A novel scan segmentation design method for avoiding shift timing failure in scan testing

著者	Yamato Yuta, Wen Xiaoqing, Kochte Michael A., Miyase Kohei, Kajihara Seiji, Wang Laung-Terng
journal or publication title	2011 IEEE International Test Conference
year	2012-01-26
その他のタイトル	A Novel Scan Segmentation Design Method for Avoiding Shift Timing Failure in Scan Testing
URL	http://hdl.handle.net/10228/00007594

doi: info:doi/10.1109/TEST.2011.6139162

A Novel Scan Segmentation Design Method for Avoiding Shift Timing Failure in Scan Testing

Yuta Yamato [1], Xiaoqing Wen [2], Michael A. Kochte [2,3], Kohei Miyase [2], Seiji Kajihara [2], and Laung-Terng Wang [4]

[1] Fukuoka Industry Science Technology Foundation, Fukuoka, Japan
[2] Kyushu Institute of Technology, Iizuka, Japan
[3] University of Stuttgart, Stuttgart, Germany
[4] SynTest Technologies, Inc, Sunnyvale, CA, USA

Abstract

High power consumption in scan testing can cause undue yield loss which has increasingly become a serious problem for deep-submicron VLSI circuits. Growing evidence attributes this problem to shift timing failures, which are primarily caused by excessive switching activity in the proximities of clock paths that tends to introduce severe clock skew due to IR-drop-induced delay increase. This paper is the first of its kind to address this critical issue with a novel layout-aware scheme based on scan segmentation design, called ***LCTI-SS*** *(**L**ow-**C**lock-**T**ree-**I**mpact **S**can **S**egmentation). An optimal combination of scan segments is identified for simultaneous clocking so that the switching activity in the proximities of clock trees is reduced while maintaining the average power reduction effect on conventional scan segmentation. Experimental results on benchmark and industrial circuits have demonstrated the advantage of the LCTI-SS scheme.*

Keywords: *scan testing, shift power reduction, scan segmentation, switching activity, clock tree, clock skew.*

1. Introduction

Scan design is the most widely used ***design-for-testability*** (DFT) technique [1]. It provides external access to the flip-flops (FFs) in a design by replacing FFs with scan cells and stitching them into one or more shift registers called ***scan chains***. As a result, scan design has made it possible to test sequential circuits with reduced complexity and in practical time. In recent years, ***at-speed scan testing***, which is realized by launching a transition and capturing its response at the system speed, has become mandatory in order to guarantee sufficient quality levels for ***deep-submicron*** (**DSM**) VLSI circuits. This is because timing-related defects have become dominant in such circuits.

In practice, at-speed scan testing is usually realized by the ***launch-on-capture*** (**LOC**) scheme since it has lower physical design complexity for the scan enable signal than other clocking schemes [2]. The basic scheme of LOC is shown in Fig. 1. In ***shift*** mode (SE = 1), a test vector is applied by operating scan chains as shift registers with multiple shift clock pulses (S_1 to S_L). Then, in ***capture*** mode (SE = 0), two capture pulses C_1 and C_2 are applied for launching a transition at the start-point of a path and capturing the circuit response to the launched transition at the end-point of the path. For at-speed scan testing, the test cycle T should be made equal to the functional clock cycle, which is extremely short for a high-speed design.

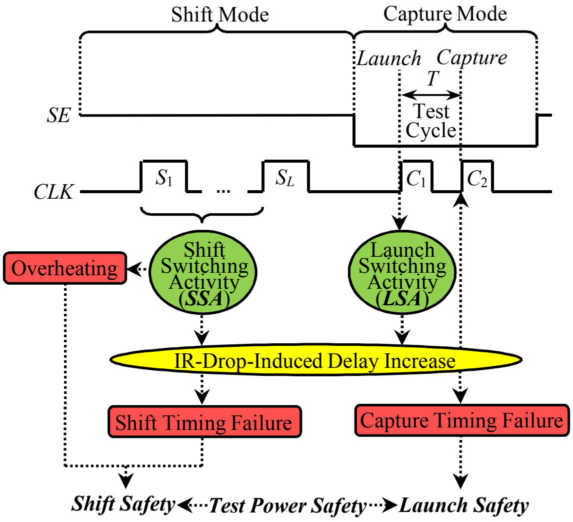


Fig. 1 Test power safety issues.

1.1 Test Power Safety in At-Speed Scan Testing

At-speed scan testing is indispensable for DSM VLSI circuits. However, its power dissipation, i.e., ***test power***, is increasingly causing various problems, threatening its ***test power safety***. The reasons are illustrated in Fig. 1 and described as follows:

In shift mode, the accumulative impact of excessive ***shift switching activity*** (***SSA***) may cause overheating of dies or chip packages due to excessively increased average power dissipation. This is because most of the test application time is spent in shift mode, especially for circuits with long scan chains. At the same time, the instantaneous impact of excessive SSA may cause IR-drop-induced delay increase along scan paths as well as clock paths, which ends up with shift timing failures such as setup or hold time violations and thus yield loss [3, 4]. On the other hand, in capture mode, the instantaneous impact of excessive ***launch switching activity*** (***LSA***) at the launch cycle C_1 may cause excessive IR-drop-induced delay increase along sensitized paths, leading to capture timing failures at the capture cycle C_2 and thus yield loss. The reasons are that the test cycle T is extremely short for high-speed circuits, and that low-power circuits are more susceptible to changes in power supply voltages [5-9].

