PAPER

Scan Shift Time Reduction Using Test Compaction for On-Chip Delay Measurement

Wenpo ZHANG[†a)], ***Student Member***, **Kazuteru NAMBA**[†], ***Member***, ***and*** **Hideo ITO**[†], ***Fellow***

SUMMARY In recent VLSIs, small-delay defects, which are hard to detect by traditional delay fault testing, can bring about serious issues such as short lifetime. To detect small-delay defects, on-chip delay measurement which measures the delay time of paths in the *circuit under test* (CUT) was proposed. However, this approach incurs high test cost because it uses scan design, which brings about long test application time due to scan shift operation. Our solution is a test application time reduction method for testing using the on-chip path delay measurement. The testing with on-chip path delay measurement does not require capture operations, unlike the conventional delay testing. Specifically, FFs keep the transition pattern of the test pattern pair sensitizing a *path under measurement* (PUM) (denoted as p) even after the measurement of p. The proposed method uses this characteristic. The proposed method reduces scan shift time and test data volume using test pattern merging. Evaluation results on ISCAS89 benchmark circuits indicate that the proposed method reduces the test application time by 6.89~62.67% and test data volume by 46.39~74.86%.

key words: *small-delay defects, test compaction, test application time, test data volume, on-chip delay measurement*

1. Introduction

In recent years, semiconductor device scaling has greatly improved performance and circuit integration density. With the increasing speed of integrated circuits, violations of the performance specifications are becoming a major factor affecting the product-quality level [1]. Timing related defects created by manufacturing process-related problems, such as resistive opens and shorts, metal mouse bites, via voids, will become more common [2]. Delay defects that degrade performance and cause timing related failures are emerging as a major problem in nanometer technologies [3]. Several delay fault models and delay test methods have been proposed. Transition fault and path delay fault are two prevalent fault models.

A small-delay defect is a delay defect with defect size not large enough to cause a timing failure under the system clock cycle. Usually, it is caused by resistive short, resistive open, or resistive via. Since they might escape detection during traditional Pass-Fail delay fault testing with functional clock, small-delay defects have become a significant problem and it is essential to detect such defects during manufacturing tests [4], [5]. Firstly, a timing failure might be triggered in the circuit during functional application caused by the increment of small delay on paths with small timing slacks [6]. The second reason is that small-delay defects can become a reliability issue because the defect might be worsened during subsequent aging in the field and cause a failure of the device [7]. In addition, in order to improve the yield and reduce the time-to-market of VLSIs, design-related failures and performance limiters need to be identified and rectified during the first silicon debug [8]. However, the cost of testing and debugging for delay defects in modern high-performance chips by using external high-speed *automatic test equipment* (ATE) is very high. Moreover, it is inherently limited by the accuracy of the provided test clock frequency with the external ATE, which can be affected by factors such as parasitic capacitance, resistance of probe and tester skew [9]. On-chip path delay time measurement is one of the best alternatives to solve these problems. By measuring delay time of the *path under measurement* (PUM), not only the gross and small-delay faults can be detected but also the amount of timing violation in the failing paths can be obtained under certain environment conditions [10], [11].

However, on-chip delay measurement incurs high test cost because it uses scan design, which brings about long test application time due to scan shift operation. Thus, a method reducing test application time is strongly required. In on-chip path delay measurement, the capture operation is unnecessary unlike the conventional delay testing. Thus, FFs keep the transition pattern (denoted as $v_{m,1}$) of the test pattern pair sensitizing a PUM p even after the measurement of p. If $v_{m,1}$ can be used as the initial pattern (denoted as $v_{n,0}$) of another test pattern pair (v_n which sensitizes another path p'), we can sensitize p' by just shifting 1 bit of the transition pattern (under LOS test). The proposed method uses this characteristic. This paper presents a method reduces scan shift time and test data volume by using scan-based test pattern merging. We can also reduce the switching activity induced by the launch pulse. As a result, this also reduces excessive IR-drop in scan testing avoiding test-induced yield loss.

The rest of the paper is organized as follows. Section 2 introduces some related works for small-delay testing. Section 3 describes some terminologies related to scan-based delay testing and the on-chip delay measurement. Section 4 explains methods to reduce test application time and test data volume. Section 5 evaluates the introduced method. Finally, Sect. 6 concludes this paper.

Manuscript received July 19, 2013.
Manuscript revised October 18, 2013.

[†]The authors are with the Graduate School of Advanced Integration Science, Chiba University, Chiba-shi, 263–8522 Japan.
a) E-mail: wenpo.zhang@ieee.org

DOI: 10.1587/transinf.E97.D.533

2. Related Work

Several on-chip methods have been proposed for delay testing and debugging. Methods using fast on-chip clock were proposed to detect delay faults [12], [13]. However, a small-delay defect occurring on a short path with a larger slack could escape the detection. To detect small-delay faults, methods with delay fault testing using a ring oscillator have been proposed [14]–[16]. In these, the PUM is made a part of ring oscillator, delay of the target path can be translated into oscillation period. However, the timing resolution is low. Some *time-to-voltage converter* (TVC) based schemes have been proposed [17]–[19]. The delay of the PUM is converted to a certain voltage, by comparing the converted voltage with the reference voltage, delay of the PUM can be got. These techniques give good timing resolution. However, the calibration is difficult.

