九州工業大学学術機関リポジトリ
Kyutacar
Kyushu Institute of Technology Academic Repository

Scan Chain Grouping for Mitigating IR-Drop-Induced Test Data Corruption

著者	Zhang Yucong, Holst Stefan, Wen Xiaoqing, Miyase Kohei, Kajihara Seiji, Qianz Jun
journal or publication title	2017 IEEE 26th Asian Test Symposium (ATS)
year	2018-01-25
URL	http://hdl.handle.net/10228/00007603

doi: info:doi/10.1109/ATS.2017.37

Scan Chain Grouping for Mitigating IR-Drop-Induced Test Data Corruption

Yucong Zhang*, Stefan Holst*, Xiaoqing Wen*, Kohei Miyase*, Seiji Kajihara* and Jun Qian‡

*Kyushu Institute of Technology, Iizuka, 820-8502, Japan
‡Advanced Micro Devices, Inc., Sunnyvale, CA 94088, USA

***Abstract*—Loading and unloading test patterns during scan testing causes many scan flip-flops to trigger simultaneously. This instantaneous switching activity during shift in turn may cause excessive IR-drop that can disrupt the states of some scan flip-flops and corrupt test stimuli or responses. A common design technique to even out these instantaneous power surges is to design multiple scan chains and shift only a group of the scan chains at a same time. This paper introduces a novel algorithm to optimally group scan chains so as to minimize the probability of test data corruption caused by excessive instantaneous IR-drop on scan flip-flops. The experiments show optimal results on all large ITC'99 benchmark circuits.**

Keywords*–*scan testing, switching activity, IR-drop, shift-failure, shift-power mitigation

1. Introduction

In scan testing, shift operations cause switching activity which is usually much higher than the nominal switching activity in functional mode [1–3]. Power delivery networks are usually designed to the demands of the functional operation of a circuit [4] and not to the increased demands in test mode, so power-aware scan test has become essential [5].

The specific problem targeted in this work is test data retention failures caused by excessive, short-term (dynamic) IR-drop at scan cells. This IR-drop arises when too many scan flip-flops switch simultaneously and the resulting power consumption is over-stressing the power delivery network [6, 7]. As soon as the area around a scan flip-flop ff experiences switching activity above a safety threshold, the supply voltage at ff might drop too low and the state of ff may flip. If any of the stimulus bits or test response bits get corrupted in this way, good chips may be falsely declared defective during test causing test-induced yield loss.

Figure 1 shows the situation at a single scan flip-flop ff. The rectangular area around the flip-flop ff is the *aggressor region* which contains all neighboring cells in the layout whose switching activity influence the supply voltage at ff. The so-called *impact areas* around the scan chains contain all the combinational cells structurally reachable by each chain. Whenever a scan chain is shifted, the cells in its impact area potentially switch. As shown in Figure 1 the aggressor region of ff and the impact areas intersect. If all chains are shifted simultaneously, test data in flip-flop ff may be corrupted due to excessive drop of its effective supply voltage v_{ff} caused by high switching activity in its aggressor region. If the chains are shifted at different times, the data in flip-flop ff may remain safe. Shift power safety is achieved only if the switching activities around all individual flip-flops $ff \in F$ in the design are within safety limits, making it a difficult problem to solve.

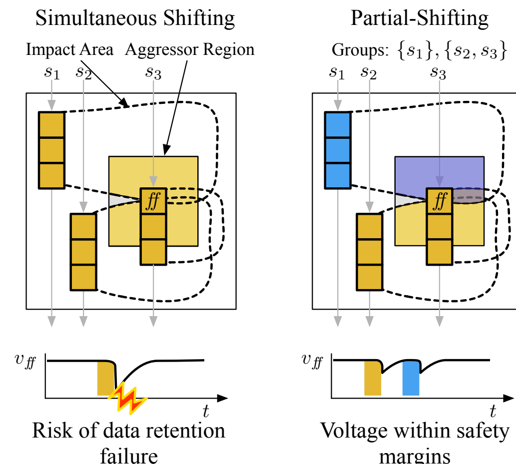

Figure 1. Potential excessive IR-drop at ff caused by simultaneous shifting of two scan chains.

