

An Optimal Time Expansion Model Based on Combinational ATPG for RT Level Circuits

Tomoo Inoue[†], Toshinori Hosokawa[‡], Takahiro Mihara[†], and Hideo Fujiwara[†]

[†] Graduate School of Information Science
Nara Institute of Science and Technology
Takayama, Ikoma, Nara 630-0101, Japan
{inoue, fujiwara}@is.aist-nara.ac.jp

[‡] Corporate Semiconductor Development Division
Matsushita Electric Industrial, Co., Ltd.
Yagumo-nakamachi, Moriguchi, Osaka 570-8501, Japan
hosokawa@vdr1.src.mei.co.jp

Abstract

We present an approach to test generation using time expansion models. The tests for acyclic sequential circuits can be generated by applying combinational ATPG to our time expansion models. We made experiments on application to partial scan designed register-transfer circuits. The results show that our approach can reduce hardware overhead and test length compared with full scan while preserving almost 100% fault efficiency.

1. Introduction

Test generation for sequential circuits is generally considered to be a hard problem. For such sequential circuits, design for testability (DFT) is an important approach to reducing the test generation complexity [1, 2]. On the other hand, for combinational circuits, efficient test generation algorithms were proposed, and hence we can obtain the complete fault efficiency¹ even if the circuit size is large. Therefore, it is significant to apply DFT to a sequential circuit so that the resultant circuit can be test-generated using a combinational ATPG (automatic test pattern generator).

Full scan design referring to chaining all of memory elements or flip-flops (FFs) into a shift register is such a traditional DFT technique. In the full scan design, the portion of the circuit excluding the scan path, which is called the kernel, is a combinational circuit, and consequently a combinational ATPG can be used. However, the full scan design requires large hardware overhead and large test application time. Although partial scan design which makes a subset of FFs scannable can avoid such penalties, the kernel circuit is still a sequential one [3, 4], and hence it requires the use of sequential ATPGs in general.

Recently, a class of sequential circuits for which complete test sets can be obtained by means of a combinational

ATPG is identified [5]–[7]. As such a class, Gupta *et al.* proposed *balanced structure* [5], Fujiwara *et al.* presented *internally balanced structure* [7] which includes a class of balanced sequential circuits properly. For any sequential circuit, by selecting a sufficient set of scan FFs so that the kernel circuit is balanced or internally balanced, it can be test-generated with a combinational ATPG in spite of partial scan.

On the other hand, with the advance of the technology of logic synthesis, the circuit specifications described by LSI designers have changed from gate level to register-transfer level (RTL), and several RTL DFT methods [8]–[14] have been proposed accordingly. RTL DFT has an advantage over gate level DFT in that the area and delay overhead can be absorbed during logic synthesis. RTL DFT, however, must ensure high testability for the resulting gate level circuits before synthesis, otherwise additional gate level DFT to supply the lack of testability may cause the degradation of performance and/or area, and consequently *re-synthesis* will be required. The structure-based partial scan approaches as mentioned above, i.e., based on balanced structures, are applicable to RTL DFT because scan registers can be determined before logic synthesis and it can ensure the complete fault efficiency of synthesized logic circuits. Gupta and Breuer [14] presented a class of *switched balanced structures (SB-structures)* which is larger than that of balanced structures, and proposed an RTL partial scan design method based on SB-structures.

A class of acyclic sequential circuits is also considerable because a test sequence for an irredundant single stuck-at fault in an acyclic structure can be generated by using a combinational ATPG that can deal with multiple stuck-at faults [15, 16]. Moreover, since a class of acyclic sequential circuits contains SB-structures properly, the scan overhead for acyclic structures must be lower than that for SB-structures. Gupta and Breuer [16] proposed a *test generation model* to compact a test set for an acyclic sequential circuit with partial scan.

In this paper, we introduce a *time expansion model*

¹ Fault efficiency is defined as the ratio of the number of faults identified as either detectable or redundant to the total number of faults.

(TEM) for an acyclic sequential circuit. Our TEM is a theoretical circuit model for test generation and a generalized one of Gupta's test generation models [16]. The test sequences for all irredundant faults in an acyclic sequential circuit can be generated by applying combinational ATPG capable of dealing with multiple faults to its TEM. In general, the number of TEMs for an acyclic sequential circuit is more than one, and the difference of TEMs affects the cost of test generation and test application. Hence, we propose a heuristic algorithm for finding an *optimal TEM* which minimizes the test generation cost.

The paper is organized as follows. Section 2 defines a time expansion model (TEM) of an acyclic sequential circuit and provides a theorem concerned with the testability of TEMs. Section 3 presents a RTL DFT method based on test generation using TEMs and proposes a heuristic algorithm for finding an optimal TEM which minimizes the test generation cost. Section 4 provides experimental results. The experimental results show that the proposed algorithm can find optimal TEMs efficiently and that test generation using optimal TEMs is effective in reduction of hardware overhead and test application time compared with full scan approach while preserving almost complete fault efficiency.

