Transition Delay Fault Test Pattern Generation Considering Supply Voltage Noise in a SOC Design

Nisar Ahmed, Mohammad Tehranipoor
Dept. of Electrical & Computer Engineering
University of Connecticut
{nisar,tehrani}@engr.uconn.edu

Vinay Jayaram
Texas Instruments, Inc.
Dallas, TX
vjayaram@ti.com

Abstract

Due to shrinking technology, increasing functional frequency and density, and reduced noise margins with supply voltage scaling, the sensitivity of designs to supply voltage noise is increasing. The supply noise is much larger during at-speed delay test compared to normal circuit operation since large number of transitions occur within a short time frame. Existing commercial ATPG tools do not consider the excessive supply noise that might occur in the design during test pattern generation. In this paper, we first present a case study of a SOC design and show detailed IR-drop analysis, measurement and its effects on design performance during at-speed test. We then propose a novel method to measure the average power of at-speed test patterns, referred to as switching cycle average power (SCAP). A new practical pattern generation methodology is proposed to generate supply noise tolerant delay test patterns using existing capabilities in commercial ATPG tools. The results demonstrate that the new patterns generated using our technique will minimize the supply noise effects on path delay.

Categories and Subject Descriptors: B.8.1 [Integrated Circuits]: Performance and Reliability-Reliability, Testing and Fault-Tolerance

General Terms: Reliability.

Keywords: supply noise, delay testing, test generation.

1. INTRODUCTION

Shrinking technology along with increase in design density and frequency have posed serious design and test challenges. One important issue of testing today's nanometer high-speed designs is the increasing number of timing-related defects and another issue is power supply noise. Among existing delay test models, transition fault model is widely practiced in industry to test delay-induced defects and is considered as a cost-effective alternative to functional pattern generation [1] [2]. The at-speed launch and capture in addition to large number of switchings in the circuit during transition delay fault testing can cause excessive peak power and result in large IR-drop.

Supply noise (including IR-drop, ground bounce, and Ldi/dt) effects has become more significant in recent years and needs to be efficiently taken into consideration, as it poses design, test and reliability challenges for the chip manufacturers/foundries. This situation has grown more complicated with reducing supply voltage and the limitation of further reduction of threshold voltage. The reduced voltage difference between the VDD and VSS pins of a standard cell reduces the cells operating performance and may result in chip performance reduction if the cell is on a critical path. The IR-drop also reduces the cell's noise immunity and in some cases may lead to functional failures [3] [4].

In order to simplify the pattern generation process, traditionally ATPGs consider zero delay gate model and target as many faults per pattern as possible in order to reduce the test pattern volume. In other words, operating and manufacturing conditions are ignored during ATPG. Patterns generated using such ATPGs may cause large number of transitions in the circuit which may not necessarily occur during functional operation. As a result, a design that may *not* have a delay fault may fail a delay test pattern due to excessive IR-drop related effects. Therefore, new pattern generation methods are required to generate test patterns that reliably distinguish between good and bad chips, i.e. the test patterns should not generate excessive supply noise in the design under test. This issue would be even more problematic when testing system-on-a-chip (SOC) designs where different on-chip blocks generate different IR-drops and in some cases the blocks are tested in parallel to reduce test time. The power consumption must be taken into account [5] [6] and the IR-drop performance degradation effects need to be considered during ATPG.

1.1 Related Prior Work

Launch-off-shift [7], launch-off-capture [8], and enhanced scan [9] are three major scan-based techniques proposed for transition delay fault testing. In all the three methods, a pattern pair (V1, V2) is applied to target delay faults but with different launch mechanisms. Pattern V2 for launch-off-shift, launch-off-capture, and enhanced scan is generated using last shift, functional response, and arbitrary using ATPG, respectively. Various techniques have also been proposed to improve the quality of at-speed test by increasing fault coverage and reducing pattern count, avoiding functionally untestable faults, or reducing scan enable design effort [10] [11] [12].