Previous works on reducing shift power can be broadly classified into *test data manipulation* techniques and *design-for-test (DFT) techniques*. Test data manipulation techniques aim to reduce shift switching activity by assigning a special pattern to primary inputs [8], filling don't-care bits to reduce the number of scan cell transitions during shift [9–12], or considering both capture and shift peak power during pattern generation [13]. These techniques can reduce peak shift power effectively and without any hardware overhead. However, they add additional constraints to test pattern generation that often have negative impact on defect coverage, diagnostic resolution, or test data compression. Moreover, as a significant amount of power is consumed by the clock tree itself, shifting even a low-power pattern into all chains simultaneously already consumes a large portion of the available power budget.

Many low-power DFT techniques like output gating [14, 15], partial scan [16], scan segmentation [17], or power-aware test compression [18] are available. Selectively disabling scan chains in combination with test pattern reordering [19] has also been proven very effective for reducing shift power. These works target average power reduction to control heat but they cannot guarantee good results in controlling peak power that leads to excessive IR-drop [20].

Several techniques shift multiple scan chains or scan segments at different times. Scan chain segmentation in combination with clock gating effectively reduces peak shift power [21]. Similar techniques have been proposed to mitigate shift timing errors caused by excessive IR-drop around clock trees [6]. In *multi-duty* scan, multiple staggered clocks are used to shift each chain at slightly different times in order to spread out peak switching activity and therefore avoid excessive IR-drop [7]. The same idea of shift clock staggering was also used at the SoC core level [22]. Furthermore, various algorithms are available to assign flip-flops to scan segments and shift clock domains to reduce peak and average shift power [23, 24].

The common feature of all of these DFT techniques is that they divide a large number of scan chains, segments or cores into a few groups and making sure that only one group is shifted at a time. We call this class of approaches *partial-shift*. Partial-shift adds no additional constraints on pattern generation and also effectively reduces active power originating from clock trees. The core problem of partial-shift is to find the optimal grouping that minimizes the IR-drop at flip-flops while maintaining fast test application time. To the best of our knowledge, all previous works either do not target this problem or approach it with heuristic algorithms that do not guarantee optimal grouping results.

In this paper, we introduce the first algorithm that is able to generate provenly-optimal partial-shift scan chain groupings for all considered benchmark circuits and DFT configurations. We use a flexible grouping cost function that is easily adaptable to the specific demands of real-world designs, it is test pattern independent and is easily integrated with available clock domains. The cost function used in our experiments considers the power consumption of all active circuitry including clock trees, the scan cells themselves and connected logic gates. The generated groupings can then be used with any of the above-mentioned partial-shift implementations. Hardware overheads and design challenges are specific to these DFT implementations and are therefore beyond the scope of this work. The results show that for all benchmarks, maximum possible IR-drop reduction is achieved with as few as 3-5 distinct groups and using more groups would just increase test time without improving shift power safety any further.

The rest of this paper is organized as follows. Section 2 provides a formal model and an analysis of the scan chain grouping problem. Section 3 describes the new scan chain grouping algorithm, and section 4 presents experimental results. Section 5 concludes the paper.

2. Model and Problem Analysis

2.1. Cost Function $d(S')$

Let C be the set of all standard cells (logic cells, sequential cells and clock buffers) in the circuit. Let $F \subset C$ be the set of all scan flip-flops in the circuit.

Each cell $c \in C$ in the circuit is associated with a *weight* $w(c)$ which models the power demand of switching this cell c. Furthermore, each pair of a cell $c \in C$ and a scan flip-flop $ff \in F$ is associated with a *proximity factor* $p(ff, c)$. This proximity factor is a value between 0 and 1 that models the amount of influence the cell c has on the effective supply voltage of the scan flip-flop ff. The weight w and proximity factor p depend on the technology and power delivery network.

Let $C' \subseteq C$ be some subset of switching cells in the circuit. We estimate the amount of IR-drop d experienced at a flip-flop $ff \in F$ by calculating the weighted sum similar to *weighted switching activity* (WSA) [20]:

$$d(ff, C') = \sum_{c \in C'} w(c) \cdot p(ff, c).$$

The cost of switching a subset of cells $C' \subseteq C$ in general is determined by the IR-drop at the most affected scan flip-flop:

$$d(C') = \max\{d(ff, C') | ff \in F\}.$$

Let S be the set of scan chains in the circuit. For a given scan chain $s \in S$, its *impact area* is the subset of cells $C(s) \subseteq C$ for which one of the following conditions is true:

- The cell is a combinational cell that is driven by some flip-flop in scan chain s via a logic path,
- the cell is a clock buffer in the clock tree of chain s, and
- the cell is a flip-flop in scan chain s.