2. Test Generation for Acyclic Sequential Circuits

Here we present *time expansion models* for acyclic sequential circuits. The problem of test generation for an acyclic sequential circuit can be reduced to that for its time expansion model.

2.1. Topology Graph

A sequential circuit can be considered to consist of several combinational logic blocks (logic block, for short) which are connected with one another directly or via registers (or flip-flops). The structure of a sequential circuit can be represented by a *topology graph* defined as follows.

Definition 1 (Topology graph): A *topology graph* is a directed graph $G = (V, A, r)$, where a vertex $v \in V$ denotes a combinational logic block which contains primary inputs/outputs and logic gates, and an arc $(u, v) \in A$ denotes a connection or a bus from u to v . Each arc has a label $r: A \rightarrow Z^+$ (Z^+ denotes a set of non-negative integers), and $r(u, v)$ represents the number of registers on a connection (u, v) . \square

Example 1: Consider a sequential circuit S_1 illustrated in Fig. 1. In this figure, $1, 2, \dots, 8$ are logic blocks and a, b, \dots, k are registers. The topology graph of this circuit S_1 is shown in Fig. 2. \square

Definition 2 (Time expansion graph (TEG)): Let S be an acyclic sequential circuit and let $G = (V, A, r)$ be the topol-

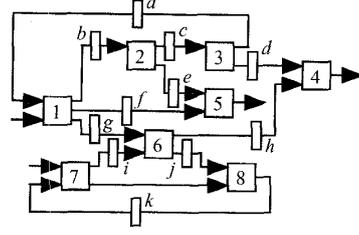


Figure 1. Sequential circuit S_1 .

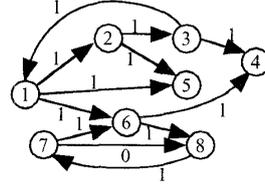


Figure 2. Topology graph of S_1 : G_1 .

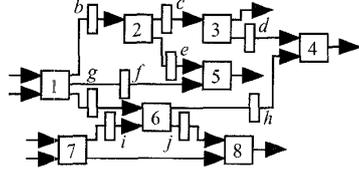


Figure 3. Acyclic sequential circuit S_2 .

ogy graph of S . Let $E = (V_E, A_E, t, l)$ be a directed graph, where V_E is a set of vertices, A_E is a set of arcs, t is a mapping from V_E to a set of integers, and l is a mapping from V_E to the set of vertices V in G . If graph E satisfies the following four conditions, graph E is said to be a *time expansion graph (TEG)* of G .

C1(Logic block preservation) The mapping l is a surjective, i.e.,

$$\forall v \in V, \exists u \in V_E \text{ s.t. } v = l(u).$$

C2(Input preservation) Let u be a vertex in E . For any direct predecessor $v (\in \text{pre}(l(u)))$ of u in G , there exists a vertex u' in E such that $l(u') = v$ and $u' \in \text{pre}(u)$. Here, $\text{pre}(v)$ denotes the set of direct predecessors of v .

C3(Time consistency) For any arc $(u, v) (\in A_E)$, there exists an arc $(l(u), l(v))$ such that $t(v) - t(u) = r(l(u), l(v))$.

C4(Time uniqueness) For any vertices $u, v (\in V_E)$, if $t(u) = t(v)$ and if $l(u) = l(v)$, then the vertices u and v are identical, i.e., $u = v$. \square

Example 2: Fig. 4 shows the topology graph G of an

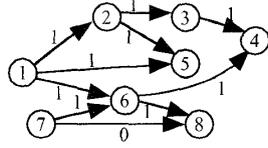


Figure 4. Topology graph of S_2 : G .

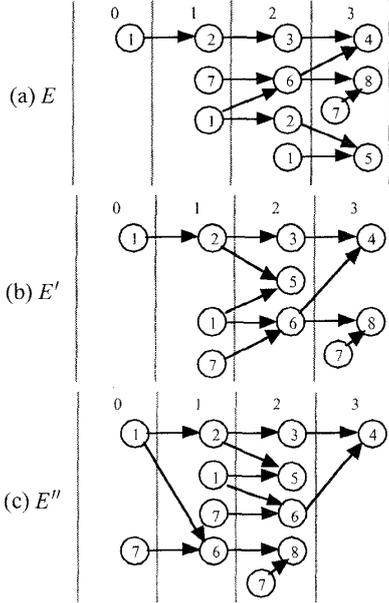


Figure 5. Example of TEMs of G .

acyclic sequential circuit S_2 in Fig. 3. Fig. 5 shows TEGs of G . In Fig. 5, the number denoted in a vertex u is the label of the corresponding one $l(u)$ in G , and the number located at the top of each column denotes the value of the labels $t(u)$ of the vertices u in the column. Each of graphs E, E', E'' satisfies all the conditions in Def. 2. \square

Note that as shown in the above example, the TEG of a topology graph is not unique in general.