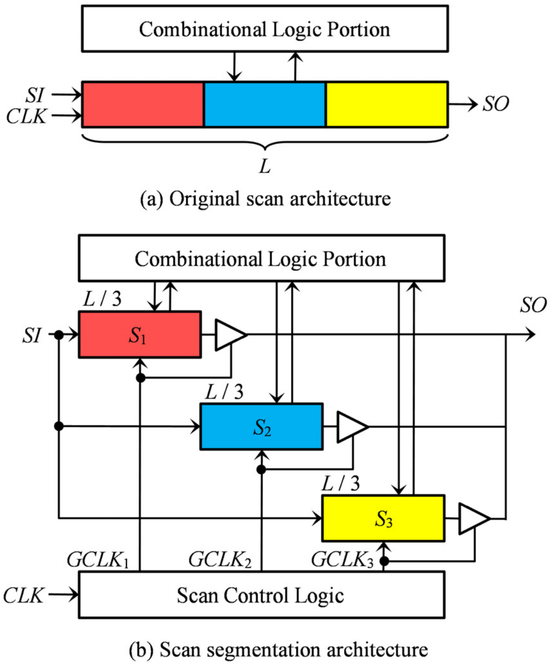
Therefore, ***test power safety***, the combination of both ***shift safety*** and ***launch safety***, should be guaranteed for at-speed scan testing in order to avoid chip/package damage, undue yield loss, and reliability degradation [7].

1.2 Previous Solutions for Test Power Safety

Generally, test power safety needs to be achieved by properly reducing both SSA and LSA, as illustrated in Fig. 1. Previous solutions for reducing LSA and SSA are based on either ***circuit modification*** or ***test data manipulation*** [6, 7]. Generally, it is preferable to reduce LSA by test data manipulation since this approach causes no adverse impact on ATPG, circuit design, and performance. Several effective test-data-manipulation-based techniques [10-13] exist for reducing LSA, which are helpful in achieving launch safety. On the other hand, it is preferable to reduce SSA by circuit modification since SSA often needs to be significantly and predictably reduced to meet the heat management requirement of packaging. Furthermore, circuit modification in shift mode causes neither ATPG change nor fault coverage loss. Several circuit-modification-based approaches for reducing SSA have been proposed so far, as summarized below:

Scan clock gating [14, 15] searches for test patterns that do not detect any new faults during BIST, and disable scan FFs while these redundant patterns are applied. Obviously, the SSA reduction effect of this approach is highly dependent on the redundant pattern count. ***Scan chain disabling*** reduces the number of active scan chains [16] during shift and capture. This approach can also be applied with power-aware test planning for BIST [17] to reduce average SSA significantly. ***Toggle suppression*** [14] inserts blocking logic to the outputs of scan FFs, thereby significantly reducing the average SSA in the combinational portion. However, circuit performance degradation may occur due to the insertion of blocking logic into functional paths. ***Scan cell ordering*** [18] tries to find a proper order of scan FFs for a given test set, but its SSA reduction effect is highly test-set-dependent. Compared with the above approaches, ***scan segmentation*** [19-22] is a more preferable approach for reducing SSA.

Fig. 2 illustrates the structure of conventional scan segmentation [19]. The basic idea is to split a scan chain into multiple segments, and make sure to shift just one segment of the scan chain at a time while keeping all other segments deactivated. In Fig. 2, the original scan chain with length *L* (Fig. 2 (a)) is split into 3 shorter segments with length *L*/3 (Fig. 2 (b)). The shift operation is conducted for segments S_1, S_2, and S_3, one by one. The currently inactive segments are silenced by gating their scan clocks. The most significant benefit of segmentation scan is that average SSA can be effectively and predictably reduced since it limits the number of scan FFs where transitions occur simultaneously. In addition, scan segmentation causes no performance degradation since it inserts no additional logic to functional paths. Furthermore, the SSA reduction effect of scan segmentation is independent of the given test set, which can be easily generated by conventional ATPG.



(a) Original scan architecture

(b) Scan segmentation architecture

Fig. 2 Conventional scan segmentation.

1.3 Shift Timing Failures

Conventional scan segmentation can effectively and predictably reduce the ***accumulative*** impact of excessive SSA, thus effectively solving the overheat problem due to average SSA. However, it is unable to mitigate the ***instantaneous*** impact of excessive SSA. As a result, IR-drop-induced delay increase may still occur along clock paths from a clock pin to scan FFs, which may lead to shift timing failures and thus severely damaging shift safety. This problem is illustrated in Fig. 3.

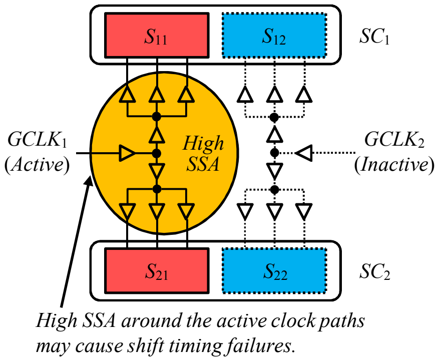

S11
S12
SC1
GCLK1
(Active)
High
SSA
GCLK2
(Inactive)
S21
S22
SC2
High SSA around the active clock paths
may cause shift timing failures.

Fig. 3 Problem of conventional scan segmentation.

In Fig. 3, scan chains SC_1 and SC_2 are split into two scan segments $\{S_{11}, S_{12}\}$ and $\{S_{21}, S_{22}\}$ respectively. The shift operation is conducted by clocking S_{11} and S_{21} together the first, followed by clocking S_{12} and S_{22} together the next. Although this scheme can reduce the global (whole-circuit) average SSA by approximately 50%, the SSA around clock paths to the active segments may still be high. That is, IR-drop-induced delay increase may still occur along the clock paths, resulting in severe clock skew. As a result, shift timing failures may occur at scan FFs, resulting in undue yield loss.

The above discussions clearly points to a new problem that threatens shift safety, i.e., excessive SSA around clock paths. In the context of scan segmentation, this problem translates into that reducing only global average SSA by conventional scan segmentation cannot guarantee shift safety. Therefore, there is a strong need to effectively reduce SSA around clock paths in scan segmentation.

1.4 Contribution and Paper Organization

This paper addresses the new shift safety problem caused by excessive SSA around clock paths in conventional scan segmentation. The basic idea is to optimize the combination of scan segments for simultaneous clocking since SSA depends on which segments are simultaneously clocked. For example, conventional scan segmentation shown in Fig. 3 uses segment groups $\{S_{11}, S_{21}\}$ and $\{S_{12}, S_{22}\}$. However, SSA around clock paths may be potentially reduced by using a different segment grouping, e.g., $\{S_{11}, S_{22}\}$ and $\{S_{12}, S_{21}\}$. Therefore, we propose a new scan segmentation scheme in which segment grouping is optimized for SSA reduction around clock paths.