Some on-chip path delay time measurement methods using embedded delay measurement were proposed [20]–[26]. In these, delay time of paths are measured. To screen small-delay fault, small-delay defect screening with criteria based on statistical analysis is used [5]. In this technique, small-delay defects are detected as outlier. The delay distributions for each path are generated in manufacturing tests. If a path delay time is beyond a specified time such as the three-sigma limit (users can set the specified time freely taking into consideration for the trade-off between yield and dependability), even if it is not beyond the system clock cycle, we regard it as a faulty path. Datta et al. proposed a modified *vernier delay line* (VDL) technique for path delay measurement [20]. High-resolution delay measurement capability can be achieved by using this method. The paper [21] presented modified boundary scan cells in which a *time-to-digital converter* (TDC) is embedded. Tsai et al. proposed a *built-in delay measurement* (BIDM) circuit consisting of coarse and fine blocks, which is an extension of the modified VDL technique. A built-in-self delay testing methodology based on BIDM and self-calibration methods can be developed [22]. The measurement used in [23] utilizes an area efficient method of the modified VDL. The feature of this method is delay range of each stage. The delay ranges increase by a factor of two gradually, which reduces the required stages. Thus, without decreasing delay measurement resolution, this method expands delay measurement range much easier with significantly less hardware overhead. The authors· group proposed a measurement system which is different from the VDL method to improve the accuracy of the measured value [24]. This method measures two paths, a path which includes the PUM and the other path whose length is almost equal to the redundant line of the path which is measured previously. The measured delay of the former path minus that of the latter path gives the delay of the PUM. This method is able to give a precise measurement. A method with smaller execution time and circuit area has been proposed [25]. In addition, a fault coverage improvement method has been proposed in [26]. However, testing using on-chip delay measurements incurs high test cost because it uses scan design, which brings about long test application time due to scan shift operation. Thus, a method reducing test application time is strongly required.

3. Preliminaries

This section introduces some terminologies related to scan-based delay testing and the on-chip delay measurement.

3.1 Terminology Related to Scan-Based Delay Testing

(1) Test pattern pair

In scan-based delay testing, a pair of test patterns $v_m = (v_{m,0}, v_{m,1})$ are applied to the PUT in two consecutive clock cycles. The pattern $v_{m,0}$ applied in first is an initial pattern, the pattern $v_{m,1}$ is a transition pattern. A pair of initial and transition patterns is called a test pattern pair. If the number of pattern pairs in one test pattern set V is n, we call n is the length of V.

(2) Transition fault and path delay fault

Transition fault and *path delay fault* are two prevalent fault models. The *transition fault* model targets each gate output in the design for a slow-to-rise and slow-to-fall delay fault while the *path delay fault* model targets the cumulative delay through the entire list of gates in a pre-defined path. *Transition fault* model is more widely used than *path delay fault* because it tests for at-speed failures at all nets in the design and the total fault list is equal to twice the number of nets. The *transition fault* is detected if a transition occurs at the fault site and if a sensitized path extends from the fault site to a primary output.

In this paper, we try to detect increases of gate and line delays caused by resistive faults to reduce early-life failure, the *transition fault* model is adopted. The small-delay fault coverage is equal to the transition fault coverage.

(3) LOS and LOC

For circuits using scan, there are two approaches to test delay faults: *launch off shift* (*LOS*) (or referred as *skewed-load*), and *launch off capture* (*LOC*) (or referred as *broad-side*). In the *LOS* method, the transition pattern is generated by one-bit shift of the initial pattern. In the *LOC* method, the transition pattern is obtained from the circuit response to the initial pattern. Note that in this paper we focus only on the *LOS* test.

3.2 On-Chip Delay Measurement Method

Figure 1 shows the architecture of the on-chip path delay measurement. The on-chip delay measurement system measures the delay of each path including a PUM, whose input and output are *start* and *stop*, respectively. The delay measurement system consists of *delay value measurement circuit* (DVMC), *stop signal generator* (SSG, which is an

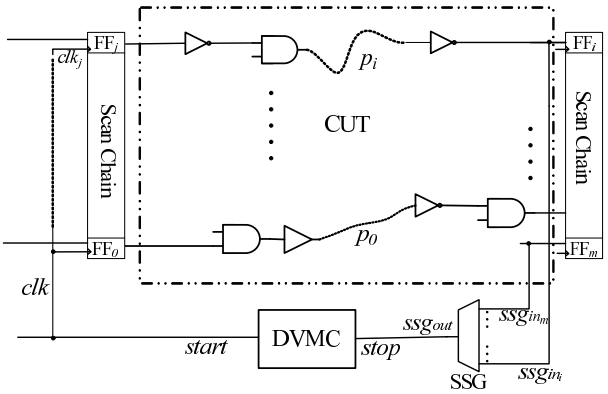

Fig. 1 Architecture of on-chip path delay measurement system.

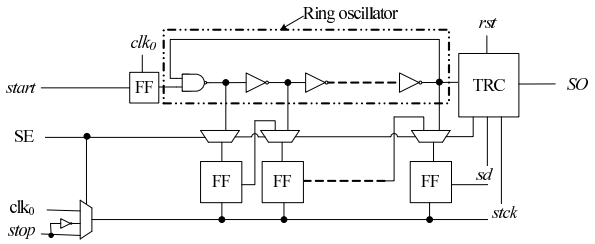

Fig. 2 Architecture of DVMC.

N-to-1 multiplexer), and *circuit under test* (CUT).