Definition 3 (Time expansion model (TEM)): Let S be an acyclic sequential circuit, let $G = (V, A, r)$ be the topology graph of S , and let $E = (V_E, A_E, t, l)$ be a TEG of G . The combinational circuit $C_E(S)$ obtained by the following procedure is said to be the *time expansion model (TEM)* of S based on E .

- (1) For each vertex $u \in V_E$, let logic block $l(u) (\in V)$ be the logic block corresponding to u .
- (2) For each arc $(u, v) \in A_E$, connect the output of u to the input of v with a bus in the same way as $(l(u), l(v)) (\in A)$. Note that the connection corresponding to (u, v) has no register even if the connection corresponding to

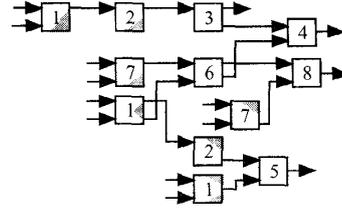


Figure 6. TEM of S_2 based on E : $C_E(S_2)$.

$(l(u), l(v))$ has a register (i.e., $r(l(u), l(v)) > 0$).

- (3) For a line or a logic gate in each logic block obtained by Steps (1) and (2), if it is not reachable to any input of other logic blocks, then it is removed. \square

Example 3: Fig. 6 shows the TEM of a sequential circuit S_2 (Fig. 3) based on TEG E (Fig. 5(a)). In this figure, a highlighted part in a logic block represents a portion of the lines and gates removed by Step (3) in Def. 3. \square

2.2. Test Generation with TEM

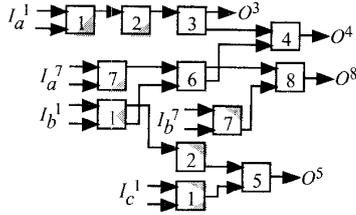
Here we consider the relationship between input/output sequences of an acyclic sequential circuit and input/output patterns of its TEM. Let S be an acyclic sequential circuit, and let $G = (V, A, r)$ be the topology graph of S . Let $E = (V_E, A_E, t, l)$ be a TEG of G , let $C_E(S)$ be the TEM of S based on E , and let t_{\min} be the minimum of labels t in $C_E(S)$.

An input pattern for $C_E(S)$ can be transformed into an input sequence for circuit S by the following procedure τ_S .

Definition 4 (Transformation procedure τ_S): For each logic block $u \in V_E$ in $C_E(S)$, let an input pattern I_u for u be the input pattern $I_{l(u)}(t(u) - t_{\min})$ which is applied to the corresponding logic block $l(u)$ at time $t(u) - t_{\min}$. \square

Lemma 1: Let I_C be an arbitrary input pattern for TEM $C_E(S)$, and let $\tau_S(I_C)$ be the input sequence obtained from I_C by procedure τ_S for S . The output pattern O_u obtained from a logic block $u \in V_E$ by applying input pattern I_C to $C_E(S)$ is equal to the output pattern $O_{l(u)}(t(u) - t_{\min})$ obtained from the corresponding logic block $l(u)$ at time $t(u) - t_{\min}$ by applying input sequence $\tau_S(I_C)$.

(Proof.) Let $u' (\in pre(u))$ be an arbitrary logic block connected to an input of logic block u . By condition C2, in S , the logic block $l(u')$ corresponding to u' is connected to the input of $l(u)$. By procedure τ_S , an input pattern $I_{u'}$ for u' is transformed into the input pattern $I_{l(u')}(t(u') - t_{\min})$ for $l(u')$ at time $t(u') - t_{\min}$. From condition C4, we can say that the number of patterns to be applied to $l(u')$ at time $t(u') - t_{\min}$ is just one. The effect of applying $I_{l(u')}(t(u') - t_{\min})$ reaches $l(u)$ $r(l(u'), l(u))$ clocks later. Here, note that $r(l(u'), l(u))$ denotes the number of registers between logic blocks $l(u')$ and $l(u)$. From condition C3, we can say that



(a) Input and output patterns for $C_E(S_2)$.

Time		0	1	2	3
Input	LB 1	I_a^1	I_b^1	I_c^1	X
	LB 7	X	I_a^7	X	I_b^7
Output	LB 3	X	X	O^3	X
	LB 4	X	X	X	O^4
	LB 5	X	X	X	O^5
	LB 8	X	X	X	O^8

(LB: Logic block)

(b) Input and output sequences transformed for S_2 .

Figure 7. Transformation procedure τ_S .

$(t(u') - t_{\min}) + r(l(u'), l(u)) = t(u) - t_{\min}$. Thus, the lemma is proved. \square

Example 4: Consider a TEM $C_E(S_2)$ (Fig. 6) of a sequential circuit S_2 shown in Fig. 3. Suppose that as shown in Fig. 7(a), an input-pattern $I_C = (I_a^1, I_b^1, I_c^1, I_a^7, I_b^7)$ is applied to $C_E(S_2)$ and the corresponding output-pattern is $O_C = (O_3, O_4, O_5, O_8)$. According to the labels t in TEG E (Fig. 5(a)), the patterns I_C and O_C are transformed into the sequences shown in Fig. 7(b) by procedure τ_S . Here, X denotes a don't-care value. \square

Note that the length of the sequence obtained from a pattern for TEM $C_E(S)$ by procedure τ_S becomes $\max_{u \in V_E} \{t(u)\} - \min_{u \in V_E} \{t(u)\} + 1$.