The major contribution of this paper is to propose a novel layout-aware scan segmentation clocking scheme, called ***LCTI-SS*** (***L****ow-****C****lock-****T****ree-****I****mpact* ***S****can* ***S****egmentation*). LCTI-SS deals with the real cause of excessive-SSA-induced yield loss by reducing SSA in proximities of active clock paths (called ***impact areas***) while preserving the benefits of conventional scan segmentation in reducing average whole-circuit shift power. A sophisticated segment regrouping algorithm is devised to directly reduce SSA in impact areas by optimizing the grouping of scan segments for simultaneous clocking. LCTI-SS improves shift safety since the reduction of instantaneous SSA is directly focused on impact areas to significantly reduce IR-drop-induced shift timing failures. To our best knowledge, this paper is the first of its kind to mitigate the impact of shift switching activity on clock paths.

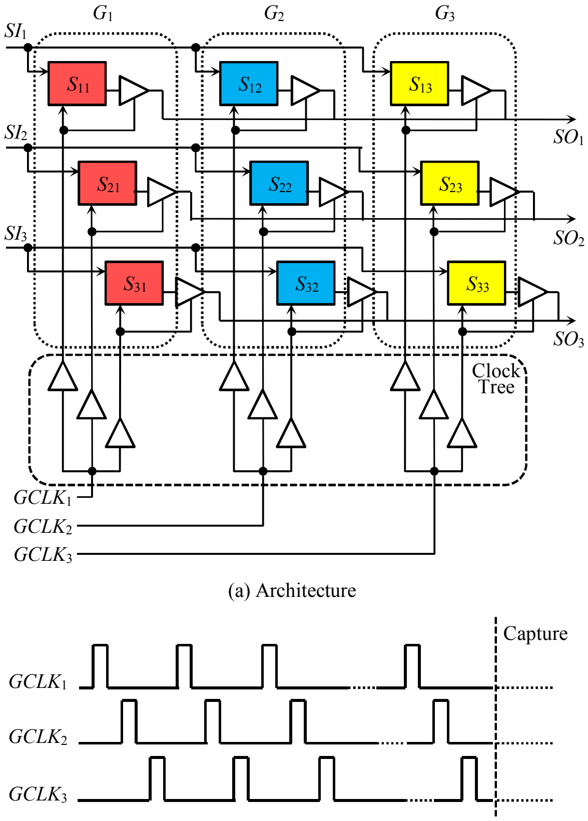
The rest of this paper is organized as follows: Section 2 reviews conventional scan segmentation, Section 3 presents the proposed LCTI-SS scheme, Section 4 and Section 5 present the details of impact area identification and segment regrouping, respectively, Section 6 shows experimental results, and Section 7 concludes the paper.

2. Background

This section first describes the details of conventional scan segmentation for circuits with multiple scan chains. It then reviews previous clocking schemes proposed for reducing shift power in such circuits.

2.1 Conventional Multi-Scan Segmentation

Most of scan circuits contain multiple scan chains. Fig. 4 shows a conventional scan segmentation design for a circuit with 3 scan chains. Each scan chain is split into 3 segments, resulting 9 segments S_{11} to S_{33}. Three gated clocks $GCLK_1$, $GCLK_2$, and $GCLK_3$ are connected to all scan FFs in 3 segment groups $G_1 = \{S_{11}, S_{21}, S_{31}\}$, $G_2 = \{S_{12}, S_{22}, S_{32}\}$, and $G_3 = \{S_{13}, S_{23}, S_{33}\}$, respectively. Similar to scan segmentation for a single scan chain, the shift operation is conducted for G_1, G_2, and G_3, one at a time. As shown in Fig. 4 (b), gated clocks $GCLK_1$, $GCLK_2$, and $GCLK_3$ are exclusively enabled during a shift operation, Note that the test response to a test vector is captured by enabling all gated clock signals after a test vector has been shifted into all segments. Since the number of simultaneously-switching FFs becomes smaller, global average SSA is effectively reduced. Note that no modification is required on functional paths, thus avoiding any performance degradation. In addition, test application time remains the same as that of the standard scan architecture. Generally, the average shift power reduction ratio is approximately 50% for a 2-segment configuration and 66% for a 3-segment configuration [7].



(a) Architecture

(b) Timing diagram

Fig. 4 Conventional multi-scan segmentation.

The proposed LCTI-SS scheme is especially suitable for such multi-scan circuits. This is because in a multi-scan segmentation design, multiple segments are simultaneously clocked and there exists a possibility of selecting an optimal group of segments for simultaneous clocking so that the impact of SSA on clock paths is reduced.

2.2 Previous Low-Shift-Power Clocking Schemes

The number of simultaneously-switching FFs can be reduced by manipulating shift clocks. In ***staggered clocking*** [20] as shown in Fig. 5 (a), the shift clock edges are skewed by staggering clocks. In ***MD-SCAN*** [23] as shown in Fig. 5(b), the shift clock edges are skewed by introducing multiple clock duty cycles with different lengths. Both clocking schemes can reduce the number of simultaneously-switching FFs. Obviously, this results in lower global average SSA.

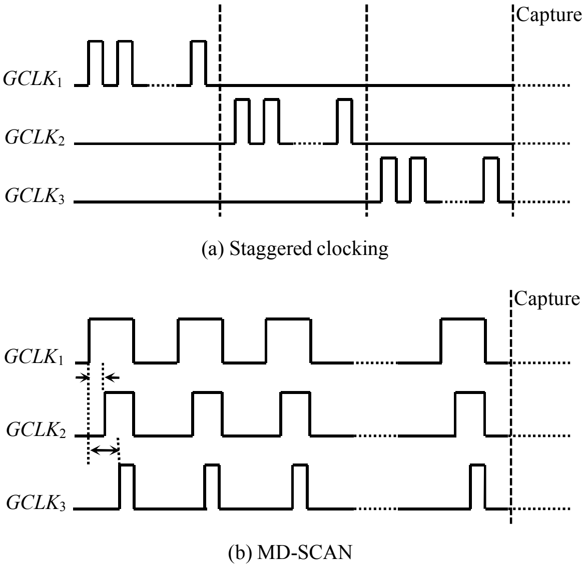


Fig. 5 Clocking schemes for shift power reduction.

