

Diagnosis of Multiple Hold-Time and Setup-Time Faults in Scan Chains

James C.-M. Li, *Member, IEEE*

Abstract—This paper presents a diagnosis technique to locate hold-time (HT) faults and setup-time (ST) faults in scan chains. This technique achieves deterministic diagnosis results by applying thermometer scan input (TSI) patterns, which have only one rising or one falling transition. With TSI patterns, the diagnosis patterns can be easily generated by existing single stuck-at fault test pattern generators with few modifications. In addition to the first fault, this technique diagnoses remaining faults by applying thermometer scan input with padding (TSIP) patterns. For the benchmark circuits (up to 6.6K scan cells), experiments show that the diagnosis resolutions are no worse than 15, even in the presence of multiple faults in a scan chain.

Index Terms—Fault diagnosis, ATPG, scan chain.

1 INTRODUCTION

THIS paper presents a technique to diagnose multiple hold-time (HT) faults and setup-time (ST) faults in scan chains. The HT and ST fault models have been proposed by previous research [1] and real diagnostic cases have recently been reported [2], [3], [4]. HT faults occur when the delay from one scan cell to its neighbor is shorter than the required hold time. The mux-scan flip-flop style designs are especially susceptible to HT faults. Changing the style of scan design may avoid HT faults, but diagnosing these faults is still essential for chips that are already designed and manufactured. ST faults occur when the delay from one scan cell to its neighbor is longer than expected. ST faults can be caused by signal integrity problems or by routing congestion. Although ST faults can be eliminated by lowering the speed of scan chain shifting [5], diagnosis of ST faults is important to identify the root cause and save the test application time.

Past research in the scan chain diagnosis can be classified into hardware solutions and software solutions. In the hardware category, Schafer et al. propose adding extra routings from one scan chain to its partner scan chain [6]. Edirisooriya and Edirisooriya propose inserting XOR gates into the scan chains so that the contents of the scan cells can be flipped before shifting into the next scan cell [7]. Wu [1] and Narayanan and Das [8] propose flipping the contents of each scan cell by modifying the scan cell design. The hardware solutions require extra hardware and, what is worse, they are too late for chips that are already fabricated.

In the software category, Kundu proposes using the sequential automatic test pattern generator (ATPG) to generate diagnosis patterns [9]. Hirase et al. present an IDDQ diagnosis technique in which one IDDQ is measured every time the scan chain is shifted by one bit [10]. Kundu's and Hirase et al.'s ideas are good for single stuck-at faults only, not for multiple faults. Stanley presents a score-based diagnosis tool that does fault simulations for all latches in scan chains [11]. The score of a fault represents the degree of similarity between a circuit's expected faulty outputs and its actual outputs. Guo and Venkataranman propose a three-step diagnosis procedure [3], [12]. Huang et al. propose a probabilistic model for intermittent timing faults in scan chains [2]. Their

techniques handle multiple faults by ranking the probability of a group of candidate faults. However, none of the above techniques produces deterministic diagnosis results in the presence of multiple faults in scan chains. IBM files a patent to diagnose ST faults by changing the speed of scan chain shifting [5]. This patent performs a slow speed shifting in suspicious intervals to locate the faulty scan cell(s) by binary search. This technique, although very useful for ST faults in LSSD designs, cannot diagnose HT faults in flip-flop designs.

The proposed diagnosis technique is divided into two parts. In the first part, the fault type, the faulty chain, and the number of faults are determined. In the second part, the locations of the faults in the chains are diagnosed. The first advantage of the proposed technique is that it provides fine and deterministic diagnosis resolutions, even in the presence of multiple faults in a scan chain. This is achieved by applying *thermometer scan input* (TSI) patterns, which have only one rising transition or one falling transition. With TSI patterns, the diagnosis patterns can be generated by widely available SSF ATPG tools. This eliminates the need for a customized diagnosis pattern generator. The second important feature is that this technique diagnoses not only the first fault but also the remaining faults in a scan chain. This is accomplished by applying thermometer scan input with padding (TSIP) patterns, which are similar to TSI patterns except that the former have a specified number of pad bits. The last feature is that this technique generates dedicated diagnosis patterns, which are much shorter than regular ATPG patterns. This is especially useful when traditional diagnosis using regular ATPG patterns fails to produce fine diagnosis resolutions due to limited tester memory.

The organization of the paper is as follows: Section 2 introduces some basic terms and background knowledge. Section 3 shows how to determine the fault type and the number of faults. Sections 4 and 5 present the diagnosis techniques for the first and the remaining faults, respectively. Section 6 discusses some issues related to the technique and Section 7 summarizes this paper.