Let I_S be an input sequence for acyclic sequential circuit S such that the sequence determines the output pattern $O_v(t)$ of a logic block $v \in V$ in S at time t . Here, a pattern that does not affect $O_v(t)$ in the input sequence I_S is considered as don't-care. Input sequence I_S for S can be transformed by the following procedure.

Definition 5 (Transformation procedure τ_C): (1) Select a logic block $u \in l^{-1}(v)$ from the set of logic blocks $l^{-1}(v)$ corresponding to v . (2) For every input pattern $I_{v'}(t')$ applied to each logic block v' at time t' , if $I_{v'}(t')$ affects output $O_v(t)$, then let $I_{v'}(t')$ be the input pattern $I_{u'}$ for the logic block that satisfies $u' \in l^{-1}(v')$ and $t(u') = t(u) - (t - t')$. \square

Lemma 2: Let $v \in V$ be an arbitrary logic block in acyclic sequential circuit S , and let I_S be an input sequence that is required to set the output of v to $O_v(t)$ at time t . The output

pattern O_u obtained from the logic block u corresponding to v by applying the input pattern $\tau_C(I_S)$ is equal to output pattern $O_v(t)$.

(Proof.) From condition C1, we can say that there exists at least one vertex $u \in l^{-1}(v)$ that corresponding to vertex v . Let $v' \in pre(v)$ be any logic block connected to the input of logic block v . By condition C2, the logic block $u' \in l^{-1}(v')$ corresponding to v' is connected to the input of u . The input pattern that is applied to v' and that affects the output pattern $O_v(t)$ of v at time t is the pattern $I_{v'}(t - r(v', v))$ which is applied at time $t - r(v', v)$. Here, note that $r(v', v)$ is the number of registers between logic blocks v' and v . Input pattern $I_{v'}(t - r(v', v))$ is transformed into the input pattern $I_{u'}$ for u' such that $t(u') = t(u) - (t - (t - r(v', v))) = t(u) - r(v', v)$. From conditions C3 and C4, we can say that the number of patterns that satisfy $t(u') = t(u) - r(v', v)$ is just one. Thus, the lemma is proved. \square

Next, let us consider the relationship between faults in an acyclic sequential circuit and those in its TEM. Here we consider single stuck-at faults in logic blocks as those in an original sequential circuit. The stuck-at faults on lines between logic blocks and in registers can be considered to be equivalent to those on input/output lines of logic blocks.

Definition 6 (Fault in TEM): Let S be an acyclic sequential circuit. Let $G = (V, A, r)$ be the topology graph of S , let $E = (V_E, A_E, t, l)$ be a TEG of G , and let $C_E(S)$ be the TEM of S based on E . Let F be the set of faults in S , and let F_E be the set of faults in $C_E(S)$. Suppose a fault $f \in F$ in a logic block u in circuit S . Let $f_e \in F_E$ be the fault corresponding to fault f . Fault f_e is a multiple fault that consists of all the faults that exist on the same line in every logic block $u \in l^{-1}(v)$. That is, if the number of logic blocks u such that $l(u) = v$ is just one, then the fault f_e is a single fault, otherwise, f_e is a multiple fault. \square

Theorem 1: Let S be an acyclic sequential circuit. Let $G = (V, A, r)$ be the topology graph of S , let $E = (V_E, A_E, t, l)$ be a TEG of G , and let $C_E(S)$ be the TEM of S based on E . Let F be the set of faults in S , and let F_E be the set of faults in $C_E(S)$. Then,

- (1) There exists a test sequence for a fault $f \in F$ if and only if there exists a test pattern for the fault $f_e \in F_E$ corresponding to fault f .
- (2) A test pattern for a fault $f_e \in F_E$ can be transformed into a test sequence for the fault $f \in F$ corresponding to fault f_e .

(Proof.) Let T_f be a test sequence for fault f . Let t_{f_e} be a test pattern for fault f_e . Let S_f be a faulty circuit with f of S . Let $C_{E, f_e}(S)$ a faulty circuit with f_e of $C_E(S)$. Fault f_e is a multiple fault that consists faults in every logic block $l^{-1}(v)$ corresponding to the logic block $v \in V$ in which exists fault f . Hence, the structure of $C_{E, f_e}(S)$ is the same as

that of the TEM $C_E(S_f)$ of S_f based on TEG E . Therefore, by Lemma 2, test sequence T_f obtained by procedure τ_c can detect fault f_e . Further, by Lemma 1, test pattern t_f obtained by procedure τ_S can detect fault f . Thus, the theorem is proved. \square

3. RT Level Partial Scan Design

As mentioned in the previous section, an acyclic sequential circuit can be tested with its TEM by using a combinational test generation algorithm (provided that it is capable of multiple stuck-at faults). Test generation with TEMs can be applied to general sequential circuits by cutting all the loops with scan FFs prior to the test generation. The feedback loops to be cut can be determined at the RT level (or before logic synthesis) since the cut loops depend only on the circuit structure². The RT level scan design method is as follows.