Let $S' \subseteq S$ be a subset of all scan chains. We define

$$C(S') = \bigcup_{s \in S'} C(s)$$

as the union set of all cells related to the given subset of scan chains. Finally, we define the cost function:

$$d(S') = d(C(S'))$$

This cost function is defined over the circuit structure and cell placements. It is independent of specific test pattern data. If fewer scan chains are shifted at the same time, fewer cells can switch and d decreases accordingly.

2.2. Problem Statement

Given a set of scan chains S, the cost function $d : \mathcal{P}(S) \rightarrow \mathbb{R}$, and a number of desired shift groups k, find a partitioning $P = \{S'_1, S'_2, ..., S'_k\}$ over S that minimizes:

$$d(P) = \max\{d(S') | S' \in P\}.$$

From the definition of a partitioning ($\cup_{S' \in P} S' = S$, and $S_i \cap S_j = \emptyset$ for all $S_i, S_j \in P, i \neq j$) follows that when all groups of scan chains are clocked exactly once, a complete shift cycle is executed. The lower the cost $d(P)$, the lower the possibility of experiencing test data corruptions during scan shifting. The decision whether the minimum achievable cost $d(P)$ indeed provides sufficient shift power safety margins can be made by designers or DFT engineers and this paper tries to present a general algorithm.

2.3. Complexity Analysis

We reformulate the problem at hand as a decision problem: Given a set of scan chains S, the cost function $d : \mathcal{P}(S) \to \mathbb{R}$, a cost threshold $d_{\mathrm{th}} \in \mathbb{R}$ and a number of desired groups k. The question is: Can S be partitioned into k subsets such that $d(S'_i) \leq d_{\mathrm{th}}$ for all $S'_i \in \{S'_1, \ldots, S'_k\}$?

Obviously, this decision problem is in NP as any partitioning can be checked by evaluating the cost function d in polynomial time.

We show that this problem is NP-complete for $k \geq 3$ by reducing the problem of graph k-colorability to it. Given a graph $G = (V, E)$ and a positive integer $k \leq |V|$. We set $S = V$, $d_{\mathrm{th}} = 1$ and the following cost function:

$$d(S') = \begin{cases} 2 & \text{when } S' \supseteq E \\ 0 & \text{otherwise} \end{cases}$$

The graph G is k-colorable if and only if there is a partitioning over S with $d(S') \leq d_{\mathrm{th}}$ for all groups S'. The partitioning over S directly corresponds to the color assignment to the graph vertices. Clearly, this transformation can be performed in polynomial time. If there is any polynomial-time algorithm that solves the problem of scan chain grouping for $k \geq 3$, it would also solve graph k-colorability efficiently.

The shown transformation generates problem instances where safety thresholds are exceeded by shifting pairs of scan chains at the same time. The scan chain grouping problem is in fact more general as there may be a subset of three or more scan chains that when all grouped together lead to higher costs than any pair of them. The problem can therefore be reformulated to hypergraph k-colorability. A hypergraph is a simple generalization of graphs where edges can contain more than two vertices.

3. Scan Chain Grouping

As the problem of scan chain grouping is NP-complete, any polynomial-time algorithm is likely to fail in at least some cases. The algorithm proposed here produces optimal solutions for all considered benchmark circuits (see Section 4) at reasonable computational costs.

Its basic principle of operation is as follows. A hypergraph is constructed with one vertex per scan chain and some edges describing groups of scan chains that, when shifted simultaneously, exceed a safety threshold of least one scan flip-flop. A coloring of the graph gives a grouping of scan chains that satisfies the described grouping constraints. Note, however, that the hypergraph is incomplete since not all possible groups of chains that exceed safety thresholds have edges in the graph. Constructing a hypergraph with all possible constraints would be in itself prohibitively expensive because of the large number of possible groups. Instead, the algorithm generates candidate groupings, evaluates these groupings and adds additional edges to the graph for the most costly groups. This loop continues until the chromatic number of the hypergraph becomes larger than the number of available groups.