The clock line is directly connected to *start* of the DVMC; the DVMC starts the measurement when a positive transition is sent to *start*. The SSG detects the transition on the input of a designated flip-flop (FF) and sends the transition to *stop* of the DVMC, by setting the corresponding control data of SSG. The input line *clk* is the clock signal of the CUT. The line clk_i is the clock line of FF_i. The input of FF_i is connected to ssg_{out} through ssg_{in_i} and the SSG. The system measures a PUM including one clock line clk_j, a path p_i, and some redundant lines ssg_{in_i} and ssg_{out}. For example, after the measurement of the path $p' = clk_j$-p_i-ssg_{in_i}-ssg_{out}, by comparing the measured delay time with the expected delay time, small-delay defects on clk_j and p_i can be detected. In this paper, we insert one DVMC circuit in one CUT. Thus, only one path is selected to be measured for each test.

The architecture of the embedded delay measurement circuit DVMC is shown in Fig. 2. The DVMC is a ring oscillator based TDC, it measures the time difference between the transition signals sent to *start* line and *stop* line. The transition of *start* triggers the measurement. The TRC (an n-bit up counter) counts the round cycles of the oscillation. Synchronizing the transition of *stop*, the FFs capture the states on the output of corresponding NOT gates and the TRC. From these values, the delay value is calculated.

In one circuit, the set of paths under measurement is denoted by P (includes paths $p_0, p_1, \ldots, p_{(m-1)}$). Let D (which includes $d_0, d_1, \ldots, d_{(m-1)}$) be the control data of SSG. The data d_i selects the path p_i as the PUM. Let V (which includes test pattern pairs $v_0, v_1, \ldots, v_{(m-1)}$ for sensitizing paths $p_0, p_1, \ldots, p_{(m-1)}$) be the test data. The test flow of the on-chip delay measurement is as follows:

1. Select a path p_i from P for delay measurement by setting the corresponding control data of SSG to d_i.
2. Assign the test pattern which sensitizes p_i to the primary inputs and flip-flops of the CUT.
3. The transition reaches *stop* of the DVMC through p_i and the SSG, thus, the DVMC stops the measurement.
4. We get the measurement result through scan out of the DVMC. Consequently, the path delay of the PUM is calculated from the read out values.
5. Delete p_i from P. If $P = \emptyset$, stop the test; else go to Step 1.

4. The Proposed Method to Reduce Test Application Time and Test Data Volume for On-Chip Delay Measurement

This section proposes a method which reduces test application time and test data volume of the on-chip delay measurement by using scan-based test pattern merging. The LOS operation of the on-chip delay measurement is introduced in Sect. 4.1. Scan-based test pattern merging technique is explained in Sect. 4.2. In Sect. 4.3, we introduce our procedure for test application time and test data volume reduction. In 4.4 we analyze the test application time and test data volume.

4.1 LOS Operation of The On-Chip Delay Measurement

Figure 3 shows the LOS operation of the on-chip delay measurement. When using the on-chip delay measurement to detect small-delay defects on path p (from FF_i to FF_j), we set the SSG to detect the transition on the input of FF_j. At the moment the transition reaches the input D of FF_j, the transition is sent to *stop* of the DVMC through the SSG. Then, the DVMC stops the measurement. In this process, the capture operation of LOS test is unnecessary unlike the conventional LOS operation. Thus, FFs keep the transition pattern $v_{m,1}$ of the test pattern pair sensitizing p even after the measurement of p. If $v_{m,1}$ can be used as the initial pattern $v_{n,0}$ of another test pattern pair (v_n which sensitizes another path p'), we can sensitize p' by just shifting 1 bit of the transition pattern. Therefore, we can reduce the test application time. We can also reduce the switching activity induced by the capture pulse. Generally, an effective approach for avoiding test-induced malfunction and reducing IR-drop is to reduce switching activity induced by the capture pulse [4]. As a result, the proposed method also reduces excessive IR-drop in scan testing avoiding test-induced yield loss.

4.2 Scan-Based Test Pattern Merging

The scan-based test pattern merging technique is based on merging compatible patterns using scan shift operation. Two

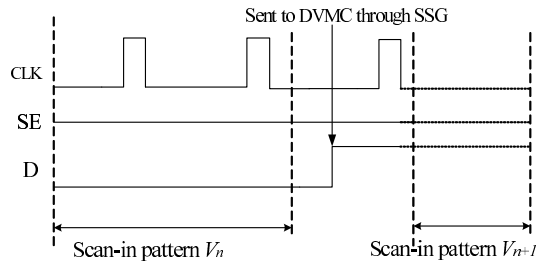

Fig. 3 LOS operation of on-chip delay measurements.

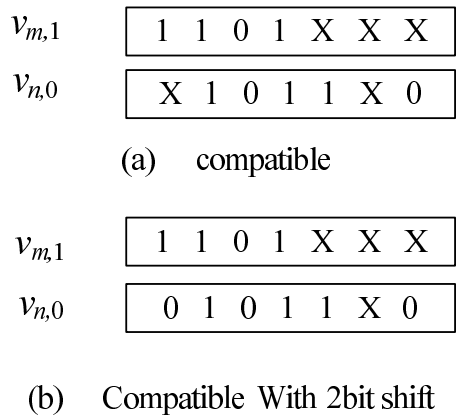

Fig. 4 Example of test pattern merging.

bits are compatible if they have the same value or any one of them is an X. Two patterns are considered compatible if every two corresponding bits in the two patterns are compatible.