However, shift timing failures may still occur in conventional scan segmentation even when these clocking schemes are employed. As described in Subsection 1.3, the reason is that excessive IR-drop around clock paths may cause severe clock skew in clock paths, resulting in hold time violations in FFs [3, 4], which cannot be avoided by simply lowering the clock frequency [25].

3. The LCTI-SS Scheme

This section describes ***Low-Clock-Tree-Impact Scan Segmentation*** (LCTI-SS), for reducing the instantaneous *shift switching activity* (SSA) in the proximities of clock trees in shift mode so as to reduce the risk of timing failures in shift chains. Together with the intrinsic benefit of scan segmentation for reducing global average SSA to mitigate the overheat problem, the proposed LCTI-SS significantly improves shift safety.

Fig. 6 shows the general flow of the proposed LCTI-SS scheme. It consists of two major steps: ***impact area identification*** (①) and ***segment regrouping*** (②), as described below:

Given a circuit netlist N with standard full-scan design, conventional scan segmentation (as illustrated in Fig. 4) is first designed. The result is a new netlist N', for which place-and-route is conducted to produce a layout design L and a clock tree design C. Based on these two types of information, *impact area identification* (①) is conducted to identify nodes (gates and FFs) whose transitions have significant impact on IR-drop-induced delay increase on clock paths. After that, *segment regrouping* (②) is conducted to minimize the number of nodes in impact areas which may affect active clock paths. As a result, netlist N'', layout L', and clock tree C' are obtained by reconnecting gated clocks to corresponding segments. An alternative to clock tree modification is to use a programmable clock control [16, 25].

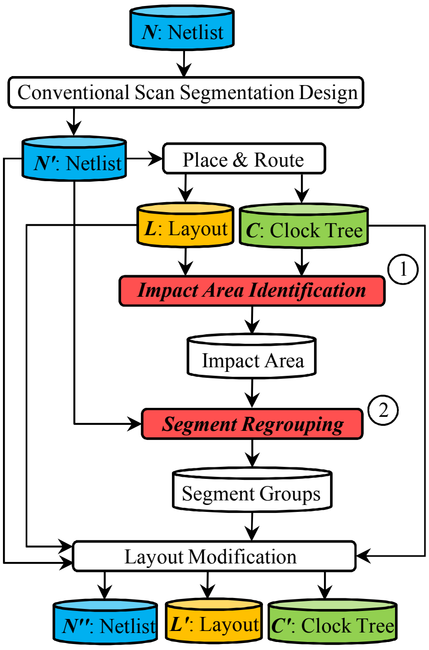


Fig. 6 General flow of the LCTI-SS scheme.

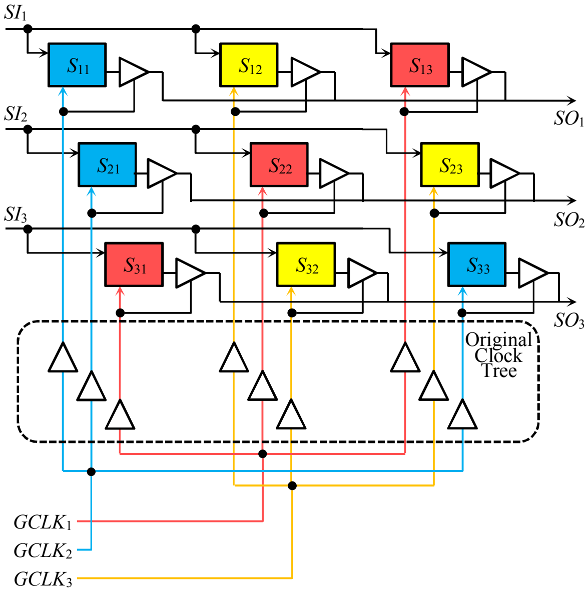


Fig. 7 Example of segment regrouping.

To illustrate the LCTI-SS scheme, let us revisit the case shown in Fig. 4. Here, the initial segment groups provided by conventional scan segmentation are $G_1 = \{S_{11}, S_{21}, S_{31}\}$,

$G_2 = \{S_{12}, S_{22}, S_{32}\}$, and $G_3 = \{S_{13}, S_{23}, S_{33}\}$. By applying the LCTI-SS scheme, scan segments are regrouped, for example, into $G_1' = \{S_{13}, S_{22}, S_{31}\}$, $G_2' = \{S_{11}, S_{21}, S_{33}\}$, and $G_3' = \{S_{12}, S_{23}, S_{32}\}$, as shown in Fig. 7. Gated clocks are reconnected to each corresponding segment group while most of the original clock tree design remains unchanged.

4. Impact Area Identification

This section presents the details about impact area identification, which is a critical step in LCTI-SS.

Definition* 1**: The ***clock aggressor set of a clock buffer B, denoted by $CAS(B)$, is a set of nodes (gates and FFs) placed near B and sharing power rails with B.

Fig. 8 shows an example, where $CAS(B) = \{N_6, N_7, N_{10}, N_{11}, N_{14}, N_{15}\}$ for the clock buffer B.

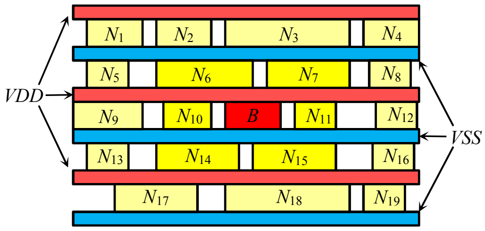


Fig. 8 Example of clock aggressor set.