The initial hypergraph construction forms the first phase of the algorithm and is described in Subsection 3.1. The second and final phase of evaluating candidates and adding constraints is described in Subsection 3.2.

3.1. Hypergraph Initialization

This phase initializes the hypergraph with all constraints (edges) of cardinality 2. In this process, a first bound on the best possible result is established.

The algorithm starts by calculating initial bounds for d. The upper bound $d_{\max}$ is simply $d(S)$ with S being the set of all scan chains. Clearly, no grouping is possible with cost higher than shifting all scan chains simultaneously. The lower bound $d_{\min}$ is the highest cost among all individually shifted scan chains: $\max\{d(\{s\}) | s \in S\}$. As all scan chains need to be shifted eventually, the result cannot be better than $d_{\min}$. Figure 2 shows on the left-hand side an example of a simple circuit with four scan chains consisting of one flip-flop each. The numbers in the intersections between impact areas and aggressor regions are their respective costs. The lower bound in this example is $d_{\min} = 3$ which is the cost s_2 exerts on ff_3. The upper bound is $d_{\max} = 8$ which is the impact of all chains on ff_3.

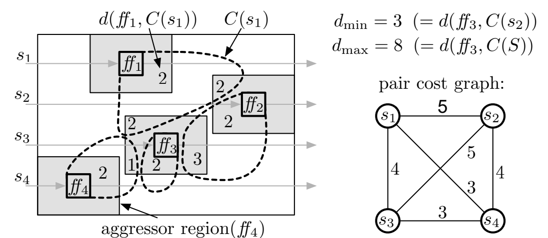

Figure 2. Example for a circuit with 4 chains and one flip-flop per chain.

The algorithm continues by constructing a *pair cost graph*, which is a complete, undirected, edge-weighted graph containing the costs of shifting pairs of scan chains together. Each scan chain is a vertex and each pair of vertices $v_1 \neq v_2$ is connected by an edge with weight $d(\{v_1, v_2\})$. This step requires $\frac{1}{2} \cdot |S|^2$ evaluations of the cost function, which is still possible in a reasonable amount of time. The right-hand side

of Figure 2 shows the pair cost graph for the given example circuit.

Let k be the number of available groups. We now search for the lowest possible $d_{\min'}$ for which a hypergraph that contains only the edges with $d(\{v_1, v_2\}) > d_{\min'}$ remains k-colorable. Since $d_{\min} \leq d_{\min'} \leq d_{\max}$, we can search in logarithmic time by continuously dividing the interval between $d_{\min}$ and $d_{\max}$, construct an unweighted hypergraph and test its colorability. The $d_{\min'}$ found in this way is a new lower bound. The standard problem of graph k-colorability is tackled by converting it into a Boolean formula and using a standard Boolean SAT solver to generate a coloring or prove that no coloring with k colors exists.

We continue the previous example by finding $d_{\min'}$ for $k = 2$. The algorithm first tries $d_{\text{th}} = 6$, then, because the resulting hypergraph (which contains no edges) is 2-colorable, moves on to $d_{\text{th}} = 5$, and $d_{\text{th}} = 4$. The upper half of Figure 3 shows the hypergraph for $d_{\text{th}} = 4$, which is still 2-colorable, and the hypergraph for $d_{\text{th}} = 3$, which is not 2-colorable anymore. The new lower bound here is $d_{\min'} = 4$.

3.2. Guided Search

Let P be the partitioning generated by coloring the initial graph. If $d(P) = d_{\min'}$, then P is already the best possible solution and the algorithm stops at this point. But in most cases we have $d(P) > d_{\min'}$. For example, the actual cost of the initial partitioning shown in Figure 3 is $d(P) = 5$. Thus we need to find another partitioning P' with a lower cost $d(P') < d(P)$. We first identify a group of chains $S' \in P$ that determines the final cost, i.e. $d(S') = d(P)$. Let d_{best} be the cost of the best grouping encountered so far. Next, we generate an irreducible subset $S'' \subseteq S'$ with $d(S'') \geq d_{\text{best}}$ and $d(S''') < d_{\text{best}}$ for any real subset $S''' \subset S''$. This is achieved with a simple greedy procedure that temporarily removes every element in turn and checks the cost. The first irreducible subset of scan chains in Figure 3 is $\{s_1, s_3, s_4\}$.

