therefore, \(K_{f} \ominus \{x_{ij} \ominus x_{ij}\}\) is of the second row, we can similarly obtain that \(K_{f} \ominus \{x_{ij} \ominus x_{ij}\}\) is of the type illustrated in Figure 5(e). For the second row, we can similarly obtain that \(K_{f} \ominus \{x_{ij} \ominus x_{ij}\}\) is of the type illustrated in Figure 5(e).

From Lemma 7, therefore, \(F(K_{f}) = F(K_{f})\).
(C) In the case that (3.2.3) in the algorithm is executed.
The algorithm 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 \(2^{m}\) 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 com-
bination of \( t_1, t_2, \ldots, t_w \) 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 \((1 \leq i \leq 5)\).

(1) Assign base column vectors \( t_1, t_2, \ldots, t_w \) to \( x_1, x_2, \ldots, \)

\( x_w \) in \( D_C \), respectively, and create a dependence matrix \( D_B^{(0)} \) 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 \((1, \ldots, 1, 0, \ldots, 0)^T\), then rearranging four rows except for the fifth one in this form and rearrange the corresponding rows of \( D_B^{(j-1)} \) in the same interchanges as \( D_C \). If the \((w + j)\)th column in \( D_C \) is the form of \((1, \ldots, 1, 0, \ldots, 0)^T\), then keep both \( D_C \) and \( D_B^{(j-1)} \) unchanged, and go to the procedure (3.2).

(3.2) Concatenate \( D_B^{(j-1)} \) 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 \( D_B^1 \) and \( f_1, f_2, \ldots, f_w \) 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 \( D_B^1 \) (the input corresponding to the \((w + 1)\)st column in \( D_B^1 \)), and create \( D_B^j \) using the algorithm in Section 4.

(3.5) \( D_B^j = D_B^{(j)} \), and increase the value of \( j \) by 1.

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

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

Since \( D_B^{(0)} = M^{(0)} \) and a base column vector \( t_j, (1 \leq j \leq w) \) is assigned to the input \( x_j \), the set of \( f_1, f_2, \ldots, f_w \) in the procedure (3.2) is \( w \)-independent. Therefore, the procedures (3.3) and (3.4) are executable and \( D_B^{(0)} \) and \( M^{(0)} \) are equivalent.

Since \( D_B^{(0)} \) 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 \( D_B^1 \) by the procedure (3.4), \( D_B^1 \) and \( M^{(1)} \) are equivalent. On the other hand, from Theorem 3, \( D_B^2 = D_B^{(1)} \) and \( D_B^3 \) are equivalent. Therefore, \( D_B^{(2)} \) and \( M^{(2)} \) are equivalent.

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

Since \( D_B^{(j+1)} \) and \( M^{(j+1)} \) are equivalent, from the condition (E3) in the definition 11, \( K_j^2 \) in \( D_B^j \) of the procedure (3.2) is \( w_3 \)-independent \((w_3 = w)\). Therefore, the set of \( f_1, f_2, \ldots, f_w \) in the procedure (3.2) is \( w \)-independent. Therefore, the procedures (3.3) and (3.4) are executable.

Since \( D_B^{(j-1)} \) 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 \( D_B^1 \) by the procedure (3.4), \( D_B^1 \) and \( M^{(j)} \) are equivalent. On the other hand, from Theorem 3, \( D_B^2 = D_B^{(j)} \) and \( D_B^3 \) are equivalent. Therefore, \( D_B^{(j)} \) and \( M^{(j)} \) are equivalent.

(3) From (1) and (2), by induction, \( D_B^{(n-w)} \) 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 \( 2^n \) test patterns.

6 Conclusion

In this paper, we showed that every CUT with five outputs has an MLTS with \( 2^n \) 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