Let $v_m = (v_{m,0}, v_{m,1})$, $v_n = (v_{n,0}, v_{n,1})$ be two test pattern pairs. As shown in Fig. 4 (a), if $v_{m,1}$ and $v_{n,0}$ are compatible, we say that the two pattern pairs are compatible. As shown in Fig. 4 (b), if we can make $v_{m,1}$ and $v_{n,0}$ compatible by shifting r bit of $v_{m,1}$, and $v_{m,1}$ and $v_{n,0}$ are not compatible without shifting, then we say that the two pattern pairs are compatible with r bit shift (in Fig. 4 (b) $r = 2$). If $v_{m,1}$ and $v_{n,0}$ are compatible, we do not need to scan-in all bits in $v_{n,0}$, and thus we need to scan-in only the last one bit in $v_{n,1}$ for v_n. If $v_{m,1}$ and $v_{n,0}$ are compatible with r bit shift, we need to scan-in the r bits in $v_{n,0}$ as well as the last one. In sum, we can reduce test data for v_n to 1 or (r+1) bits. Beside the 1 or (r+1) bits, we need to know the control data of shift times (denoted as S which includes $s_0, s_1, \ldots, s_{(m-1)}$ for controlling the shift time of test pattern pairs $v_0, v_1, \ldots, v_{(m-1)}$).

4.3 Procedure for Test Application Time and Test Data Volume Reduction

In this subsection, we describe the procedure for test application time and test data volume reduction. Specifically, we introduce the generation of the test data, the corresponding control data of SSG and the control data of shift times.

For illustration, an example of test compaction is shown in Fig. 5. We assume that the CUT contains three PUMs: p_0, p_1, p_2. Paths p_0, p_1 end in FF_i and p_2 ends in FF_j. Test pattern pairs for sensitizing paths p_0, p_1, p_2 are v_0 = (X1101XX, 1101XXX), v_1 = (11X011X, 1X011X1), v_2 = (X1011X0, 1011X0X). Here we use the LOS method to sensitize paths.

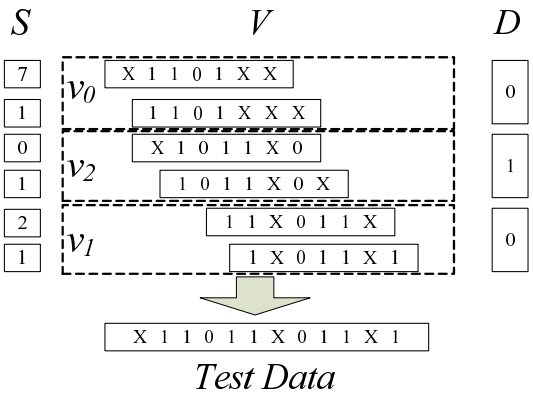

Fig. 5 Example of test compaction.

At first we need to decide the first path to be sensitized, here we choose p_0. Because p_0 ends in FF_i, the control data of SSG d_0 is set to 0 (00). The control data of shift time s_0 is 7 (111), which equals to the length of the scan chain. After the sensitization of p_0, the data stored in the FFs are $v_{0,1}$ = 1101XXX. Next, we try to sensitize p_2. The reason why we do not select p_1 is to reduce the test application time (in greedy way). Here, v_0 and v_2 are compatible, and v_0 and v_1 are compatible with 3 bit shift. This means that sensitizing p_1 requires shifting of 4=3+1 bits while sensitizing p_0 requires only 1 bit shift. Here, the control data of SSG d_1 is set to 1 (01), and the control data of shift time s_1 is set to 0 (000). At last we sensitize p_1, when the control data of SSG d_2 is set to 0 (00), and the control data of shift time s_2 is set to 2 (010) (v_2 and v_1 are compatible with 2 bit shift). After all the steps, we get the compacted test data V = (X11011X011X1), the corresponding D (the control data of SSG) and S (the control data of shift times). The procedure for reducing test application time and test data volume is given as follows.

Procedure 1: test application time and test data volume reduction.

1. Let V' be a set of test pattern pairs without applying the proposed method. Let V be an empty set (The objective compacted data will be obtained as V). Let i be an integer, and set i=0. Select and delete one test pattern pair v_m' from V'. Add v_m' to V as v_i.
2. Select and delete one test pattern pair v_n' from V', which is compatible with v_i with the minimum shift times. Add v_n' to V as v_{i+1}; i++.
3. If $V' = \emptyset$, stop; else go to Step 2.

4.4 Test Application Time and Test Data Volume

The test application time T is the sum of the scan shift time of test data T_S and the measurement result read out time T_R.