The irreducible group of chains S'' is now added as a new edge to the hypergraph and the we attempt to color the new graph again with k colors. This process of re-coloring is very efficient with Boolean SAT solving as the the SAT instance and all learned clauses from the previous iterations can be re-used. This loop continues until one of the following two conditions become true: (1) A partitioning with $d(P) = d_{\min'}$ is encountered, or (2) the new graph is not k-colorable anymore. If $d(P)$ reaches the lower bound $d_{\min'}$ like shown in Figure 3, the algorithm terminates with the found optimal partitioning. If the graph becomes uncolorable and the best solution so far has cost $d_{\text{best}} = d(P) > d_{\min'}$, all initial edges with cost less than d_{best} are removed from the hypergraph. The reason is that there may still be a valid partitioning P with $d_{\text{best}} > d(P) > d_{\min'}$, but it requires that some pairs of scan chains s_i, s_j with cost $d(\{s_i, s_j\}) > d_{\min'}$ are being grouped together. The loop continues on the new hypergraph

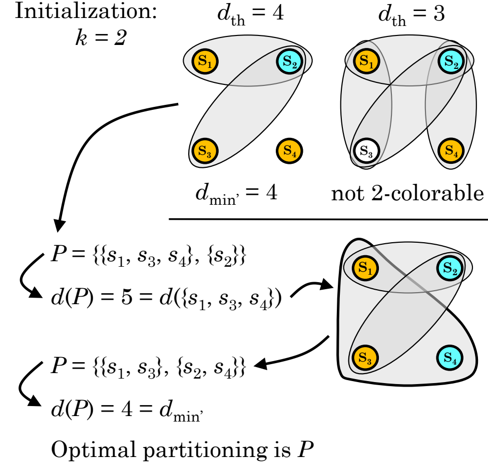

Figure 3. Guided search for the optimal result by iteratively adding necessary constraints to the hypergraph.

without over-constraints until it becomes finally uncolorable.

Since the problem is NP-complete, the loop described above may not terminate for a very long time or the SAT solving behind the graph coloring may time-out. If the algorithm hits a time-out limit, the last temporary solution P with $d(P) = d_{\text{best}}$ is returned as the final result.

4. Experimental Results

The goal of the experiments is to show the effectiveness and scalability of the proposed scan chain grouping approach. The algorithm was implemented in Java and run on Intel(R) Xeon(R) X5690 CPUs at 3.47 GHz. The memory requirements never exceeded 4 GB for any experiment and for easier comparison, the execution time measurements were performed without thread-level parallelism.

The experiments were performed on the six largest ITC'99 benchmark circuits. The benchmark circuits were synthesized, placed and routed using the SAED90nm EDK Digital Standard Cell Library [25] with a standard commercial tool flow and typical operating conditions. We used a standard full-scan test infrastructure with various numbers of parallel scan chains for each benchmark circuit depending on its size. After scan chain insertion, a separate clock tree was generated for each scan chain so that each chain can be shifted independently from the other chains.

Table 1 shows the basic characteristics of the synthesized circuits. Column $|C|$ shows the number of cells in the circuit. Column $|F|$ shows the number of flip-flops among these cells. Column *area* shows the chip area after place and route. Columns $|S|$ and *maxlen* show the number of inserted scan chains and the maximum chain length respectively. We synthesized multiple versions with 10, 30, and 50 scan chains. We did not synthesize the smaller benchmarks with a large number of chains in order to avoid unrealistic configurations with scan chains much shorter than 50 flip-flops.

TABLE 1. BASIC BENCHMARK CIRCUIT STATISTICS

circuit	$\|C\|$	$\|F\|$	*area*	$\|S\|$	*maxlen*
b17	29k	1317	38 mm^2	10	132
				30	44
b18	103k	3020	122 mm^2	10	302
				30	101
				50	61
b19	185k	6042	221 mm^2	10	605
				30	202
				50	121
b20	34k	430	41 mm^2	10	43
b21	34k	430	38 mm^2	10	43
b22	53k	613	60 mm^2	10	62

The different scan chain configurations did not change the number of cells or the circuit area significantly.