- (P1) **Scan register selection.** Determine a set of scan registers that make a given RT level circuit S^R acyclic. It can be done with the minimum hardware overhead (or the minimum number of scan registers) by applying the minimum feedback vertex set (MFVS) algorithm [3,4]. Let S_{DFT}^R be a scan-designed circuit S^R , and let S_a^R be an acyclic kernel of S_{DFT}^R .
- (P2) **TEM construction.** Construct the topology graph G of S_a^R . The topology graph of an RT level circuit can be obtained by substituting ‘a functional module’ such as adders for ‘a logic gate’ in Def. 1. Create a TEG E of G . Construct the TEM of S_a^R based on E .
- (P3) **Logic synthesis.** Obtain a gate level circuit S_{DFT} corresponding to the scan-designed RT level circuit S_{DFT}^R by logic synthesis. Further, obtain a gate level circuit $C_E(S_a)$ corresponding to acyclic kernel $C_E(S_a^R)$.
- (P4) **Test generation.** Define the fault list F for acyclic kernel S_a . Based on Def. 6, define the fault list F_E for $C_E(S_a)$, corresponding to F . Generate a test set T_c for F_E with a combinational test generator capable of dealing with multiple faults.
- (P5) **Test sequence transformation.** Transform test set T_c into a test sequence T_a for acyclic kernel S_a by procedure τ_S . Finally, transform T_a into a test sequence T_S applicable to S_{DFT} via scan.

Note that in the second procedure (P2), the time required to obtain a gate level TEM $C_E(S_a)$ is small because all the combinational modules in the TEM are the same as those in its original RTL circuit.

3.1. Testing Cost Minimization Problem

In general, the number of TEMs for an acyclic sequential circuit is more than one. The difference of TEMs is consid-

²Provided that memory elements such as flip-flops are neither removed nor retimed during logic synthesis.

ered to affect its test generation time and the generated test sequence. Hence, the TEM constructed at Step (2) in the above-mentioned procedure is important to reduce the cost of testing. The relationship between a TEM and the testing cost can be considered as follows.

- (1) **TEM size.** As the size of a TEM increases, the test generation time for it also increases.
- (2) **Multiple faults.** Recall that there exist multiple faults in a TEM. The number of multiple faults and the number of faults that compose a multiple fault affects the cost of test generation.
- (3) **Number of PPIOs.** In a TEM, the primary inputs and outputs corresponding to scan registers are referred to as pseudo-primary inputs and outputs (PPIOs). Patterns for PPIOs are transformed into scan sequences. Note that an input/output pattern for a logic block u whose label is $t(u)$ are transformed into those at time $t(u)$. Hence, as the number of labels t that are affixed to logic blocks having PPIOs increases, the test application time also increases.

Example 5: A sequential circuit S_1 in Fig. 1 becomes acyclic by replacing registers a and k with scan registers. Suppose that its acyclic kernel and the topology graph are S_2 and G shown in Figs. 3 and 4, respectively. Note that the vertices labeled by 1,3,7 and 8 represent logic blocks which have PPIOs. Then, consider three TEGs E , E' and E'' of G illustrated in Fig. 5. The total number of vertices, the maximum number of faults composing a multiple fault and the number of vertices having PPIOs of E , E' and E'' are (12,3,4),(10,2,4) and (12,3,3), respectively. \square

Here, let us consider the first problem (1).

3.2. Problem Formulation

Let $G = (V, A, r)$ be the topology graph of an acyclic sequential circuit S_a . Let $E = (V_E, A_E, t, l)$ be a TEG of G , and let $C_E(S_a)$ be the TEM of S_a based on E . Let $w(v)$ ($w: V \rightarrow Z^+$) be the number of gates in logic block $v \in V$ (or the estimate of the logic size for an RT level module v). The logic size of TEM $C_E(S_a)$, $f(C_E(S_a))$, is expressed as

$$f(C_E(S_a)) = \sum_{u \in V_E} w(l(u)).$$

Then, the following problem is formulated using the above equation.

Minimum TEM Problem

Given: The topology graph $G = (V, E, r)$ of an acyclic sequential circuit S_a , the number $w(v)$ of gates in each logic block $v \in V$.

Solution: An optimal TEM E_{opt} which minimizes function $f(C_E(S_a))$.

3.2.1. Algorithm and Complexity

Recall that for a topology graph G , multiple TEGs E can be derived. For any vertex v in E , a subgraph such that vertex v is a sink vertex (i.e., its outdegree is zero) is, however, uniquely determined by conditions C2, C3 and C4. Accordingly, an algorithm for constructing a TEG E is as follows.