Definition* 2:** Let P be a path consisting of all clock buffers $\{B_1, B_2, ..., B_m\}$ from a gated clock pin to the clock input of a scan FF. The ***clock aggressor region of P, denoted by $CAR(P)$, is defined as

$$CAR(P) = \bigcup_{i=1}^{m} (CAS(B_i))$$

Definition* 3**: Let S be a scan segment consisting of scan FFs $\{FF_1, FF_2, ..., FF_n\}$ and let P_i be a clock path to FF_i ($i = 1, 2, \ldots, n$). The ***impact area of S, denoted by $IA(S)$, is defined as

$$IA(S) = \bigcup_{i=1}^{n} (CAR(P_i))$$

An example is shown in Fig. 9, where two scan FFs, FF_1 and FF_2, are assumed to form the scan segment S_{11}. Here, $CAR(P_1) = CAS(B_1) \cup CAS(B_2) \cup CAS(B_3)$, $CAR(P_2) = CAS(B_1) \cup CAS(B_2) \cup CAS(B_4)$. As a result, $IA(S_{11}) = CAR(P_1) \cup CAR(P_2)$.

Although each segment has an impact area, it does not necessarily mean that all nodes (i.e., clock aggressors) in the impact area may affect propagation delay of clock paths. Generally, a node impacting active clock buffers needs to satisfy the following two conditions:

***Condition* A**: The node belongs to at least one impact area of active segments.
***Condition* B**: The node is structurally reachable from at least one scan FF in active segments.

Definition* 4**: Let $RCAS(S)$ be a set of clock aggressors structurally reachable from all FFs in a segment S, and let G be a segment group composed of segments S_1, S_2, ..., and S_n to be clocked simultaneously. The ***impact aggressor set of G, denoted by $IAS(G)$, is defined as

$$IAS(G) = \bigcup_{i=1}^{n} (IA(S_i)) \cap \bigcup_{i=1}^{n} (RCAS(S_i))$$

Clearly, the impact aggressor set of G contains only clock aggressors that may affect active clock paths, i.e., clock aggressors satisfying both Condition A and Condition B. An example is shown in Fig. 10. Here, two scan segments S_{11} and S_{21} are assumed to belong to G_1. $IA(S_{11}) = \{N_1, N_2, N_3, N_5, N_7, N_6\}$, $IA(S_{21}) = \{N_4, N_5, N_6, N_7, N_8, N_9\}$, $RCAS(S_{11}) = \{N_1, N_2, N_3, N_5, N_7\}$, and $RCAS(S_{21}) = \{N_3, N_5, N_6, N_8\}$. In this case, $IAS(G_1) = (IA(S_{11}) \cup IA(S_{21})) \cap (RCAS(S_{11}) \cup RCAS(S_{21})) = \{N_1, N_2, N_3, N_5, N_6, N_8\}$.

From above definitions, the impact aggressor set of a segment group with arbitrary combinations of scan segments can be calculated. This information is used to estimate the risk of shift timing failures.

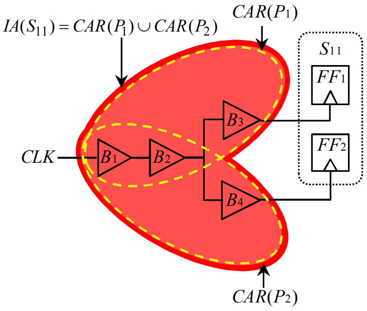


Fig. 9 Example of clock aggressor region and impact area.

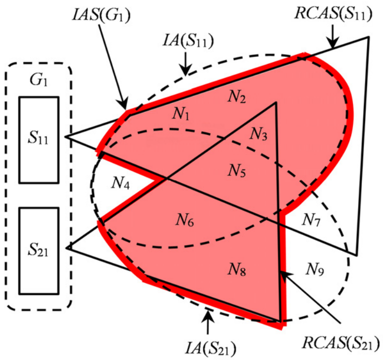


Fig. 10 Example of impact aggressor set.

5. Segment Regrouping

Generally, the number of impact aggressors depends on the combination of segments to be simultaneously clocked. The smaller the number of impact aggressors, the lower the probability of simultaneous transitions at impact aggressors. This indicates that it is possible to regroup segments optimally so that each segment group has a smaller number of impact aggressors. This section presents an effective algorithm for segment regrouping, which is another critical step in the LCTI-SS scheme.

The proposed algorithm for segment regrouping uses the ***weighted switching activity*** (WSA) metric for SSA estimation since this metric has good correlation with power dissipation [5, 11] and IR-drop [26].

Definition* 5**: Let *IAS* be an impact aggressor set. The ***weighted impact of *IAS*, denoted by *WI*(*IAS*), is defined as

$$WI(IAS) = \sum_{i=1}^{n}(w_i)$$

where n is the number of nodes in the impact aggressor set, and w_i is the weight of node i (i = 1, 2, …, n), which can be approximated by the number of its fanout branches.

Based on the above definitions, the problem of segment regrouping can be formalized as follows:

Segment Regrouping Problem: Given a scan segmentation design with m scan chains and n segments for each scan chain, find n segment groups G_1, G_2, ..., G_n such that the weighted impact of the impact aggressor set for each segment group G_i (i = 1, 2, …, n), namely $WI(IAS(G_i))$, is minimized.

Theoretically, the total number of segment group combinations can be expressed by the following theorem:

***Theorem* 1**: For a scan segmentation design with m scan chains and n segments for each scan chain, the total number of segment group combinations is $(n!)^m$.

Proof: For the first segment group, n segments can be selected from each of the m scan chains, which results in n^m possible combinations. Then, repeating this until the n-th segment group result in $(n\text{-}1)^m$ possible combinations for the second segment group, $(n\text{-}2)^m$ possible combinations for the third segment group, ..., and 1 combination for the n-th segment group. Therefore, the total number of segment group combinations is as follows:

$$\prod_{k=0}^{n-1}(n-k)^m = (n!)^m \qquad \square$$

Theorem 1 indicates that it is impractical to check all possible segment group combinations to find the best one for large industrial circuits with a large number of scan chains. Therefore, we propose a heuristic two-phase algorithm to efficiently find an optimal segment group combination with low SSA at clock aggressors.