2 BACKGROUND

2.1 Scan Chain Fault Models

The scan cells are indexed in descending order, from scan input (index = L-1) to scan output (index = 0). The length of the scan chain (L) is the total number of scan cells in the chain. For a given scan cell i , cells that are indexed higher or lower than i are *upstream* or *downstream* of cell i , respectively [12]. Let j denote the scan clock cycle number and let $g(i, j)$ represent the good content of scan cell i at clock cycle j . Assuming no inversion between scan cells, the shift operation of a good chain is modeled by $g(i+1, j) = g(i, j+1)$. Let $a(i, j)$ denote the actual content of scan cell i at cycle j . For a faulty chain, the actual content of a faulty cell is different from its good content when certain *excitation conditions* are met. An ST fault in cell i is excited when cell i is expected to make a rising or falling transition in the current cycle—that is, $g(i, j-1) \neq g(i, j)$. The effect of the ST fault is that cell i remains the previous value in the current cycle—that is, $a(i, j) = g(i, j-1)$. An HT fault in cell i is excited when cell i is expected to make a rising or falling transition in the next clock cycle—that is, $g(i, j+1) \neq g(i, j)$. The effect of the HT fault is that cell i makes the transition in the current cycle—that is, $a(i, j) = g(i, j+1)$. The HT faulty cell can be regarded as a transparent buffer that bypasses the scan data without holding them [1], [12].

Causes of HT faults and ST faults can be classified into three major categories: the scan cell internal problems, the scan signal timing problems, and the clock timing problems. In the first category, defective transistors (such as leakage and incorrect implants) and highly resistive bridging defects are shown to be culprits for ST faults [13] and HT faults [4], respectively. For the

• The author is with the Electrical Engineering Institute and the Graduate Institute of Electronic Engineering, National Taiwan University, Taipei, Taiwan. E-mail: cml@cc.ee.ntu.edu.tw.

Manuscript received 7 Sept. 2004; revised 2 Mar. 2005; accepted 8 June 2005; published online 16 Sept. 2005.

For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-0289-0904.

TABLE 1
TSI Patterns of Length 2L

Type	Tail (length L)	Head (length L)
T_{S10}	11111...1	<u>1</u> 0000...0
T_{H10}	11111...1	0 <u>0</u> 000...0
T_{S01}	00000...0	0 <u>1</u> 111...1
T_{H01}	00000...0	<u>1</u> 1111...1

second category, signal integrity problems such as crosstalk and ground bounce are responsible for scan chain faults in nano-meter technologies [2]. On top of that, ST faults can be caused by setup time violation due to routing congestion, especially when the router gives priority to functional signals over scan signals. HT faults can be caused by hold-time violations due to insufficient buffers between scan flip-flops. The third category, clock skew, is the result of design errors or process defects. The mux-scan flip-flop style designs are especially susceptible to HT faults. HT faults occur if the active clock edge arrives at the faulty cell i much later than at its upstream cell $i + 1$ [1]. Some of the scan chain faults are permanent (such as clock skew) and the others are intermittent (such as crosstalk). Please see Section 6 for more details about the intermittent fault diagnosis.

2.2 Thermometer Scan Input Patterns

Thermometer scan input (TSI) patterns are scan input patterns that have only one rising transition or one falling transition. Table 1 lists TSI patterns of length $2L$, which are evenly divided into two parts: the *head* portion and the *tail* portion. (Note that the rightmost bit is shifted into the chain first.) The underlined bits, which are flipped after the fault excitation, are called *sensitive bits*. By definition, there is only one sensitive bit in a TSI pattern. There are four types of TSI patterns: S10, H10, S01, and H01. The "01" types have a group of ones followed by a group of zeros; the "10" types have a group of zeros followed by a group of ones. TSI patterns for the ST fault, subscripted "S", have sensitive bits equal to the tail portion. TSI patterns for the HT fault, subscripted "H", have sensitive bits equal to the head portion. The reason why the sensitive bits are always placed at the end of the head portion will become clear in the coming sections.

As a TSI pattern is shifting in a scan chain with a single fault, the sensitive bit is flipped as soon as it passes the faulty cell. To diagnose the single fault is to find out, as early as possible, where the sensitive bit is flipped. Things become more complicated in the presence of multiple faults. Fig. 1 shows how the actual contents of scan cells change as the T_{H10} pattern shifts in a scan chain with two HT faults. Suppose that the first and the second faults are located in cell 11 and cell 8, respectively. (The n th fault, f_n , is the faulty cell that has the n th highest index.) When the sensitive bit reaches

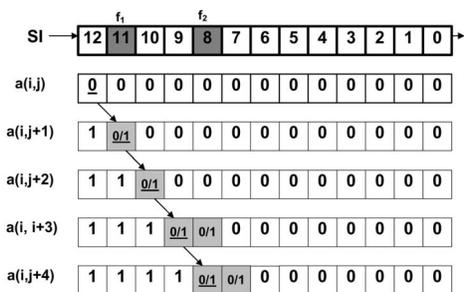


Fig. 1. Contents of scan cells as T_{H10} shifts (double faults).