Algorithm: ConstructTEM

1. Let $V_0 (\subseteq V)$ be a set of vertices whose outdegree is 0 in topology graph G . For each vertex $v \in V_0$, determine the corresponding vertex $u \in V_E$ such that $l(u) = v$, and let U be the set of those vertices u .
2. For each vertex $u \in U$, define $t(u) = t$ where t is an arbitrary integer, and construct a subgraph whose sink vertex is u under conditions C2, C3 and C4. Let E be the set of those graphs.
3. Let $E := \phi$.
4. (1) Select a subgraph $E_i = (V_i, E_i, t_i, l_i) \in E$.
(2) For each vertex $v \in V_i$, let $t_i(v) := \text{retime}(t_i(v))$, where $\text{retime}()$ is a function that returns an integer (describe below).
(3) Merge E_i with E under the condition C4, and substitute the resultant graph for E .
(4) Let $E := E - \{E_i\}$. If $E \neq \phi$, then go to 1.
5. The resulting graph E becomes a TEG for G .

At Step 4.(2), the function $\text{retime}()$ modifies the label $t()$ of graph E_i , and accordingly the TEG E obtained by the above algorithm will change. Consequently, the size $f(C_E(S_a))$ of TEG $C_E(S_a)$ will also change.

For each subgraph, $E_i = (V_i, A_i, t_i, l_i) \in E$, which is obtained at Step 2, let

$$d_i = \max_{u \in V_i} \{t_i(u)\} - \min_{u \in V_i} \{t_i(u)\}.$$

The range of values calculated by function $\text{retime}()$ is expressed as $D = \sum_{i=1}^m d_i$ where $m = |E|$. Hence, the range of the label of each vertex in E_i is expressed as $s_i = D - d_i + 1$. Therefore, the total number of feasible solutions, i.e., the total number of TEMs that can be calculated to find an optimal TEM is expressed as $\prod_{i=1}^m s_i = O(D^m)$. In particular, when the length of the test sequence obtained by procedure τ_S is minimized, it is expressed as $d_{\max} = \max\{d_i\}$, and hence, the number of TEMs that should be considered becomes $O(d_{\max}^m)$.

3.2.2. Heuristic Algorithm

Here we present a heuristic algorithm for finding an optimal TEG E_{opt} which minimizes function $f(C_E(S_a))$. In order to find an optimal solution efficiently, the order of selecting a subgraph E_i at Step 4.(1) and the value determined by function $\text{retime}()$ at Step 4.(2) are important. Thus, the followings are substituted for Steps 4.(1) and 4.(2).

Algorithm: ConstructMinTEM

- 4.(1') Select a subgraph E_i such that d_i is the maximum.

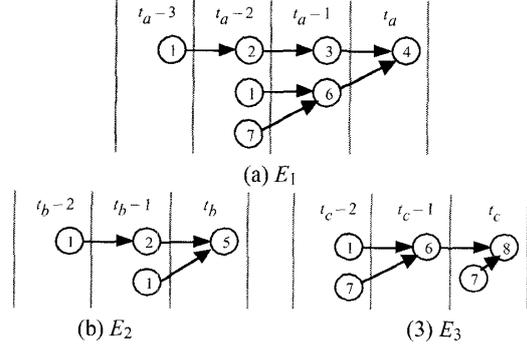


Figure 8. Subgraphs obtained at Step 2.

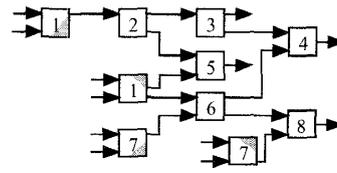


Figure 9. TEM of S_2 based on E' : $C_{E'}(S_2)$.

- 4.(2') Let function $\text{retime}()$ return a value that minimizes function $f(E)$ of graph E constructed at Step 4.(3).

Example 6: Consider a topology graph G illustrated in Fig. 4. Here, we assume that $\forall v \in V, w(v) = 1$ for the sake of simplicity. Fig. 8 shows the subgraphs obtained at Step 2 in the algorithm. The order of subgraphs selected at Step 4.(1') becomes E_1, E_2, E_3 , and the resultant TEG becomes E' , whose function is expressed as $f(C_{E'}(S_2)) = 10$, shown in Fig. 5(b). This value 10 is the minimum, and hence E' is optimal. Fig. 9 shows the corresponding TEM $C_{E'}(S_2)$. \square

4. Experimental Results

We made experiments on test generation with time expansion models (TEMs) using several register-transfer level (RTL) data path circuits. Table 1 shows the characteristics of the data path circuits. In this table, the values in parentheses present the characteristics of gate level implementations obtained by Synopsys Design Compiler. The number of gates denoted by '#gates' was evaluated using two-input NAND gate equivalent.

On these experiments, we made the circuits acyclic with the minimum number of scan FFs (Table 2), derived optimal TEMs for the acyclic kernels by the proposed heuristic algorithm *ConstructMinTEM*, and generated test patterns for the optimal TEMs using a combinational ATPG capable of dealing with multiple stuck-at faults (denoted by MC-ATPG). In order to explore the effectiveness of test generation with the optimal TEMs, we also made experi-

Table 1. Circuit characteristics.