The proposed segment regrouping algorithm is shown in Fig. 11. In Phase 1, a segment group G_{tmp} with the maximum weighted impact is identified, and segments in G_{tmp} are placed into separate groups G_1, G_2, ..., G_n in order to divide the segments in the worst case segment group into discrete groups. Then, in Phase 2, a segment S_{min} is selected such that the union ($G_i \cup S_{min}$) has the minimum weighted impact, and S_{min} is added to G_i. This process is repeated until all segments are selected. This algorithm tries to reduce SSA at clock aggressors by minimizing the weighted impact for each segment group. This way, the clock aggressors of this particular segment group in the affected area can be reduced.

As shown in Fig. 11, in Phase 1 and Phase 2 of the algorithm, segments are selected one at a time and added to a particular segment group. In Phase 1, the segment which maximizes the weighted impact of *IAS* for group G_{tmp} is selected. In Phase 2, the segment which results in the minimum weighted impact of *IAS* of a particular group G is selected for addition to G.

```
Algorithm: Segment_Regrouping{
  INPUT: netlist, impact area, initial segment groups
  OUTPUT: updated segment groups

  n = the number of groups;
  for (i = 1 to n ) {
    $G_i = \varnothing$;
  }

  // Phase 1:
  $G_{tmp} = \varnothing$;
  for (i = 1 to n ) {
    foreach ( unselected segment S ) {
      compute $WI(IAS(G_{tmp} \cup \{S\}))$;
    }
    $S_{max}$ = the segment with the maximum $|IAS(G_{tmp} \cup \{S\})|$;
    // Select $S_{max}$
    $G_i = G_i \cup \{S_{max}\}$;
    $G_{tmp} = G_{tmp} \cup \{S_{max}\}$;
  }

  // Phase 2:
  while ( not all segments are selected yet ) {
    for (i = 1 to n ) {
      foreach ( unselected segment S ) {
        if ( S shares same scanchain
              with at least one segment in $G_i$ ) {
          continue;
        } else {
          compute $WI(IAS(G_i \cup \{S\}))$;
        }
      }
      $S_{min}$ = the segment with the minimum $|IAS(G_i \cup \{S\})|$;
      // Select $S_{min}$
      $G_i = G_i \cup \{S_{min}\}$;
    }
  }
  return { $G_1, G_2, ..., G_n$ };
}
```

Fig. 11 Segment regrouping algorithm.

To find and select the segment with minimum or maximum *WI*, we compute the resulting *WI* for the considered group and all yet-unselected segments. Each segment is selected exactly once and before the selection, *WI* is computed with respect to each yet-unselected segment. Thus, the number of *WI* computations is

$$\sum_{i=1}^{NS} i = \frac{NS(NS+1)}{2}$$

where *NS* is the total number of segments. To compute *WI*, we use optimized set operations (union, intersection) on the pre-computed sets of *IA* and *RCAS* to reduce runtime.

6. Experimental Results

The proposed LCTI-SS scheme was implemented in C language for evaluation. Six large ITC'99 benchmark circuits (*b17* to *b22*) [27] and one industrial circuit (*ck1*) were used in the experiments. Logic synthesis, layout, and transition delay ATPG were conducted by Design

Compiler®, IC Compiler, and TetraMax® from Synopsys®, respectively. Table 1 shows the profile of the circuits and corresponding test sets. The low testability of some of the ITC'99 benchmark circuits causes low fault coverage since no further test point insertion was conducted.

Table 1 Profile of Circuits and Test Sets

Circuit	*# of Gates*	*# of FFs*	*# of Clock Aggressors*	*# of Test Vectors*	*Fault Cov. (%)*
b17	37K	1317	5939	999	76.7
b18	92K	3064	13068	2038	69.7
b19	174K	6130	28268	2763	71.7
b20	19K	430	1841	1514	94.3
b21	19K	430	1811	1509	94.9
b22	28K	645	2812	1913	95.0
ck1	2M	99759	282519	2257	97.4

We prepared various scan configurations with different numbers of scan chains and segments for each circuit. For *b17*, *b20*, *b21*, and *b22*, configurations with 3, 4, and 5 scan chains were used. For *b18* and *b19*, configurations with 10, 30, and 50 scan chains were used. For *ck1*, configurations with 100, 200, and 300 scan chains were used. After that, conventional scan segmentation with 3, 4, and 5 segments were applied for each configuration.

For evaluation, we used the *WSA* metric to estimate SSA. A more precise evaluation, based on electrical or circuit-level simulation, is more accurate but computationally very expensive since hundreds to thousands of shift cycles have to simulated for a single test vector alone. Since WSA has been shown to correlate well with IR-drop [26] and thus IR-drop induced delay, we employed WSA in our experiments. We compared the proposed LCTI-SS scheme with conventional scan segmentation in terms of the weighted impact *WI* and WSA at impact aggressor sets.

Table 2 summarizes the experimental results. The reduction ratio of the maximum and the average weighted impact ("*WI*") and the maximum and the average WSA at impact aggressor sets among segment groups ("*WSA at IAS*") are shown in columns 4 to 7. CPU runtime for segment regrouping ("*CPU* (*sec*)") is shown in column 8. It can be seen that, for over 80% of circuits and scan configurations used in the experiments, targeted maximum WSA at impact aggressor sets were significantly reduced, on average as much as over 10% compared with conventional scan segmentation. The maximum reduction exceeded 25% in the case of *b21*. In addition to the reduction of maximum WSA, average WSA at impact aggressor sets was slightly reduced by 1.1% on average for all experimented configurations. Furthermore, the runtime of the proposed algorithm was relatively short even for the large industrial circuit with 2 million gates. This indicates the scalability of the proposed algorithm.

Fig. 12 shows a more detailed analysis by plotting the maximum and average WSA at the clock aggressor set per test vector for *b21* for the configuration with 5 scan chains and 5 segments in each scan chain. It can be seen that both maximum and average WSA at the clock aggressor set were effectively reduced for all test vectors.