TABLE 2
Two Test Patterns to Determine Fault Type and Number of Faults
(Two Faults Assumed)

Pattern	Pattern 1 (L)	Pattern 2 (L)
Scan in	11110000	00001111
Scan out of ST	11000000	00111111
Scan out of HT	11111100	00000011

cell 11 in cycle $j + 1$, it is flipped as if cell 11 is transparent. Similarly, in cycle $j + 3$, faulty cell 8 is also flipped. The actual contents now have two flipped bits: the sensitive bit and its immediate downstream bit. It can be generalized that a TSI pattern has at most n flipped bits in the presence of n faults in a scan chain.

3 DETERMINING FAULT TYPES AND NUMBER OF FAULTS

The three-pattern test proposed by [12] cannot count the number of faults in the presence of multiple faults. This paper proposes a two-pattern test (Table 2) to determine the fault type as well as the number of faults in a scan chain. Each of the patterns is of equal length L . In the first pattern, the downstream half is all zeros and the upstream half is all ones. The second pattern is the bitwise complement of the first pattern. On the tester, the scan chain is initialized by shifting the first (rightmost) bit of the first pattern for L times. Alternatively, the scan chain can be initialized by asserting the independent reset or preset signals if available. The first pattern is then scanned in, followed by an immediate scan out without any system clock. The scan outputs are recorded by the tester without an immediate pass/fail decision. The same procedure is repeated for the second pattern. If there exist f ST faults, there will be f more zeros and f more ones than expected in the first and the second scan outputs, respectively. The number of HT faults can be counted in a similar way.

The scan chains that fail either test are faulty; those chains that pass both tests are good. In the following section, the faulty chains are diagnosed one chain by one chain—that is, one scan *chain under diagnosis* (ChUD) at a time. The scan chains other than the ChUD are the scan *chains under no diagnosis* (ChUNDs). The ChUD must be faulty and ChUNDs can be either good or faulty.

4 DIAGNOSIS OF THE FIRST FAULT

Fig. 2 illustrates the flow chart to diagnose the first fault in a *circuit under diagnosis* (CUD). The combinational automatic diagnosis pattern generator (C-ADPG) generates diagnosis patterns of coarse resolution within a short time, whereas the sequential automatic diagnosis pattern generator (S-ADPG) generates diagnosis patterns of fine diagnosis resolutions at the cost of relatively long CPU time. The ADPG patterns include scan inputs (SI), primary inputs (PI), good primary outputs (PO_g), and good scan outputs (SO_g). All the actually observed primary outputs (PO_a) and scan outputs (SO_a) are logged into a file without an immediate pass/fail decision. Finally, PO_a and SO_a are matched with PO_g and SO_g offline to

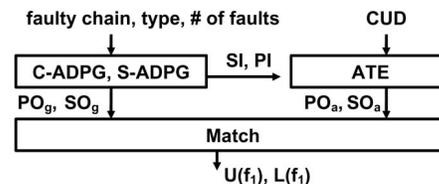


Fig. 2. Diagnosis of the first fault.

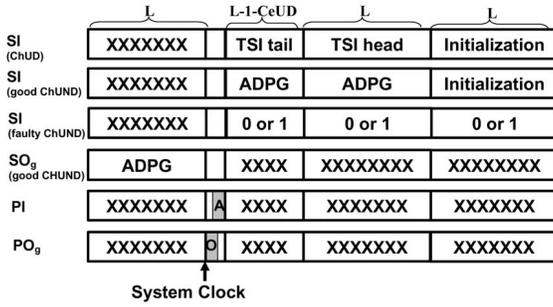


Fig. 3. C-ADPG pattern format.

obtain the diagnosis results, the upper bound $U(f_1)$, and the lower bound $L(f_1)$ of the first fault.

4.1 C-ADPG

4.1.1 Combinational Diagnosis Procedure, CDP

The CDP is applied to one single scan cell, the scan cell under diagnosis (CeUD), every time a ChUD is loaded. The CDP steps are as follows:

1. Initialize the ChUD by shifting in the first bit of the TSI pattern for L times.
2. Keep shifting in the TSI pattern until the sensitive bit reaches the CeUD.
3. Apply a primary input pattern.
4. Observe the primary outputs.
5. Pulse a system clock and shift out the scan chains. Observe the scan outputs of good ChUNDS. Mask the scan outputs of the other chains.