Several approaches have been proposed for power supply noise analysis and estimation in recent years. Some closed-form equations are derived in [4] to calculate simultaneous switching noise. Estimation of ground bounce, caused by the switching in internal circuitry for deep-submicron circuits, using a scaling model is discussed in [13]. Reference [14] proposes a simulated switching circuit model to estimate PSN which includes IR voltage drop and ΔI noise based on an integrated package-level and chip-level power bus mode. Modeling of PSN on distributed on-chip power networks is described in [15]. ATE and neural network are used to find the patterns generating maximum instantaneous current [16]. The test scheduling for SOC designs considering power consumption is dis-

DAC 2007, June 4–8, 2007, San Diego, California, USA.

Table 1: Design Characteristics

Clock Domains	6
Scan Chains	16
Total Scan Flops	23420
Negative Edge Scan Flops	22
Transition Delay Faults	9651568

cussed in literature [5] [6] to minimize test time while ensuring the SOC test power is lower than functional power threshold.

The issue of overkill during delay test is addressed in [17] and a vector-based approach for power supply noise analysis in test compaction is proposed. A power supply noise model is developed and used during test compaction. The procedure may become slow for large designs since all the patterns are generated without *random-fill* and the power supply noise needs to be estimated in every compaction loop. The method proposed in [18] verifies test vectors for IR-drop failures and identifies failing vectors. The method estimates the average current drawn from power rails and compares it against a pre-defined threshold set by designer. A pattern generation technique is proposed in [19] by building current/voltage libraries to maximize the power supply noise along targeted paths and cause longer propagation delays for the nodes along the paths. The computation complexity of the pattern generation procedure is high since it targets one pattern at a time.

1.2 Contribution and Paper Organization

In this paper, a new method is presented to measure the average power during at-speed test (during fast launch and capture), referred to as *switching cycle average power*. The method considers both length of the paths affected by each pattern and number of transitions occurred during the switching time frame window as opposed to calculating switching power for entire clock cycle in statistical approach. A novel pattern generation procedure taking supply voltage noise into account is proposed for SOC designs ensuring that the supply noise will always be lower than defined threshold. Note that, in this paper, we do not address the shift IR-drop as lower frequencies are used during test pattern shift and is of less concern for test engineers compared to at-speed launch and capture IR-drop. The proposed procedure uses existing commercial ATPG tools. The results show that the new pattern set generated for an industrial-sized SOC design using our proposed procedure significantly reduces the IR-drop and minimizes the performance degradation.

The rest of the paper is organized as follows. Section 2 explains a case study for a SOC design with detailed statistical and dynamic IR-drop analysis for at-speed test pattern application in addition to a new power model to measure the average switching power for at-speed patterns. Section 3 presents pattern generation, SCAP, and pattern validation procedures. This section also shows the experimental results of applying new pattern set to the target SOC. Finally, concluding remarks are in Section 4.

2. CASE STUDY

In this section, firstly we describe the physical design implementation and statistical IR-drop analysis. We, then present a detailed dynamic IR-drop analysis for two types of patterns with different path delay distribution. Also, a new power model is explained to measure the average power of at-speed test patterns which takes both the switching activity and the pattern path delay distribution into account.

2.1 SOC Design and Implementation

In this case study, we experimented with an industrial-strength SOC design (Turbo-Eagle) that had the following characteristics (Table 1). It is a dual-processor SOC [24] and contains a host of peripherals like USB, HDD, VGA and DMA controllers. Figure 1 shows the entire chip floorplan with six major sub-blocks *B1* through *B6*. All these blocks are connected by the AMBA bus from ARM. The design-for-test was implemented hierarchically using full-scan methodology (Synopsys DFT Compiler [22]) with 16 scan chains inserted and approximately $23K$ scan cells. There are six internal clock domains and 22 negative edge scan cells which are placed on a separate scan chain. Due to tight timing requirements, the chip input pins are not registered for coverage improvement. Also, the bi-directional pins were configured to operate in input mode during test mode.