Table 2 Experimental Results

Circuit	*#Chains*	*#Seg-ments*	*Reduction (%)*				*CPU (sec)*
			WI		*WSA at IAS*		
			Max.	*Ave.*	*Max.*	*Ave.*	
b17	3	3	8.3	1.1	11.2	1.8	0
		4	9.0	2.3	9.8	1.7	0
		5	16.1	1.1	16.8	0.9	0
	5	3	0.0	2.4	-5.9	-12.6	0
		4	10.1	4.5	-4.6	-1.2	0
		5	10.3	2.8	12.6	0.0	0.1
	10	3	3.6	4.3	3.9	-1.0	0.1
		4	11.0	5.1	1.2	-2.7	0.1
		5	-6.9	1.5	12.1	0.8	0.1
b18	10	3	5.1	2.8	2.4	-0.7	0
		4	3.3	3.5	0.2	-0.6	0
		5	7.9	4.6	-1.9	0.0	0.1
	30	3	5.7	3.9	13.2	1.2	0.3
		4	10.0	4.1	10.5	1.6	0.6
		5	10.0	3.9	9.2	0.4	0.9
	50	3	0.2	0.4	-2.7	-1.4	1.4
		4	0.3	2.3	1.4	1.9	2.6
		5	6.9	4.0	2.1	1.7	4.1
b19	10	3	8.0	-0.5	3.2	5.6	0
		4	8.6	-1.3	-1.3	4.8	0.1
		5	6.7	-1.7	3.0	5.4	0.1
	30	3	0.7	2.1	6.5	0.9	0.7
		4	5.9	2.9	10.6	0.6	1.2
		5	8.2	5.0	4.8	1.3	1.9
	50	3	0.8	1.6	4.2	1.1	3
		4	0.3	2.8	1.8	0.1	5.1
		5	6.3	3.1	0.3	0.3	8.2
b20	3	3	2.1	3.7	1.6	7.5	0
		4	1.9	-2.6	-2.9	1.7	0
		5	3.4	-0.9	6.7	5.1	0
	5	3	-2.3	7.0	7.1	-2.1	0
		4	9.6	5.8	7.2	-0.3	0
		5	11.4	7.0	1.2	2.4	0
	10	3	2.4	7.1	-1.7	0.9	0
		4	7.3	-1.3	7.5	-2.1	0
		5	10.1	3.7	1.3	4.1	0
b21	3	3	17.4	6.2	7.3	-0.4	0
		4	7.1	1.1	-14.7	-4.5	0
		5	21.2	3.6	10.4	-4.8	0
	5	3	4.3	0.4	6.8	0.3	0
		4	10.3	1.7	6.7	4.0	0
		5	15.9	7.8	15.5	4.9	0
	10	3	3.3	0.6	6.4	0.8	0
		4	13.7	10.4	25.1	7.6	0
		5	8.5	7.7	4.8	6.2	0
b22	3	3	3.8	0.2	-1.2	0.4	0
		4	8.9	-0.4	15.0	0.7	0
		5	15.4	-0.4	18.3	-2.3	0
	5	3	-1.4	-1.1	5.0	2.3	0
		4	4.7	2.9	18.2	1.8	0
		5	7.8	4.9	7.1	1.3	0
	10	3	-1.6	0.7	6.2	-2.2	0
		4	11.1	0.4	13.7	-0.3	0.1
		5	6.9	0.5	-0.9	-1.8	0.1
ck1	100	3	12.2	8.9	0.5	3.1	52.3
		4	3.3	3.9	5.8	2.5	99.4
		5	8.5	5.9	-2.7	0.1	156.5
	200	3	5.2	3.5	11.0	0.6	453.8
		4	5.1	1.4	10.4	2.0	840.4
		5	12.1	3.1	10.3	2.3	1307.3
	300	3	1.9	2.5	4.5	1.1	2112.4
		4	5.8	4.3	15.8	0.3	4046.6
		5	2.5	1.8	9.4	0.5	6423.4
Ave.			5.4	1.5	10.3	1.1	

Fig. 13 depicts the reduction ratio of the maximum weighted impact and the reduction ratio of the maximum WSA at the impact aggressor set and their correlation for all circuits and scan configurations used in the experiments. It can be seen that with the increasing reduction ratio of *WI*, the WSA reduction also tends to increase. There are a few outliers, e.g., for the case of *b21* with 3 scan chains and 4 segments in each scan chain. This indicates that even though the weighted impact has a rather decent correlation with the WSA at impact aggressor sets, a more accurate metric for the segment regrouping algorithm may be needed to further improve the maximum WSA reduction at impact aggressor sets.

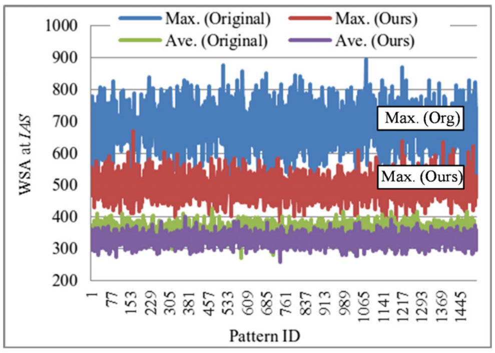


Fig. 12 WSA plot for b21.

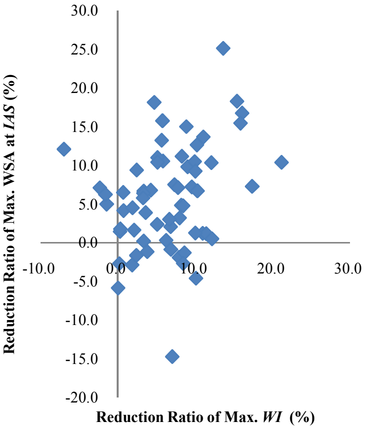


Fig. 13 MAX. *WI* reduction vs. MAX. WSA reduction at *IAS*.

7. Conclusions

This paper is the first of its kind to address the problem of IR-drop-induced shift timing failures by a novel layout-aware scan segmentation scheme, namely *Low-Clock-Tree-Impact Scan Segmentation* (LCTI-SS). The LCTI-SS scheme identifies an optimal combination of scan segments for simultaneous clocking so that shift switching activity in the proximities of clock trees is reduced. This helps reduce IR-drop-induced shift clock skew which is becoming a major cause for scan shift failures, thus helping improve shift safety in scan testing.