Fig. 3 shows the C-ADPG pattern format. The initialization pattern is the first bit of TSI patterns replicated L times. This initialization ensures that the head portion of the TSI pattern is shifted into the scan chain without being changed by faults. The head portion of the TSI pattern is then shifted in without observing any SO or PO. At this time, the sensitive bit is located at the most upstream scan cell. The ChUD is again shifted until the sensitive bit reaches the position of the CeUD. The number of shifts equals $L - 1$ minus the index of the CeUD. At this time, the sensitive bit is located at the CeUD. One primary input pattern is applied (A) and the corresponding primary outputs are observed (O). A system clock is pulsed to capture the responses into the scan chains. The scan outputs of good ChUNDS are observed; the scan outputs of the other chains are masked. To ensure only one sensitive bit at a time, all-zero patterns or all-one patterns are applied to faulty ChUNDS. On the contrary, the scan input patterns to the good ChUNDS can be specified by the C-ADPG.

4.1.2 C-ADPG

To detect a flipped sensitive bit is equivalent to detecting a stuck-at S fault in the cell that contains the sensitive bit, where S is opposite to the good value of the sensitive bit. In the case of single fault, the C-ADPG patterns can simply be generated by a regular combinational SSF ATPG tool because the contents of the ChUD are fully specified except the sensitive bit. In the presence of multiple faults, however, the actual contents of the ChUD are not fully specified because the number of flipped bits is unknown. To circumvent this problem, the TSI with unknown (TSIX) patterns is proposed. The TSIX patterns are TSI patterns that have consecutive unknown values to the immediate upstream or downstream of the sensitive bits. TSIX patterns for the ST fault are denoted as $T_{S01}X_x$ or $T_{S10}X_x$, in which x represents the number of unknown values. $T_{S01}X_x$ and $T_{S10}X_x$ patterns have x unknown values in the upstream of the sensitive bit; $T_{H10}X_x$ and $T_{H01}X_x$ patterns have x unknown values

in the downstream of the sensitive bit. TSIX patterns with x unknowns are used for a scan chain with $(x + 1)$ faults.

With TSIX patterns, the C-ADPG can now be implemented by a regular combinational SSF ATPG tool. The C-ADPG steps as follows:

1. Right shift the TSIX pattern until the sensitive bit reaches the position of the CeUD in the head portion. Initialize the scan cells of the ChUD to the head portion of the shifted TSIX pattern.
2. Initialize the scan cells in faulty ChUNDS to all ones or zeros.
3. Inject a stuck-at S fault at the sensitive bit; S is opposite to the sensitive bit.
4. Run combinational SSF ATPG to detect the fault. Allow observation only at PO and SO of good ChUNDS.
5. If ATPG is successful, cell i is C-observable.

For a given fault type and a given number x , a cell is C-observable if C-ADPG succeeds using either the $T_{01}X_x$ or the $T_{10}X_x$ pattern; otherwise, it is C-unobservable. Scan cells are C-unobservable for two possible reasons. The first is lack of propagation paths due to the constraints of TSI patterns. The second is the constraint that faulty ChUNDS are forced to all ones or all zeros. To enhance the diagnosis resolution, the C-unobservable cells must be diagnosed by the sequential diagnosis procedure (SDP).

4.2 S-ADPG

4.2.1 Sequential Diagnosis Procedure, SDP

The SDP is different from the CPD in that the former has more than one system clock. The SDP steps are the following:

1. Initialize the ChUD by shifting in the first bit of the TSI pattern L times.
2. Keep shifting in the TSI pattern until the sensitive bit reaches the CeUD.
3. Apply a primary input pattern.
4. Observe primary outputs (PO_a) and pulse a system clock.
5. Repeat Steps 3 and 4 a specified number of times.
6. Shift out the scan chains. Observe scan outputs of good ChUNDS. Mask scan outputs of the other chains.

Fig. 4 illustrates how the SDP detects a flipped sensitive bit at primary outputs. In the first time frame, the TH_{10} pattern "1100" is shifted into the chain and the sensitive bit is flipped by the HT fault in cell 2. The actual contents of scan cells become "11100." After one system clock, the fault effect of the flipped sensitive bit is captured in cell 3 of the ChUD. The fault effect is then propagated from cell 3 to a primary output by applying $PI_2 = 1$. Note that, even though the fault effect is propagated to cell 2 in the second time frame, the S-ADPG is essentially a single fault (not multiple fault) sequential ATPG because the origin of the fault is uniquely generated by cell 2 in the first time frame.