Here, we use time normalized as clock cycles. By considering the implementation of LOS test, we have the scan shift time of test data:

$$T_S = \sum_{i=0}^{n-1}(s_i + 1), \tag{1}$$

where n is the number of the test pattern pairs. Let T_D be the read out time of the DVMC, and the measurement result read out time appears as:

$$T_R = \sum_{i=0}^{n-1} T_D. \tag{2}$$

Therefore, the test application time is:

$$T = \sum_{i=0}^{n-1}(s_i + 1 + T_D). \tag{3}$$

The test data volume is the sum of data volume of $V(V_V)$, $S(V_S)$ and $D(V_D)$. By considering the implementation of LOS test, we have the data volume of test data:

$$V_V = \sum_{i=0}^{n-1}(s_i + 1). \tag{4}$$

When we implement the scan shift of LOS, the maximum shift number is equal to the length of scan chain. Thus, the data volume of the shift time and the data volume of the control data of SSG are:

$$V_S = V_D = n \log_2 N, \tag{5}$$

where N is the length of the scan chain. Therefore, the test data volume is:

$$V_{total} = \sum_{i=0}^{n-1}(s_i + 1) + 2n \log_2 N. \tag{6}$$

5. Experimental Result

In this section, we provide experimental results of the proposed test compaction method. In this evaluation, we use ISCAS 89 benchmark circuits. The initial test sets are constructed from the LOS test sets of [26]. The test set detects all the detectable transition faults under the single-path sensitization condition. A register is inserted to each primary input, and arbitrary values can be assigned to each register with scan-in operation. We use the ring oscillator based DVMC which has 14bit registers. Thus, we need 14 clock cycles to read out the result of the DVMC. First, we evaluate the test application time reducing effect of the proposed procedure. Next, we evaluate the data volume compaction effect of the proposed procedure. In the proposed procedure, we get a test pattern pair v_m from the original test set (the LOS test set of [26]). Then, we add v_m to the new test set and delete it from the original test set. After that we find another test pattern pair v_n which is compatible with v_m with the minimum shift times, add v_n to the new test set and delete it from the original test set. We repeat the above steps until the original test set is empty. We also compare the results of methods with/without reordering test patterns. In the procedure without pattern reordering, we get a test pattern pair v_m from the original test set (the LOS test set of [26]). Then, we add v_m to the new test set and delete it from the original test set. After that we get another test pattern pair v_{m+1} which is the next pair to v_m in the original test set. We find the minimum shift times which let v_{m+1} compatible with v_m, add v_{m+1} to the new test set and delete it from the original test set. We repeat the above steps until the original test set is empty. This paper just proposed a test compaction method by optimizing the test pattern. Note that the proposed method does not change the area overhead of the conventional on-chip delay measurement. The area overhead of the on-chip delay measurement compared to conventional scan design are 12~20% for some large ISCAS89 circuits [25], [26].

Table 1 shows the test application time of ISCAS89 benchmark circuits by using the conventional method [26] and the proposed method. Table 2 compares the test application time of methods with/without reordering test patterns. Here, test application time is calculated by using formula (3). In these Tables, the column *circuit* shows the circuit name. The column *CNV* shows the results of the conventional method in 10^4 clock cycles. The column *PRO* shows the results of the method using the proposed procedure1 (in 10^4 clock cycles). The column *COM* shows the results of the method using scan based pattern merging without reordering test patterns (in 10^4 clock cycles). The columns T_S and T_R show the scan shift time of test data and the measurement result read out time, respectively. The column T shows the test application time. The column T_{RED} shows the percentage of test application time reduction of each circuit using our method. From Table 1 we observe that the proposed method can effectively reduce the test application time very significantly in most of the benchmark circuits. Specifically, the test application time is reduced by 6.89~62.67% after the proposed compaction procedure. From Table 2, we notice that we reduce the test application time by 2.14~49.01% using scan based pattern merging without reordering test patterns. Our proposed procedure has better effect in test application time compaction than the method using only scan based pattern merging without reordering test patterns.

Table 3 shows the test data volume of ISCAS89 benchmark circuits by using the conventional method [26] and the proposed method. Table 4 compares the test data volume for methods with/without reordering test patterns. Here, test data volume is calculated by using formula (6). In these tables, the column *circuit* shows the circuit name. The column *CNV* shows the results of the conventional method in 10^4 bits. The column *PRO* shows the results of the method using the proposed procedure1 (in 10^4 bits). The column *COM* shows the results of the method using scan based pattern merging without reordering test patterns (in 10^4 bits). The columns V_S, V_D and V_V show the data volume of the shift time, the data volume of the control data of SSG and the data volume of test patterns V. The column V_{total} shows

Table 1 Test application time of ISCAS89 benchmark circuits.

Circuit	*CNV*(10^4 *clocks*)			*PRO*(10^4 *clocks*)			$T_{RED}(\%)$
	T_S	T_R	T	T_S	T_R	T	
s298	0.15	0.14	0.29	0.06	0.14	0.20	29.48
s344	0.26	0.23	0.49	0.16	0.23	0.38	21.46
s349	0.26	0.23	0.49	0.15	0.23	0.38	21.76
s382	0.34	0.22	0.56	0.14	0.22	0.36	35.33
s386	0.07	0.15	0.22	0.05	0.15	0.20	9.04
s444	0.29	0.18	0.47	0.11	0.18	0.29	38.39
s510	0.06	0.12	0.18	0.05	0.12	0.16	6.89
s526	0.38	0.24	0.62	0.13	0.24	0.37	40.87
s641	0.66	0.46	1.12	0.32	0.46	0.78	29.85
s713	0.42	0.29	0.71	0.17	0.29	0.47	34.64
s820	0.10	0.24	0.34	0.06	0.24	0.30	11.11
s832	0.10	0.23	0.33	0.06	0.23	0.29	11.23
s953	0.71	0.33	1.04	0.36	0.33	0.69	34.18
s1196	1.74	1.28	3.01	1.03	1.28	2.31	23.39
s1238	1.75	1.29	3.03	1.06	1.29	2.34	22.81
s1423	4.74	0.88	5.62	1.22	0.88	2.10	62.67
s1488	0.14	0.29	0.43	0.08	0.29	0.37	14.20
s1494	0.14	0.28	0.42	0.08	0.28	0.36	14.88
s5378	25.81	2.01	27.82	10.67	2.01	12.68	54.44
s9234	34.56	2.11	36.67	14.86	2.11	16.97	53.71
s13207	133.66	2.79	136.46	64.07	2.79	66.87	51.00
s38584	1823.95	17.57	1841.52	905.14	17.57	922.72	49.89
s35932	1890.83	15.31	1906.14	985.42	15.31	1000.73	47.50

Table 2 Test application time of with/without reordering test patterns.