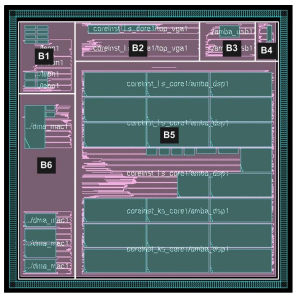

Figure 1: The SOC design floorplan.

The physical design implementation was performed using Cadence SOC Encounter place and route tool [23]. The maximum frequency of the design is $100MHz$ for the master processor. A slow scan shift frequency of 10MHz was used. It is implemented in $180nm$ standard cell library [24] and timing closed at nominal operating voltage ($1.8V$) and temperature (25^oC) conditions. The design contains 37 power (VDD) and ground (VSS) pads each inserted uniformly around the entire chip periphery. The placement and routing of the design was performed hierarchically along with clock-tree synthesis and scan cell ordering to minimize scan chain wirelength. Finally, the empty spaces in the design were inserted with filler cells and metal fill was performed to increase the density of the metal layers, which makes the topology of the layers more uniform.

2.2 Statistical IR-drop Analysis

Before implementing the transition fault test pattern generation, we perform a vector-less statististical IR-drop analysis to estimate the IR-drop during functional operation. There are two reasons why it is important to perform statistical IR-drop analysis. Firstly, the physical design engineer requires it to validate the power/ground network of the design so that it can handle the functional operation activity. Also, it is used to verify design timing with IR-drop effects using a static timing analysis tool. Secondly, the test engineer can use the functional IR-drop or alternatively the related average switching power dissipation to generate IR-drop tolerant delay test patterns.

Although, statistical IR-drop analysis seems a very simple technique but it gets more complicated to get a good IR-drop estimate with several clock domains and blocks in a SOC design. It assumes uniform toggle activity over the entire design and measures IR-drop and average power over the same time frame window which never occurs during real time or test pattern application. Moreover, performing statistical IR-drop analysis at top-level alone does not provide total insight of the individual block-level IR-drop and power consumption.

Our main objective is to identify blocks which consume more power and are likely to observe higher IR-drop during test pattern application. We simplify the process by first splitting the number of scan flip-flops in each individual clock domain and identify the dominant clock domains. A clock domain with high number

Table 2: Clock Domain Analysis

Clock Domain	*#Scan Cells*	*Frequency [MHz]*	*Blocks Covered*
clkA	17966	50	B1 to B6
clkB	1165	100	B1
clkC	1673	50	B3
clkD	724	25	B6
clkE	1560	25	B6
clkF	142	25	B2

of controlled scan flip-flops is referred to as the dominant clock domain. Table 2 shows the number of scan flip-flops in each of the six clock domains. It can be noticed that *clkA* clock domain is the dominant clock domain with approximately $18K$ scan flip-flops. Since transition fault test patterns are mostly generated per clock domain, we then perform block-level statistical IR-drop analysis for only the dominant clock domains (in our design only for *clkA* domain).

The design net parasitics (resistance and capacitance) were extracted using Synopsys STAR-RCXT [22] extraction tool. The worst average statistical IR-drop using vector-less approach for each block was measured for both VDD and VSS network considering 30% net toggle probability during functional operation. In general, designers assume 20% functional toggle activity. But here, we perform analysis for a greater toggle activity rate as later this average switching power threshold will be used to identify and generate delay test patterns. The reason for such a pessimistic analysis is because the switching activity during test is far greater and non-uniform than during functional operation. Also, the average switching power threshold limit determines the number of patterns if test pattern generation procedure (or ATPG) generates IR-drop tolerant patterns. The lower the threshold set implies lesser transition faults detected by each patten and greater number of delay test patterns generated during ATPG process. Note that currently there is no ATPG that takes IR-drop performance degradation effect into account and this work, to the best of authors' knowledge, is first in proposing a solution to the problem.