4.2.2 S-ADPG

The S-ADPG is similar to the sequential SSF ATPG, so it can be implemented by existing tools with modifications. The S-ADPG includes seven steps:

1. Right shift the TSIX pattern until the sensitive bit reaches the position of the CeUD in the head portion. Initialize the scan inputs of scan cells in the ChUD to the head portion of the shifted TSIX pattern.
2. Inject a stuck-at S fault at the scan input of the CeUD. S is opposite to the sensitive bit.
3. Initialize the scan inputs of all scan cells in faulty ChUNDS to all ones or zeros.

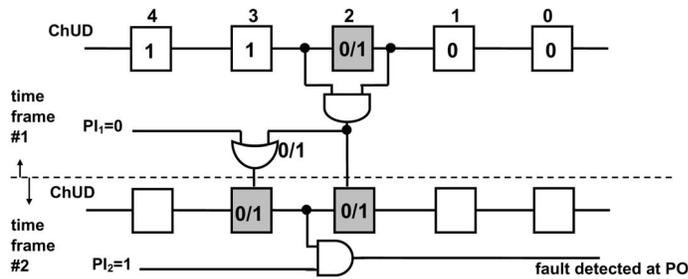


Fig. 4. Fault in cell 2 detected by SDP (good/faulty).

4. Assert the scan-enable signal (shift mode) and pulse one clock. By doing so, all the scan cells are initialized and the fault is injected into the CeUD.
5. Deassert the scan_enable (normal operation mode) and run sequential ATPG to detect the injected fault. Allow observation at the primary outputs only.
6. After the last system clock, observe scan outputs of good ChUDs. Mask the scan outputs of the other chains.
7. If S-ADPG is successful, the CeUD is S-observable.

There is one major difference between the regular sequential SSF ATPG and the S-ADPG—the time frame during which the stuck-at fault is present. The regular sequential SSF ATPG assumes that the stuck-at fault is present in every time frame; the S-ADPG assumes that the stuck-at fault is present only in the first time frame, not in the subsequent time frame. This is because the scan cells are assumed to be faulty only in shift mode, not in normal operation mode. Based on this assumption, the fault is injected at the scan input, not the output, of the CeUD. Because the scan_enable is forced to zero after Step 4, only the fault effects (not the stuck-at fault itself) remain in the circuit after the first time frame.

4.2.3 Diagnosis Resolution

A cell is *observable* if it is either C-observable or S-observable; otherwise, it is *unobservable*. The *diagnosis upper bound* of cell i , $DUB(i)$, is the index of the nearest upstream observable cell of cell i (excluding cell i itself). The *diagnosis lower bound* of cell i , $DLB(i)$, is the index of the nearest downstream observable cell of cell i (including cell i itself). The *diagnosis resolution* of a cell i , $DR(i)$, is equal to $DUB(i)$ minus $DLB(i)$. In Fig. 5, cells 4, 3, and 0 are C-observable and the others are C-unobservable. After performing S-ADPG on cell 2 and cell 1, the former becomes S-observable. Overall, cells 4, 3, 2, 0 are observable and cell 1 is unobservable.

4.3 Match

After testing the CUD on the ATE, the actually observed PO_a and SO_a are compared with the good outputs, PO_g and SO_g , offline. The *first mismatch cell* is the most upstream cell in which a mismatch occurs. The first mismatch cell is obtained from either the CDP or the SDP, whichever is more upstream. The upper bound of the first fault, $U(f_1)$, and the lower bound of the first fault, $L(f_1)$, are the DUB and the DLB of the first mismatch cell, respectively.

Scan In	4	3	2	1	0
C-observable ?	Y	Y	N	N	Y
S-observable ?	-	-	Y	N	-
overall ob. ?	Y	Y	Y	N	Y
DUB(i)	5	4	3	2	2
DLB(i)	4	3	2	0	0
DR(i)	1	1	1	2	2

Fig. 5. DUB, DLB, and DR.