Circuit	bitwidth	#modules (#gates)	#PIs	#POs	#regs (#FFs)
A	8	3 (2207)	2 (9)	1 (8)	11 (88)
B	16	8 (14730)	7 (82)	3 (48)	19 (384)
C	32	8 (66688)	7 (162)	3 (96)	11 (384)

bitwidth: Bit width of an input/output port.
 #modules: Number of modules (e.g. adders, multipliers).
 #PIs: Number of input ports.
 #POs: Number of output ports.
 #regs: Number of registers.

Table 2. Partial scan for cycle-breaking.

Circuit	#scan-regs	#scan-FFs	depth	#terminals
A	1	8	3	1
B	2	32	4	3
C	2	64	3	3

#scan-regs: Number of scan registers.
 #scan-FFs: Number of scan flip-flops.
 depth: sequential depth, i.e., maximum number of FFs on any path in the circuit.
 #terminals: Number of modules that has no output connected to other modules (i.e., #terminals corresponds to the node whose outdegree is zero in the topology graph).

ments on test generation with non-optimal TEMs and those on test generation using a sequential ATPG without TEMs. Non-optimal TEMs were calculated so that the TEM size became large. Moreover, assuming full scan, we generated test patterns for the combinational parts of the circuits using a combinational ATPG capable of dealing with only single stuck-at faults (denoted by C-ATPG). All experiments were performed on a SUN workstation SS20.

Table 3 shows the characteristics of optimal and non-optimal TEMs. For all the circuits, it was confirmed that the optimal TEM was calculated by *ConstructMinTEM* in less than one CPU second. Note that the TEM for the kernel of circuit A is determined uniquely since it has just one module that has no output connected to other modules (column #terminals in Table 2), and accordingly the number of subgraphs constructed in the algorithm *ConstructTEM* is just one. For circuits B and C, it was confirmed that the size (i.e., the number of gates) of the TEM obtained by *ConstructMinTEM* is the minimum.

As a test generator, Matsushita Mint [17] was used. Mint has two modes for single stuck-at faults; combinational test generation and sequential one. The MC-ATPG was implemented by modifying the combinational ATPG package in Mint.

Table 4 and 5 report the CPU time required by ATPG algorithms and the fault efficiency, respectively. From these tables we can see that the CPU time required by MC-ATPG for the optimal TEM is smaller than that for the non-optimal TEM for both circuits B and C. Moreover, for circuits B and

Table 3. Characteristics of optimal and non-optimal TEMs.

Circuit	Optimal TEM			Non-optimal TEM		
	#modules (#gates)	#PIs	#POs	#modules (#gates)	#PIs	#POs
A	5 (3530)	4 (32)	3 (24)	—	—	—
B	15 (21050)	20 (290)	5 (80)	21 (33794)	32 (452)	5 (80)
C	10 (82131)	14 (448)	5 (160)	16 (227496)	27 (864)	5 (160)

Table 4. Experimental results. CPU time [sec.] required by ATPG.

Circuit	MC-ATPG with TEM		Sequential ATPG	C-ATPG (Full scan)
	Optimal	Non-opt.		
A	36	-	248	6
B	674	881	1882	109
C	26520	204413	10405	1758

Table 5. Experimental results. Fault efficiency [%].

Circuit	MC-ATPG with TEM		Sequential ATPG	C-ATPG (Full scan)
	Optimal	Non-opt.		
A	99.98	-	94.42	100.00
B	99.87	99.85	98.26	100.00
C	99.76	99.01	99.52	99.89

C, we can see that the time required by MC-ATPG with the optimal TEM is smaller than that by sequential ATPG while the fault efficiency obtained by the MC-ATPG with optimal TEM is larger than that by sequential ATPG. Although the CPU time required by MC-ATPG with the optimal TEM is larger than that by C-ATPG for full scan for circuit A, the fault efficiency obtained by MC-ATPG using its optimal TEM is larger than that obtained by sequential ATPG and it is almost complete (100%).

Table 6 shows the number of test patterns to be applied to scan-designed circuits provided that the scan design is implemented with one scan chain. In this table, the number in parentheses denotes the number of test patterns generated by ATPGs (not including patterns required by scan). From this table we can see that the number of test patterns generated by test generation with optimal TEMs is smaller than that for full scan circuits for all the circuits. This is because the number of scan FFs in the acyclic partial scan design is smaller than that in the full scan design (Table 2), and thus the number of test patterns required for scan operation becomes smaller.

As a result, from these experiments, we can see that our partial scan and test generation approach is effective in reduction of hardware overhead and test length compared to full scan approach.

Table 6. Experimental results. Number of test patterns.