To ensure that the results are easily verifiable, we chose rather simple parameters for the cost function. We set $w(c) = 1$ for all cells $c \in C$ in the circuit. The proximity factor $p(\mathit{ff}, c)$ is set to 1 whenever the cell c is located within 7 rows in y-direction and 300 NAND2X1 cell-widths in x-direction of the flip-flop ff. If a cell is outside this rectangular area, the influence factor is set to 0. The size of the aggressor region was chosen rather large to demonstrate our algorithm with a lot of interactions between the scan chains. In practice, these weights can be readily selected based on a more realistic IR-drop model and our approach will work in the same way.

Table 2 compares our results with MD-Scan [7]. The first column shows the circuit name and the number of scan chains. Column d_1 shows the worst-case cost when all scan chains are shifted simultaneously, i.e. the value of $d(S)$. Column d_∞ shows the best-case cost when all scan chains are shifted individually, i.e. the value of $\max\{d(s)|s \in S\}$. As no partitioning method can yield costs outside the bounds of $k = 1$ and $k = \infty$, we use for the remaining columns an efficiency measure relative to these bounds:

$$\mathrm{e_x} = 100\% \cdot \left(1 - \frac{d_x - d_\infty}{d_1 - d_\infty}\right)$$

The efficiency is $e_1 = 0\%$ when the cost equals d_1 and $e_\infty = 100\%$ when the cost equals d_∞. The remaining columns show the results for various numbers of scan chain groups k. For each group count k, sub-column [7] shows the efficiency of random grouping (average over 128 different random groupings). Sub-columns e_{res} show the results of our grouping method, and sub-columns e_{min} show the lowest possible cost with each number of groups. We observe that in all cases our algorithm achieves much better results than the average random groupings. Furthermore, our algorithm was able to prove in all cases that the result is indeed the one with the lowest possible cost, i.e. best possible resilience against IR-drop induced test data corruption. The results also show that spending more than 5 groups would not improve scan shift safety any further. This is an important finding, because test time increases with the number of groups in various DFT implementations. Finally, sub-columns *CPU* show the execution times. As expected, our algorithm runs longer with larger number of scan chains, because the search space is much larger in these cases. Still, all benchmarks were completed within a few minutes.

For validating the new scan chain groupings, they were simulated using random patterns and the maximum observed local WSA around the scan flip-flops were recorded. Table 3 compares a representative random grouping in columns [7] (i.e. a grouping from the 128 used as baseline in Table 2 that had cost closest to the average) to the optimal groupings generated by our algorithm in columns Δ*opt.*. A very few cases show an slight increase in switching activity which is expected, since the switching activity is not uniform across all parts of logic circuits. In almost all cases, the maximum observed switching activity was reduced and therefore the worst IR-drop during shift improved significantly.

5. Conclusions

Partial-shift is a common technique for reducing peak shift power and the chance of IR-drop induced test data corruption. We have proposed the first algorithm that computes optimal groupings of simultaneously shift-able scan chains with the least possible IR-drop impact on scan cells. Although the scan chain grouping problem is NP-complete, our algorithm can produced provenly-optimal groupings within a few minutes for all considered benchmark circuits. The flexible cost function is easily adaptable to the specific demands of real-world designs, it is test pattern independent and is easily integrated with available clock domains. The experiments have shown that maximum possible IR-drop reduction is achieved with as few as 3-5 distinct groups and the benefit of optimal groupings have been confirmed by WSA simulations.

Acknowledgments

This work was supported in part by JSPS Grant-in-Aid for Scientific Research (B) #17H01716, JSPS Grant-in-Aid for Scientific Research on Innovative Areas #15K12003 and JSPS Grant-in-Aid for Scientific Research (C) #17K00081.

References

[1] M. L. Bushnell and V. D. Agrawal, *Essentials of Electronic Testing for Digital, Memory, and Mixed-Signal VLSI Circuits*. Springer, Nov. 2000.

[2] L.-T. Wang, C.-W. Wu, and X. Wen, *VLSI Test Principles and Architectures: Design for Testability (Systems on Silicon)*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., Jun. 2006.

[3] J. Saxena, K. Butler, V. Jayaram, S. Kundu, N. Arvind, P. Sreeprakash, and M. Hachinger, "A case study of IR-drop in structured at-speed testing," in *Proc. IEEE International Test Conference (ITC)*, Sep. 2003, pp. 1098–1104.