4.4 Experimental Results

Table 3 shows the experimental results of the ISCAS '89 and ITC '99 benchmark circuits. A commercial tool that supports both combinational and sequential ATPG is used. The average DRs and the worst DRs are the average and the maximum $DR(i)$ values among all scan cells in the ChUD. The first row (C) shows DRs obtained from the C-ADPG only. The second row (C+S) shows DRs obtained from the C-ADPG plus the S-ADPG. Three system clocks are applied in the S-ADPG. The total number of combinational logic gates (G), the number of flip-flops (FF), and the ChUD length (L) are shown for reference. Flip-flops in the ISCAS and the ITC circuits are evenly stitched into two and 16 scan chains, respectively. The experimental data show that worst DRs are better than 15, even in the presence of 11 faults ($X = 10$). For comparison, the worst DR of a previous single HT fault diagnosis is 27 on an industry design ($G = 430K$, $FF = 22K$, $L = 410$) [12]. In addition to good diagnosis resolutions, our diagnosis pattern is much shorter than regular ATPG patterns. For the case of B19, only 409 patterns, instead of 10K, regular ATPG patterns are needed for diagnosis.

Table 4 shows DRs of one, two, and three system clocks (HT fault, $X = 10$). The CPU time is measured on a Sun Blade 2500 workstation with 8GB memory. The DRs of C-ADPG, which can be regarded as one SCK, are shown in the first column. As the number of system clocks increases, DRs improve at the cost of CPU time. The B19 S-ADPG with three system clocks is finished within one hour of CPU time, demonstrating the feasibility of the technique on large designs. The S-ADPG experiment on B19 has no abort or memory problem. (See Section 6 for the sequential ATPG details.) The number of system clocks needed in the S-ADPG is dependent on the sequential depth of the CUD. The sequential depth of a circuit represents the maximum level of flip-flops that a fault effect must propagate through before reaching observation points [14].

5 DIAGNOSIS OF REMAINING FAULTS

5.1 TSIP Patterns

The *thermometer scan input with padding* (TSIP) patterns are TSI patterns that have one or more consecutive *pad bits* immediately upstream or downstream of the sensitive bits. The value of a pad bit is the same as its sensitive bit. Table 5 shows the TSIP patterns with two pad bits (double underlined). TSIP patterns for the ST fault are denoted as $T_{S01}P_p$ or $T_{S10}P_p$, where the subscript p represents the number of pad bits. TSIP patterns for the HT fault are denoted in the same way. The pad bits are in the upstream or the downstream of the sensitive bits for the HT fault and the ST fault, respectively. Fig. 6 demonstrates how the actual contents of scan cells change as a $T_{H10}P_1$ pattern is shifting in the scan chain with two HT faults in regions R2 and R4. As soon as the sensitive bits pass R2, the pad bit is flipped while the sensitive bit remains unchanged. From this point on, the actual contents of the scan cells becomes a T_{H10} patterns without any pad bit.

TABLE 3
Diagnosis Resolutions of ST and HT Faults (Average Worst)

CUD	ADPG	ST X=0	ST X=10	HT X=0	HT X=10
s838 (G=446; FF=32; L=16)	C	3.3 / 4	9.4 / 12	3.3 / 4	10 / 12
	C+S	1.0 / 1	1.0 / 1	1.0 / 1	1.0 / 1
s5378 (G=2,779; FF=179; L=90)	C	1.8 / 6	2.5 / 9	1.8 / 6	2.2 / 7
	C+S	1.2 / 3	1.5 / 4	1.2 / 3	1.5 / 4
s9234 (G=5,597; FF=211; L=106)	C	2.2 / 5	2.8 / 8	2.2 / 5	2.9 / 9
	C+S	1.1 / 3	1.2 / 3	1.1 / 3	1.2 / 3
s15850 (G=9,772; FF=534; L=267)	C	1.7 / 4	1.7 / 4	1.7 / 4	1.7 / 4
	C+S	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2
s38584 (G=19,253; FF=1,426; L=713)	C	1.7 / 6	1.7 / 6	1.7 / 6	1.7 / 6
	C+S	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2
B19 (G=230K; FF=6.6K; L=409)	C	6.5 / 19	7.9 / 20	6.4 / 19	8 / 21
	C+S	3.2 / 9	5.4 / 15	3.4 / 11	5.6 / 15

5.2 ADPG

With the application of TSIP patterns, the diagnosis of the n th fault is transformed into the diagnosis of the first fault. For the example in Fig. 6, after the sensitive bit enters region R3, diagnosing the second fault is virtually the same as diagnosing the first fault. Specifically, at cycle $(j + 7)$, the ADPG pattern to detect the second fault in cell 6 is the same as the ADPG pattern to detect the first fault in cell 6 (as if there were no f_1). Since the ADPG patterns for the first fault are already generated in Section 4, ADPG patterns for the second fault need not be generated again. The primary inputs, the scan inputs to the ChUNDS, and the expected good outputs remain the same. The only difference is the scan input patterns to the ChUD. TSIP patterns with $n - 1$ pad bits are applied to the ChUD for detecting the n th fault.