Circuit	*COM*(10^4 *clocks*)				*PRO*(10^4 *clocks*)			
	T_S	T_R	T	$T_{RED}(\%)$	T_S	T_R	T	$T_{RED}(\%)$
s298	0.11	0.14	0.25	13.06	0.06	0.14	0.20	29.48
s344	0.21	0.23	0.43	11.25	0.16	0.23	0.38	21.46
s349	0.21	0.23	0.44	10.15	0.15	0.23	0.38	21.76
s382	0.25	0.22	0.46	16.98	0.14	0.22	0.36	35.33
s386	0.07	0.15	0.21	2.64	0.05	0.15	0.20	9.04
s444	0.21	0.18	0.39	15.52	0.11	0.18	0.29	38.39
s510	0.05	0.12	0.17	3.05	0.05	0.12	0.16	6.89
s526	0.25	0.24	0.49	21.94	0.13	0.24	0.37	40.87
s641	0.44	0.46	0.90	19.60	0.32	0.46	0.78	29.85
s713	0.28	0.29	0.58	19.00	0.17	0.29	0.47	34.64
s820	0.08	0.24	0.32	5.05	0.06	0.24	0.30	11.11
s832	0.08	0.23	0.31	5.51	0.06	0.23	0.29	11.23
s953	0.51	0.33	0.84	19.35	0.36	0.33	0.69	34.18
s1196	1.38	1.28	2.66	11.74	1.03	1.28	2.31	23.39
s1238	1.41	1.29	2.70	11.14	1.06	1.29	2.34	22.81
s1423	1.98	0.88	2.87	49.01	1.22	0.88	2.10	62.67
s1488	0.12	0.29	0.41	6.05	0.08	0.29	0.37	14.20
s1494	0.11	0.28	0.39	6.18	0.08	0.28	0.36	14.88
s5378	16.17	2.01	18.18	34.66	10.67	2.01	12.68	54.44
s9234	20.37	2.11	22.48	38.70	14.86	2.11	16.97	53.71
s13207	95.94	2.79	98.73	27.65	64.07	2.79	66.87	51.00
s38584	1216.46	17.57	1234.03	32.99	905.14	17.57	922.72	49.89
s35932	1271.48	15.31	1286.78	32.49	985.42	15.31	1000.73	47.50

the test data volume of each circuit. The column V_{RED} shows the percentage of test data volume reduction of each circuit by using our method. From Table 3 we observe that the proposed method can effectively reduce the test data volume very significantly in most of the benchmark circuits. Specifically, the test data volume is reduced by 46.39~74.86% after the proposed compaction procedure. Table 4 shows that we reduce the test data volume by 42.18~68.65% using scan based pattern merging without reordering test patterns. The proposed procedure has better effect in test data volume compaction than the method using only scan based pattern merging without reordering test patterns.

From the results of these Tables, we note that the proposed method can effectively reduce the test application time and test data volume very significantly in most of the benchmark circuits. Moreover, it has better effect on large circuits. The reason is that large circuits have more test patterns. We have more choice to minimize the shift times.

Table 3 Test data volume of ISCAS89 benchmark circuits.

Circuit	*CNV*(10^4 *bits*)				*PRO*(10^4 *bits*)				$V_{RED}(\%)$
	V_S	V_D	V_V	V_{total}	V_S	V_D	V_V	V_{total}	
s298	0.04	0.04	0.33	0.41	0.04	0.04	0.08	0.16	60.59
s344	0.06	0.06	0.78	0.91	0.06	0.06	0.29	0.42	54.22
s349	0.06	0.06	0.78	0.91	0.06	0.06	0.28	0.41	54.39
s382	0.08	0.08	0.74	0.89	0.08	0.08	0.18	0.33	63.15
s386	0.03	0.03	0.27	0.33	0.03	0.03	0.12	0.18	46.39
s444	0.06	0.06	0.62	0.75	0.06	0.06	0.13	0.26	65.04
s510	0.03	0.03	0.42	0.47	0.03	0.03	0.20	0.25	47.11
s526	0.09	0.09	0.83	1.00	0.09	0.09	0.16	0.33	66.62
s641	0.16	0.16	3.54	3.87	0.16	0.16	1.44	1.77	54.33
s713	0.10	0.10	2.26	2.47	0.10	0.10	0.88	1.09	55.69
s820	0.05	0.05	0.78	0.88	0.05	0.05	0.35	0.45	48.45
s832	0.05	0.05	0.75	0.85	0.05	0.05	0.34	0.44	48.50
s953	0.12	0.12	2.12	2.36	0.12	0.12	0.71	0.95	59.96
s1196	0.46	0.46	5.84	6.76	0.46	0.46	2.22	3.13	53.65
s1238	0.46	0.46	5.88	6.80	0.46	0.46	2.25	3.17	53.39
s1423	0.44	0.44	11.48	12.37	0.44	0.44	2.23	3.11	74.86
s1488	0.06	0.06	0.57	0.70	0.06	0.06	0.23	0.35	49.87
s1494	0.06	0.06	0.56	0.68	0.06	0.06	0.22	0.34	50.29
s5378	1.15	1.15	61.33	63.63	1.15	1.15	15.54	17.83	71.97
s9234	1.21	1.21	74.50	76.91	1.21	1.21	17.58	19.99	74.01
s13207	1.99	1.99	279.16	283.15	1.99	1.99	70.06	74.04	73.85
s38584	13.81	13.81	3675.23	3702.84	13.81	13.81	918.95	946.57	74.44
s35932	12.03	12.03	3855.68	3879.74	12.03	12.03	1022.60	1046.65	73.02