TABLE 2. SCAN CHAIN GROUPING RESULTS

circ.	$\|S\|$	d_1	d_∞	$k=2$				$k=3$				$k=4$				$k=5$			
				[7]	e_{res}	$e_{\min}$	*CPU*	*[7]*	e_{res}	$e_{\min}$	*CPU*	*[7]*	e_{res}	$e_{\min}$	*CPU*	*[7]*	e_{res}	$e_{\min}$	*CPU*
b17	10	2054	960	17	58	58	2.0s	25	79	79	1.5s	36	100	100	2.0s	41	100	100	1.5s
	30	2127	1834	60	100	100	8.9s	77	100	100	8.8s	85	100	100	9.0s	89	100	100	9.0s
b18	10	4156	2655	21	44	44	6.7s	31	98	98	4.7s	38	100	100	4.7s	47	100	100	4.8s
	30	4056	2373	11	55	55	1.2m	21	100	100	1.9m	27	100	100	1.2m	37	100	100	1.2m
	50	3918	2953	14	93	93	2.5m	25	100	100	6.1m	38	100	100	6.2m	46	100	100	6.3m
b19	10	4338	3301	18	59	59	12s	26	100	100	12s	31	100	100	12s	33	100	100	11s
	30	4403	3836	20	99	99	1.1m	31	100	100	59s	40	100	100	1.0m	53	100	100	1.0m
	50	4311	2663	12	62	62	8.1m	18	82	82	22m	25	99	99	14m	30	100	100	8.1m
b20	10	2309	2302	29	100	100	0.7s	29	100	100	0.6s	43	100	100	0.8s	57	100	100	0.6s
b21	10	2172	2078	31	91	91	0.7s	49	100	100	0.7s	59	100	100	0.7s	66	100	100	0.7s
b22	10	2879	2665	21	51	51	1.0s	33	100	100	1.0s	40	100	100	1.0s	51	100	100	1.0s

TABLE 3. WEIGHTED SWITCHING ACTIVITY SIMULATION.

circ.	$\|S\|$	$k=2$		$k=3$		$k=4$		$k=5$	
		[7]	Δ*opt.*	*[7]*	Δ*opt.*	*[7]*	Δ*opt.*	*[7]*	Δ*opt.*
b17	10	635	-116	598	-186	653	-290	566	-184
	30	706	-16	633	57	685	5	684	6
b18	10	1470	-127	1365	-369	1425	-431	1008	-14
	30	1802	-158	1771	-359	1680	-268	1514	-102
	50	1823	-8	1612	-108	1523	-19	1730	-226
b19	10	1925	-229	1702	-1	1925	-224	1942	-241
	30	2076	78	1962	-32	1932	-2	2022	-92
	50	1871	-430	1601	-160	1455	-189	1361	-95

[4] M. Pedram, "Power minimization in IC design: Principles and applications," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 1, no. 1, pp. 3–56, Jan. 1996.

[5] M. Tehranipoor and K. M. Butler, "Power supply noise: A survey on effects and research," *IEEE Design & Test*, vol. 27, no. 2, pp. 51–67, March. 2010.

[6] Y. Yamato, X. Wen, M. A. Kochte, K. Miyase, S. Kajihara, and L. T. Wang, "LCTI-SS: Low-Clock-Tree-Impact scan segmentation for avoiding shift timing failures in scan testing," *IEEE Design & Test*, vol. 30, no. 4, pp. 60–70, Aug. 2013.

[7] T. Yoshida and M. Watari, "MD-SCAN method for low power scan testing," in *Proc. IEEE Asian Test Symposium (ATS)*, Nov. 2002, pp. 80–85.

[8] T.-C. Huang and K.-J. Lee, "An input control technique for power reduction in scan circuits during test application," in *Proc. IEEE Asian Test Symposium (ATS)*, Nov. 1999, pp. 315–320.

[9] R. Sankaralingam, R. R. Oruganti, and N. A. Touba, "Static compaction techniques to control scan vector power dissipation," in *Proc. IEEE VLSI Test Symposium (VTS)*, Apr. 2000, pp. 35–40.