5.3 Experimental Results

Table 6 shows the DRs of the benchmark circuits for the second, third, fifth, and 10th faults. The results show a small increase in average DR (S5378, B19), but no increase in worst DR. The experiment indicates that DRs of the 10th fault are very close to those of the first fault. This is especially true for large CUDs, of which the chain lengths are much larger than the number of faults.

6 DISCUSSIONS

6.1 Intermittent Faults

A scan chain fault is *intermittent* if it is excited with a probability lower than one [2]. The proposed technique can correctly diagnose the intermittent faults as long as the fault is excited by either the C-ADPG or S-ADPG patterns. For a scan chain of length L , there are approximately L distinct C-ADPG and S-ADPG patterns, which could be sufficient when the chain is long. The intermittent fault is not diagnosed, however, if it is not excited by any ADPG patterns. In this case, the diagnosis patterns can be applied under different test conditions (such as different voltages or different speeds) to increase the probability of exciting the intermittent

TABLE 4
DRs and CPU Time of Different System Clocks (X = 10, HT)

CUD	1SCK (C-ADPG)		2SCK		3SCK	
	DR (A/W)	CPU (Sec.)	DR (A/W)	CPU (Sec.)	DR (A/W)	CPU (Sec.)
s838	9.4 / 12	0.0	1.0 / 1	1.3	1.0 / 1	1.6
s5378	2.5 / 9	0.2	1.6 / 4	4.4	1.5 / 4	6.7
s9234	2.8 / 8	0.4	1.2 / 3	19.1	1.2 / 3	35.5
s15850	1.7 / 4	2.1	1.0 / 2	49.7	1.0 / 2	73.2
s38584	1.7 / 6	14.6	1.0 / 2	213.9	1.0 / 2	238.4
B19	8.0 / 21	54.6	6.1 / 17	2,940.0	5.6 / 15	3,265.0

faults.

6.2 Distinguish a Pair of STR and STF

One interesting question that might arise is that how a pair of STF and STR faults is distinguished from a single ST fault. The key to this question is whether there exist at least one T_{S01} observable cell and at least one T_{S10} observable cell between these two faults. Suppose that there are two faults in a chain; the first fault, f_1 , is STR and the second fault, f_2 , is STF. If there is one T_{S10} observable cell between f_1 and f_2 , a failure is observed when the sensitive bit of T_{S10} passes f_1 . If there is one T_{S01} observable cell between f_1 and f_2 , no failure is observed when the sensitive bit of T_{S01} passes f_1 . In this case, it is known that the f_1 is an STR fault, not an ST fault. On the contrary, if there is no T_{S10} observable cell or if there is no T_{S01} observable cell between f_1 and f_2 , then we cannot tell the two faults from a single ST fault. Nevertheless, the upper and lower bounds are still correctly diagnosed by our technique and the fault(s) can be distinguished by physical analysis.

6.3 Issues of Sequential Pattern Generation

Although S-ADPG is implemented by sequential ATPG, it actually requires less memory than regular sequential ATPG. Because

TABLE 5
TSIP Patterns (P = 2)

Type	Tail (length L)	Head (length L)
$T_{S10}P_2$	111...111	<u>111</u> 00...0
$T_{H10}P_2$	111...1 <u>00</u>	<u>00</u> 000...0
$T_{S01}P_2$	000...000	<u>000</u> 11...1
$T_{H01}P_2$	000...0 <u>11</u>	<u>11</u> 111...1

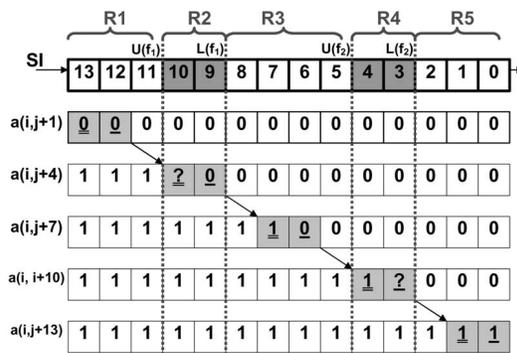


Fig. 6. Actual contents of scan cells as $T_{H10}P_1$ pattern shifts right.