Table 4 Test data volume of with/without reordering test patterns.

Circuit	*COM*(10^4 *bits*)					*PRO*(10^4 *bits*)				
	V_S	V_D	V_V	V_{total}	$V_{RED}(\%)$	V_S	V_D	V_V	V_{total}	$V_{RED}(\%)$
s298	0.04	0.04	0.13	0.21	49.20	0.04	0.04	0.08	0.16	60.59
s344	0.06	0.06	0.34	0.47	48.74	0.06	0.06	0.29	0.42	54.22
s349	0.06	0.06	0.34	0.47	48.15	0.06	0.06	0.28	0.41	54.39
s382	0.08	0.08	0.28	0.43	51.72	0.08	0.08	0.18	0.33	63.15
s386	0.03	0.03	0.13	0.19	42.18	0.03	0.03	0.12	0.18	46.39
s444	0.06	0.06	0.24	0.37	50.78	0.06	0.06	0.13	0.26	65.04
s510	0.03	0.03	0.21	0.26	45.66	0.03	0.03	0.20	0.25	47.11
s526	0.09	0.09	0.28	0.45	54.83	0.09	0.09	0.16	0.33	66.62
s641	0.16	0.16	1.55	1.88	51.37	0.16	0.16	1.44	1.77	54.33
s713	0.10	0.10	1.00	1.20	51.17	0.10	0.10	0.88	1.09	55.69
s820	0.05	0.05	0.37	0.47	46.12	0.05	0.05	0.35	0.45	48.45
s832	0.05	0.05	0.36	0.46	46.30	0.05	0.05	0.34	0.44	48.50
s953	0.12	0.12	0.86	1.10	53.42	0.12	0.12	0.71	0.95	59.96
s1196	0.46	0.46	2.57	3.48	48.46	0.46	0.46	2.22	3.13	53.65
s1238	0.46	0.46	2.60	3.52	48.19	0.46	0.46	2.25	3.17	53.39
s1423	0.44	0.44	2.99	3.88	68.65	0.44	0.44	2.23	3.11	74.86
s1488	0.06	0.06	0.26	0.38	44.83	0.06	0.06	0.23	0.35	49.87
s1494	0.06	0.06	0.25	0.37	44.91	0.06	0.06	0.22	0.34	50.29
s5378	1.15	1.15	21.04	23.34	63.32	1.15	1.15	15.54	17.83	71.97
s9234	1.21	1.21	23.08	25.49	66.85	1.21	1.21	17.58	19.99	74.01
s13207	1.99	1.99	101.92	105.91	62.60	1.99	1.99	70.06	74.04	73.85
s38584	13.81	13.81	1230.27	1257.88	66.03	13.81	13.81	918.95	946.57	74.44
s35932	12.03	12.03	1308.65	1332.71	65.65	12.03	12.03	1022.60	1046.65	73.02

Then, we have better compaction effect on large circuits.

6. Conclusion

This paper proposed a test compaction method for on-chip delay measurements. To reduce test application time and test data volume of the on-chip delay measurement, this paper presented a method that uses scan-based test pattern merger. Experimental results on ISCAS89 benchmark circuits showed that the proposed method reduced the test application time by 6.89~62.67% and the test data volume by 46.39~74.86%.

In this work, we proposed a method to reduce test application time and test data volume by using only scan-based test pattern merging. By analyzing the results in Table 1, we noticed that the measurement result read out time occupied a considerable part of the total test time. In our future work, we will consider a new method to reduce the measurement

result read out time. In this work, we used an in-house ATPG. As our future work, we also try to use a commercial ATPG for efficient test generation.

Acknowledgment

This work was supported by VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Synopsys, Inc., Cadence Design Systems, Inc. and Mentor Graphics, Inc.