[10] R. Sankaralingam and N. A. Touba, "Controlling peak power during scan testing," in *Proc. IEEE VLSI Test Symposium (VTS)*, May 2002, pp. 153–159.

[11] K. M. Butler, J. Saxena, T. Fryars, G. Hetherington, A. Jain, and J. Lewis, "Minimizing power consumption in scan testing: Pattern generation and DFT techniques," in *Proc. IEEE International Test Conference (ITC)*, Oct. 2004, pp. 355–364.

[12] N. Badereddine, P. Girard, S. Pravossoudovitch, C. Landrault, A. Virazel, and H. J. Wunderlich, "Minimizing peak power consumption during scan testing: Structural technique for don't care bits assignment," in *Proc. Ph.D. Research in Microelectronics and Electronics*, Jun. 2006, pp. 65–68.

[13] H. T. Lin and J. C. M. Li, "Simultaneous capture and shift power reduction test pattern generator for scan testing," *IET Computers Digital Techniques*, vol. 2, no. 2, pp. 132–141, Mar. 2008.

[14] S. Gerstendörfer and H.-J. Wunderlich, "Minimized power consumption for scan-based BIST," *Journal of Electronic Testing (JETTA)*, vol. 16, no. 3, pp. 203–212, Jun. 2000.

[15] S. Potluri, A. S. Trinadh, S. B. Ch., V. Kamakoti, and N. Chandrachoodan, "Dft assisted techniques for peak launch-to-capture power reduction during launch-on-shift at-speed testing," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 21, no. 1, pp. 1–25, Dec. 2015.

[16] V. D. Agrawal, K.-T. Cheng, D. D. Johnson, and T. S. Lin, "Designing circuits with partial scan," *IEEE Design & Test*, vol. 5, no. 2, pp. 8–15, Apr. 1988.

[17] L. Whetsel, "Adapting scan architectures for low power operation," in *Proc. IEEE International Test Conference (ITC)*, Oct. 2000, pp. 863–872.

[18] D. Czysz, M. Kassab, X. Lin, G. Mrugalski, J. Rajski, and J. Tyszer, "Low-power scan operation in test compression environment," *IEEE Trans. Comp.-Aided Design of Integr. Circuits and Systems*, vol. 28, no. 11, pp. 1742–1755, Nov. 2009.

[19] R. Sankaralingam, B. Pouya, and N. A. Touba, "Reducing power dissipation during test using scan chain disable," in *Proc. IEEE VLSI Test Symposium (VTS)*, Apr. 2001, pp. 319–324.

[20] P. Girard, N. Nicolici, and X. Wen, Eds., *Power-Aware Testing and Test Strategies for Low Power Devices*. Springer, Oct. 2010.

[21] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "A gated clock scheme for low power scan testing of logic ics or embedded cores," in *Proc. IEEE Asian Test Symposium (ATS)*, Nov. 2001, pp. 253–258.

[22] R. Wang, B. Bhaskaran, K. Natarajan, A. Abdollahian, K. Narayanun, K. Chakrabarty, and A. Sanghani, "A programmable method for low-power scan shift in soc integrated circuits," in *Proc. IEEE VLSI Test Symposium (VTS)*, Apr. 2016, pp. 1–6.

[23] P. Rosinger, B. M. Al-Hashimi, and N. Nicolici, "Scan architecture with mutually exclusive scan segment activation for shift- and capture-power reduction," *IEEE Trans. Comp.-Aided Design of Integr. Circuits and Systems*, vol. 23, no. 7, pp. 1142–1153, Jul. 2004.

[24] Y. C. Huang, M. H. Tsai, W. S. Ding, J. C. M. Li, M. T. Chang, M. H. Tsai, C. M. Tseng, and H. C. Li, "Test clock domain optimization to avoid scan shift failure due to flip-flop simultaneous triggering," *IEEE Trans. Comp.-Aided Design of Integr. Circuits and Systems*, vol. 32, no. 4, pp. 644–652, Apr. 2013.

[25] R. Goldman, K. Bartleson, T. Wood, K. Kranen, C. Cao, V. Melikyan, and G. Markosyan, "Synopsys' open educational design kit: Capabilities, deployment and future," in *Proc. IEEE International Conf. on Microelectronic Systems Education*, Jul. 2009, pp. 20–24.