Future work to further improve shift safety includes: (1) evaluating whether the LCTI-SS scheme is sufficient to totally avoid shift timing failures by using precise circuit-level power analysis; and (2) finding a metric which correlates more closely with IR-drop than WSA.

Acknowledgments

This work was partly supported by JSPS KAKENHI Grant-in-Aid for Scientific Research (B) 22300017. M. Kochte was a Visiting Researcher at Kyushu Institute of Technology in 2010, supported by the German Academic Exchange Service (DAAD).

References

[1] L.-T. Wang, C.-W. Wu, and X. Wen, Editors, *VLSI Test Principles and Architectures: Design for Testability*, San Francisco: Morgan Kaufmann, 2006.

[2] J. Saxena, et al., "Scan-Based Transition Fault Testing – Implementation and Low Cost Test Challenges," *Proc. IEEE Intl. Test Conf.*, pp. 1120-1129, 2002.

[3] Y. Huang, et al., "Statistical Diagnosis for Intermittent Scan Chain Hold-Time Fault," *Proc. IEEE Intl. Test Conf.*, pp.319-328, 2003.

[4] Y. Wu, "Diagnosis of Scan Chain Failures," *Proc. IEEE Intl. Symp. on Defect and Fault Tolerance*, pp. 217-222, 1998.

[5] P. Girard, "Survey of Low-Power Testing of VLSI Circuits," *IEEE Design & Test of Computers*, Vol. 19, No. 3, pp. 82-92, Feb. 2002.

[6] J. Saxena, K.M. Butler, and L. Whetsel, "An Analysis of Power Reduction Techniques in Scan Testing," *Proc. IEEE Intl. Test Conf.*, pp. 670-677, 2001.

[7] P. Girard, N Nicolici, and X. Wen, Editors, *Power-Aware Testing and Test Strategies for Low Power Devices,* Springer, 2009.

[8] C. P. Ravikumar, M. Hirech and X. Wen, "Test Strategies for Low-Power Devices," *J. of Low Power Electronics,* Vol. 4, No.2, pp. 127-138, Aug. 2008.

[9] J. Saxena, et al., "A Case Study of IR-Drop in Structured At-Speed Testing," *Proc. IEEE Intl. Test Conf.*, pp. 1098-1104, 2003.

[10] X. Wen, et al., "On Low-Capture-Power Test Generation for Scan Testing," *Proc. IEEE VLSI Test Symp.*, pp. 265-270, 2005.

[11] S. Remersaro, et al., "Preferred Fill: A Scalable Method to Reduce Capture Power for Scan Based Designs," *Proc. IEEE Intl. Test Conf.*, Paper 32.2, 2006.

[12] K. Enokimoto, et al., "CAT: A Critical-Area-Targeted Test Set Modification Scheme for Reducing Launch Switching Activity in At-Speed Scan Testing," *Proc. IEEE Asian Test Symp.,* pp. 99-104, 2009.

[13] Y. Yamato et al., "A GA-Based Method for High-Quality X-Filling to Reduce Launch Switching Activity in At-Speed Scan Testing," *Proc. IEEE Pacific Rim Intl. Symp. on Dependable Computing*, pp. 81-86, 2009.

[14] S. Gerstendorfer and H. -J. Wunderlich, "Minimized Power Consumption for Scan-Based BIST," *Proc. IEEE Intl. Test Conf.*, pp. 77-84, 1999.

[15] P. Girard, et al., "A Test Vector Inhibiting Technique for Low Energy BIST Design," *Proc. IEEE VLSI Test Symp.*, pp. 407-412, 1999.

[16] R. Sankaralingam and N. A. Touba, "Reducing Test Power During Test Using Programmable Scan Chain Disable," *Proc. Intl. Workshop on Electronic Design, Test and Applications,* pp. 159-163, 2002.

[17] M.E. Imhof, et al., "Scan Test Planning for Power Reduction," *Proc. Design Automation Conf.,* pp. 521-526, 2007.

[18] Y. Bonhomme, et al., "Efficient Scan Chain Design for Power Minimization during Scan Testing under Routing Constraint," *Proc. IEEE Intl. Test Conf.*, pp. 488-493, 2003.

[19] L. Whetsel, "Adapting Scan Architectures for Low Power Operation," *Proc. IEEE Intl. Test Conf.*, pp. 863-872, 2000.

[20] Y. Bonhomme, et al., "A Gated Clock Scheme for Low Power Scan Testing of Logic ICs or Embedded Cores," *Proc. IEEE Asian Test Symp.*, pp. 253-258, 2001.

[21] P. Girard, et al., "A Modified Clock Scheme for a Low Power BIST Test Pattern Generator," *Proc. IEEE Intl. Test Conf.*, pp. 652-661, 2001.

[22] P. Rosinger, et al., "Scan Architecture With Mutually Exclusive Scan Segment Activation for Shift- and Capture-Power Reduction," *IEEE Trans. Computer-Aided Design*, Vol. 23, No. 7, pp. 1142-1153, Jul. 2004.

[23] T. Yoshida, and M. Watari, "A New Approach for Low Power Scan Testing," *Proc. IEEE Intl. Test Conf.*, pp. 480-487, 2003.

[24] A. Al-Yamani, E. Chmelar, and G. Grinchuck, "Segmented addressable scan architecture," *Proc. IEEE VLSI Test Symp.*, pp. 405- 411, 2005.

[25] E G. Friedman, "Clock Distribution Networks in Synchronous Digital Integrated Circuits," *Proc. of The IEEE*, Vol. 89, No. 5, pp. 665–692, May 2001

[26] K. Noda, et al., "Power and Noise Aware Test Using Preliminary Estimation," *Proc. VLSI Design, Automation and Test, pp. 323-326, 2009.*

[27] IWLS 2005 Benchmarks, *http://www.iwls.org/iwls2005/benchmbenc.html*

[28] X. Wen, et al., "Power-Aware Test Generation with Guaranteed Launch Safety for At-Speed Scan Testing," *Proc. IEEE VLSI Test Symp.*, pp. 166-171, 2011.