References

[1] A. Krstic and K.T. Cheng, Delay fault testing for VLSI circuits, Kluwer Academic Publishers, 1998.
[2] C. Hawkins, A. Keshavarzi, and J. Segura, "A view from the bottom: Nanometer technology AC parametric failures-why, where, and how to detect," Proc. IEEE Int'l Symp. Defect Fault Tolerance VLSI Syst., pp.267–276, 2003.
[3] A.D. Singh and G. Xu, "Output hazard-free transition tests for silicon calibrated scan based delay testing," Proc. IEEE VLSI Test Symp., pp.349–357, 2006.
[4] M. Tehranipoor and N. Ahmed, Nanometer Technology Designs: High-Quality Delay Tests, Springer Publishing Company, 2007.
[5] K. Noguchi, K. Nose, T. Ono, and M. Mizuno, "A small-delay defect detection technique for dependable LSIs," Proc. IEEE Symp. VLSI Circuits, pp.64–65, 2008.
[6] B. Kruseman, A.K. Majhi, G. Gronthoud, and S. Eichenberger, "On hazard-free patterns for fine-delay fault testing," Proc. IEEE Int'l Test Conf., pp.213–222, 2004.
[7] P. Nigh and A. Gattiker, "Test method evaluation experiments & data," Proc. IEEE Int'l Test Conf., pp.454–463, 2000.
[8] H. Balachandran, K.M. Butler, and N. Simpson, "Facilitating rapid first silicon debug," Proc. IEEE Int'l Test Conf., pp.628–637, 2002.
[9] S.K. Sunter, "BIST vs. ATE: Need a different vehicle?," Proc. IEEE Int'l Test Conf., p.1148, 1998.
[10] R. Datta, A. Sebastine, and J.A. Abraham, "Delay fault testing and silicon debug using scan chains," Proc. IEEE European Test Symp., pp.46–51, 2004.
[11] R. Datta, G. Carpenter, K. Nowka, and J.A. Abraham, "A scheme for on-chip timing characterization," Proc. IEEE VLSI Test Symp., pp.24–29, 2006.
[12] N. Ahmed, M. Tehranipoor, and V. Jayaram, "Timing-based delay test for screening small delay defects," Proc. IEEE/ACM Design Automation Conf., pp.320–325, 2006.
[13] X. Fan, Y. Hu, and L.T. Wang, "An on-chip test clock control scheme for multi-clock at-speed testing," Proc. IEEE Asian Test Symp., pp.341–348, 2007.
[14] M. Collins and B.M. Al-Hashimi, "On-chip time measurement architecture with femtosecond timing resolution," Proc. IEEE European Test Symp., pp.103–110, 2006.
[15] X. Wang, M. Tehranipoor, and R. Datta, "Path-RO: A novel on-chip critical path delay measurement under process variations," Proc. IEEE/ACM Int'l Conf. Computer-Aided Design., pp.640–646, 2008.
[16] A. Jain, A. Veggetti, D. Crippa, and P. Rolandi, "An on-chip flip-flop characterization circuit," Proc. Int'l conf. Integr. Circuit and Sys. Design, pp.41–50, 2010.
[17] M. Collins, B.M. Al-Hashimi, and N. Ross, "A programmable time measurement architecture for embedded memory characterization," Proc. IEEE European Test Symp., pp.128–133, 2005.
[18] S. Ghosh, S. Bhunia, A. Raychowdhury, and K. Roy, "A novel delay fault testing methodology using low-overhead built-in delay sensor," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.25, no.12, pp.2934–2943, Dec. 2006.
[19] A. Raychowdhury, S. Ghosh, and K. Roy, "A novel on-chip delay measurement hardware for efficient speed-binning," Proc. IEEE Int'l On-Line Testing Symp., pp.287–292, 2005.
[20] R. Datta, A. Sebastine, A. Raghunathan, and J.A. Abraham, "On-chip delay measurement for silicon debug," Proce. ACM Great Lakes Symp. VLSI, pp.145–148, 2004.
[21] H. Yotsuyanagi, H. Makimoto, and M. Hashizume, "A boundary scan circuit with time-to-digital converter for delay testing," Proc. IEEE Asian Test Symp., pp.539–544, 2011.
[22] M.C. Tsai, C.H. Cheng, and C.M. Yang, "An all-digital high-precision built-in delay time measurement circuit," Proc. IEEE VLSI Test Symp., pp.249–254, 2008.
[23] S. Pei, H. Li, and X. Li, "A low overhead on-chip path delay measurement circuit," Proc. IEEE Asian Test Symp., pp.145–150, 2009.
[24] T. Tanabe, K. Katoh, K. Namba, and H. Ito, "A delay measurement for VLSI circuit by subtraction," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J93-D, no.4, pp.460–468, April 2010.
[25] K. Katoh, K. Namba, and H. Ito, "A low area on-chip delay measurement system using embedded delay measurement circuit," Proc. IEEE Asian Test Symp., pp.343–348, 2010.
[26] W. Zhang, K. Namba, and H. Ito, "Improving small-delay fault coverage for on-chip delay measurement," Proc. IEEE Int'l Symp. Defect Fault Tolerance VLSI Syst., pp.193–198, 2012.

Wenpo Zhang received B.E. degree from University of Electronic Science and Technology of China in 2004, and the M.E. degree from Chiba University in 2012. Currently, he is working toward the Ph.D. degree at Chiba University. His research interests include very large scale integration (VLSI) testing and design for testability. He is a student member of the IEEE.

Kazuteru Namba received B.E., M.E. and Ph.D. from Tokyo Institute of Technology in 1997, 1999 and 2002, respectively. He joined Chiba University in 2002. He is currently an Assistant Professor of Graduate School of Advanced Integration Science, Chiba University. His current research interests include dependable computing. He is a member of the IEEE and the IPSJ.

Hideo Ito was born in Chiba, Japan, on June 1, 1946. He received the B.E. degree from Chiba University in 1969 and the D.E. degree from Tokyo Institute of Technology in 1984. He joined Nippon Electric Co. Ltd. in 1969 and Kisarazu Technical College in 1971. He had been a member of Chiba University from 1973 until March 2012. He is currently a Professor Emeritus of Graduate School of Advanced Integration Science. He had been a member of the IEEE and the IPSJ.