Circuit	MC-ATPG with TEM		Sequential ATPG	C-ATPG (Full scan)
	Optimal	Non-opt.		
A	875 (35)	-	134 (14)	3114 (34)
B	9536 (74)	13184 (64)	1913 (57)	28104 (72)
C	37108 (189)	203840 (392)	13324 (204)	84699 (219)

5. Conclusions

We introduced time expansion models (TEMs) for acyclic sequential circuits and proposed a register-transfer level partial scan design method based on the approach. We proved that an acyclic sequential circuit can be test-generated with its TEM using a combinational ATPG capable of dealing with multiple faults, and proposed a heuristic algorithm for finding an optimal TEM which minimizes the test generation time. Experimental results show that the proposed algorithm can find optimal TEMs efficiently and that test generation using optimal TEMs is effective in reduction of hardware overhead and test application time compared with full scan approach while the fault efficiency is almost complete.

In this paper, we formulated a TEM optimization problem provided that the test generation time increases as the size of a TEM increases. However, test generation time for TEMs may correlate with other factors such as multiple faults and the logic level. We will further investigate the characteristics of TEMs with respect to the test generation time. To consider an optimal TEM which minimizes the test length is a remaining problem.

Acknowledgments

The authors would like to thank Mr. Michiaki Muraoka, Mr. Shiro Tsuji, Mr. Mitsuyasu Ohta and Mr. Toshihiro Hiraoka of Matsushita Electric Industrial Co., Ltd., Japan, and Prof. Toshimitsu Masuzawa and Prof. Michiko Inoue of Nara Institute of Science and Technology, Japan for their fruitful comments and discussions.

References

[1] H. Fujiwara, *Logic Testing and Design for Testability*, The MIT Press, 1985.
 [2] M. Abramovici, M.A. Breuer, and A.D. Friedman, *Digital Systems Testing and Testable Design*, Computer Science Press, 1990.
 [3] K.-T. Cheng and V.D. Agrawal, "A partial scan method for sequential circuits with feedback," *IEEE Trans. Comput.*, Vol.39, No.4, pp.544-548, Apr. 1990.

[4] D.H. Lee and S.M. Reddy, "On determining scan flip-flops in partial-scan design approach," in *Proc. Int. Conf. Computer-Aided Design*, pp.322-325, Nov. 1990.
 [5] R. Gupta, R. Gupta, and M.A. Breuer, "The BALLAST methodology for structured partial scan design," *IEEE Trans. Comput.*, Vol.39, No.4, pp.538-544, Apr. 1990.
 [6] A. Balakrishnan and S.T. Chakradhar, "Sequential circuits with combinational test generation complexity," in *Proc. Int. Conf. on VLSI Design*, Jan. 1996, pp.111-117.
 [7] T. Takasaki, T. Inoue, and H. Fujiwara, "Partial scan design methods based on internally balanced structure," in *Proc. Asia & South Pacific Design Automation Conf.*, 1998, pp.211-216.
 [8] V. Chickermane, J. Lee, and J.H. Patel, "Design for testability using architectural descriptions," *Proc. IEEE Int. Test Conf.*, pp.752-761, 1992.
 [9] T.-C. Lee, N.K. Jha, and W.H. Wolf, "Behavioral synthesis of highly testable data paths under the non-scan and partial scan environments," in *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp.292-297.
 [10] S. Dey and M. Potkonjak, "Non-scan design for testability of RT-level data paths," in *Proc. Int. Conf. Computer-Aided Design*, 1994, pp.640-645.
 [11] A. Motohara, S. Takeoka, T. Hosokawa, M. Ohta, Y. Takai, M. Matsumoto, and M. Muraoka, "Design for testability using register-transfer level partial scan design," in *Proc. Asia & South Pacific Design Automation Conf.*, 1995, pp.209-215.
 [12] K. Takabatake, M. Inoue, T. Masuzawa, and H. Fujiwara, "Non-scan design for testable data paths using thru operation," in *Proc. Asia & South Pacific Design Automation Conf.*, 1997, pp.313-318.
 [13] T. Hosokawa, K. Kawaguchi, M. Ohta, and M. Muraoka, "A design for testability method using RTL partitioning," in *Proc. IEEE Asian Test Symp.*, 1996, pp.88-93.
 [14] R. Gupta and M.A. Breuer, "Partial scan design of register-transfer level circuits," *Journal of Electronic Testing: Theory and Applications*, Vol.7, pp.25-46, 1995.
 [15] A. Kunzmann and H.-J. Wunderlich, "An analytical approach to the partial scan problem," *Journal of Electronic Testing: Theory and Applications*, Vol.1, pp.163-174, 1990.
 [16] R. Gupta and M.A. Breuer, "Testability properties of acyclic structures and applications to partial scan design," in *Proc. IEEE VLSI Test Symp.*, 1992, pp.49-54.
 [17] A. Motohara, T. Hosokawa, M. Muraoka, H. Maekawa, K. Kayashima, Y. Shimeki, and S. Shin, "A State Traversal Algorithm Using a State Covariance Matrix," in *Proc. ACM/IEEE Design Automation Conf.*, 1993, pp.97-101.