TABLE 6
DRs of the N th Fault ($X = 10$, $SCK = 3$)

CUD	Fault Type	n=1	n=2	n=3	n=5	n=10
s5378	ST	1.5 / 4	1.5 / 4	1.5 / 4	1.5 / 4	1.6 / 4
	HT	1.5 / 4	1.5 / 4	1.5 / 4	1.5 / 4	1.5 / 4
s9234	ST	1.2 / 3	1.2 / 3	1.2 / 3	1.2 / 3	1.2 / 3
	HT	1.2 / 3	1.2 / 3	1.2 / 3	1.2 / 3	1.2 / 3
s15850	ST	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2
	HT	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2
s38584	ST	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2
	HT	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2	1.0 / 2
B19	ST	5.4 / 15	5.4 / 15	5.4 / 15	5.4 / 15	5.5 / 15
	HT	5.6 / 15	5.6 / 15	5.6 / 15	5.6 / 15	5.6 / 15

S-ADPG generates one pattern for a fault at a time, the memory requirement is smaller than regular sequential ATPG, which generates patterns for many faults at a time. On top of that, S-ADPG loads all chains before generating sequential patterns, so the initial state of the CUD is known. This requires far less memory than a regular sequential ATPG in which the initial state is usually unknown. Also, the number of system clocks in S-ADPG can be specified by the user, so the memory requirement can be controlled. Last, S-ADPG can be implemented by a "fast-sequential" ATPG mode [15], which is supported by modern ATPG tools to speed up the sequential pattern generation.

7 SUMMARY

This paper presents a technique that diagnoses multiple setup-time and hold-time faults in scan chains. The diagnosis consists of two parts. The first part determines the fault type, faulty chain, and the number of faults. The second part locates the faulty cell. The proposed technique has the advantage of handling scan chains with multiple faults because of the use of thermometer scan input patterns. This diagnosis technique provides deterministic results even in the presence of multiple faults. Experiments on benchmark circuits (up to 6.6K flip-flops) show that the worst diagnosis resolutions are no worse than 15.

ACKNOWLEDGMENTS

This work is supported by the National Science Council of Taiwan under Grant NSC 93-2215-E-002-030.

REFERENCES

- [1] Y. Wu, "Diagnosis of Scan Chain Failures," *Proc. Int'l Symp. Defect and Fault Tolerance in VLSI Systems*, pp. 217-222, 1998.
- [2] Y. Huang, W.-T. Cheng, S.M. Reddy, C.-J. Hsieh, and Y.-T. Hung, "Statistical Diagnosis for Intermittent Scan Chain Hold-Time Fault," *Proc. IEEE Int'l Test Conf.*, pp. 319-327, 2003.
- [3] R. Guo and S. Venkataranman, "A New Technique for Scan Chain Failure Diagnosis," *Proc. Int'l Symp. Testing and Failure Analysis*, pp. 723-732, 2002.
- [4] B. Kruseman, A. Majhi, C. Hora, S. Eichenberger, and J. Meirlevede, "Systematic Defects in Deep Sub-Micron Technologies," *Proc. IEEE Int'l Test Conf.*, pp. 290-298, 2004.
- [5] F. Motika, P. Nigh, P. Song, and H.B. Druckerman, "AC Scan Diagnostic Method," US Patent 6,516,432 B1, 4 Feb. 2003.
- [6] J. Schafer, F. Policastro, and R. McNulty, "Partner SRLs for Improved Shift Register Diagnostics," *Proc. IEEE VLSI Test Symp.*, pp. 198-201, 1992.
- [7] S. Edirisooriya and G. Edirisooriya, "Diagnosis of Scan Path Failures," *Proc. IEEE VLSI Test Symp.*, pp. 250-255, 1995.
- [8] S. Narayanan and A. Das, "An Efficient Scheme to Diagnose Scan Chains," *Proc. Int'l Test Conf.*, pp. 704-713, 1997.
- [9] S. Kundu, "Diagnosis Scan Chain Faults," *IEEE Trans. VLSI Systems*, vol. 2, no. 4, pp. 512-516, Dec. 1994.
- [10] J. Hirase, N. Shindou, and K. Akahori, "Scan Chain Diagnosis Using IDDQ Current Measurement," *Proc. Asian Test Symp.*, pp. 153-157, 1999.
- [11] K. Stanley, "High-Accuracy Flush-and-Scan Software Diagnostic," *IEEE Design and Test of Computers*, vol. 18, no. 6, pp. 56-62, Nov./Dec. 2001.
- [12] R. Guo and S. Venkataranman, "A Technique for Fault Diagnosis of Defects in Scan Chains," *Proc. IEEE Int'l Test Conf.*, pp. 268-277, 2001.
- [13] P. High et al., "Failure Analysis of Timing and IDDq Failures from the SEMATECH Test Methods Experiment," *Proc. IEEE Int'l Test Conf.*, pp. 43-52, 1998.
- [14] M.L. Bushnell and V.D. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits*, p. 226. Boston: Kluwer Academic, 2000.
- [15] Synopsys, TetraMAX ATPG User Guide, V-2003.12, Dec. 2003.