The IR-drop analysis for the entire cycle provides an underestimation of average IR-drop during functional operation. This is because the tool considers the probability of net toggle activity over the entire cycle period. However, most of the entire switching activity occurs during the early clock cycle period and the switching time frame varies for every pattern. Therefore, to measure the average IR-drop experienced by the transitions, it is important to estimate the average switching time frame. The time span during which all the transitions occur is referred to as the *switching time frame (STW)*. For a transition fault pattern, the maximum path length affected determines this time frame. Note that for different test vectors, the longest path exercised will be different. From our previous experiments on ITC'99 benchmark design (*b19*) during transition fault test patterns application [20], we have seen an *average switching time frame window* close to half the clock cycle period. This shows that the actual average functional power surge observed during an average switching time frame for a pattern is almost twice the measured value during one full cycle period.

Table 3 shows the statistical IR-drop analysis results for the entire cycle period *(Case1)* and average switching time frame *(Case2)*. Also, it shows the average power consumption reported for the two cases measured using Cadence SOC Encounter tool. Several important observations can be derived from this table: 1) Only the average switching power is almost doubled for all blocks when the switching time frame window is halved but not the worst average IR-drop. It is because most of the blocks *B1*, *B2*, *B3*, *B4* and *B6* are smaller and closer to the chip periphery and therefore, the logic in these blocks observe lower IR-drop even though the switching time frame window is reduced. 2) Block *B5* consumes most of the switching power and also observes the highest IR-drop when the switching time frame window was reduced. This shows that more

Table 3: Statistical functional IR-drop analysis results for each block in SOC.

	Case1 (Full cycle period)			*Case2 (Half cycle period)*		
	Avg. Switching Power [mW]	*Worst Avg. IR-drop [V]*		*Avg. Switching Power [mW]*	*Worst Avg. IR-drop [V]*	
		VDD	VSS		VDD	VSS
B1	20.8	0.033	0.033	30.6	0.034	0.034
B2	34.5	0.035	0.036	87.2	0.043	0.044
B3	12.9	0.028	0.028	17.6	0.029	0.029
B4	4.8	0.019	0.019	9.3	0.020	0.020
B5	108.6	0.076	0.076	**204.9**	0.119	0.120
B6	63.8	0.045	0.045	114.6	0.051	0.050
Chip	265.2	0.077	0.077	404.5	0.126	0.125

focus is needed on the switching activity of block *B5* during pattern generation to avoid IR-drop problems. Although, the above analysis might appear over pessimistic but it provides a good estimate of the average IR-drop and identifies the blocks in the design which will experience higher IR-drop during both functional and delay test pattern application. Also, *Case2* provides an average power threshold that can be used to identify high toggle activity transition fault test patterns at a later stage.

2.3 Average Power Model

The IR-drop depends on two factors: 1) the total switching capacitance and 2) the time frame window during which it occurs. Since, dynamic IR-drop analysis for each delay test pattern set is prohibitively expensive, we require a model to identify test patterns which have a high probability of failure due to IR-drop effects, during their application. The *cycle average power (CAP)* [21] is defined as the average power consumed during a single tester cycle. However, it does not factor in the varying time frame window of the entire switching activity for each pattern. Therefore, a pattern with relatively lesser switching activity but with a very short switching time frame window will not be considered as a potential pattern of IR-drop failure by the CAP power model. Therefore, we define a new term referred to as *switching cycle average power (SCAP)* which is the average power consumed by the test pattern during the time frame of the entire switching activity (STW). CAP_j and $SCAP_j$ of jth pattern in a pattern set are calculated by:

$$CAP_j = (\Sigma C_i \times VDD^2)/T$$

$$SCAP_j = (\Sigma C_i \times VDD^2)/STW_j$$

where C_i is the output gate capacitance of gate G_i. Table 4 shows the comparison of average power and IR-drop analysis of a delay test pattern exercising the dominant clock domain *clkA* using the CAP and SCAP model. The pattern was generated by Synopsys TetraMax [22] using launch-off-capture method. The switching time frame window for this pattern was $8.34ns$ and the clock period being $20ns$. It can be noticed that the power surge during the switching time frame window (SCAP) is more than $2X$ compared to the cycle average power (CAP). Also, the worst average IR-drop on *VDD* and *VSS* using CAP model ($0.128V$ and $0.134V$ respectively) reported are within the V_{min} operating conditions for which the IR-drop effect will not be of concern. In general, during sign-off, the design is made sure to work under V_{min} and V_{max} operating conditions. However, with the SCAP model, the worst average IR-drop experienced by the design on *VDD* and *VSS* network is doubled during the switching time window which is significantly higher than average IR-drop obtained from statistical IR-drop analysis.

2.4 Dynamic IR-drop Analysis

In this section, we try to validate the block level average switching power thresholds derived using statistical IR-drop analysis technique (as explained in Section 2.2) as a good measure to identify patterns which might experience high IR-drop. Transition fault pattern set was generated per clock domain with Synopsys TetraMAX

Table 4: Average dynamic power/IR-drop analysis results of a pattern for CAP and SCAP model.

	Avg. Switching Power [mW]		*Worst Avg. IR-drop [V]*	
	VDD	VSS	VDD	VSS
CAP	206.3	224.4	0.128	0.134
SCAP	457.5	451.6	0.272	0.275

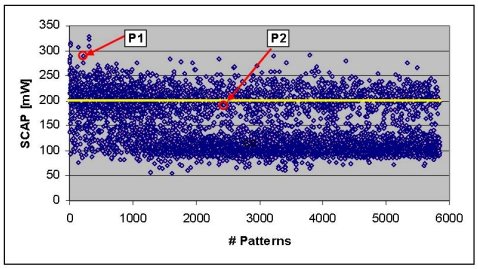

Figure 2: SCAP measured in block *B5* for transition fault pattern set in *clkA* domain.

[22] tool using *random-fill* for don't-care bits which increases fortuitous detection of un-modeled faults. However, random-fill increases the switching activity significantly and the actual IR-drop during transition fault test pattern application is much higher compared to statistical IR-drop. Also, the switching time frame window is much smaller as most commercial tools target easy-to-find paths for fault activation/propagation rather than longer paths through the target fault sites. During ATPG, the primary inputs are held constant and primary outputs are not measured due to low-cost tester speed limitations. For the dominant clock domain *clkA* which was our major concern, 5846 delay test patterns were generated.

The switching cycle average power (SCAP) value for each pattern was measured at both block-level and top-level using gate level timing simulations with Synopsys VCS [22] simulator and programming language interface (PLI) which will be explained in detail later in our pattern validation techniques (Section 3.2). Figure 2 shows the measured SCAP value for the patterns in block *B5* which was observed to be the most power consuming block during statistical IR-drop analysis (Section 2.2). It can be seen that a large number of patterns (approximately 2253 patterns) exercise SCAP value greater than the average switching power threshold ($204mW$) determined during statistical IR-drop analysis for block *B5* (see Table 3). To show the relationship between the measured SCAP value and related IR-drop in the design, we selected two patterns *P1* and *P2*. As shown in Figure 2, pattern *P1* experiences very high SCAP whereas the SCAP value for *P2* is close to the threshold limit. The measured VDD network SCAP value in block *B5* was $283.5mW$ and $190.7mW$ for pattern *P1* and *P2*, respectively. The maximum path delay (STW) for pattern *P1* and *P2* were $7.8ns$ and $8.6ns$, respectively and the clock period being $20ns$. Therefore, the switching time frame window (STW) for these patterns does not vary significantly and pattern *P1* observes very high switching activity.

To measure the IR-drop of a pattern, the switching activity inside the design was captured in the standard value change dump (VCD) format during gate-level timing simulation. The timing information of the gates and the extracted parasitic interconnect delay information was back-annotated using the standard delay format (SDF) file. The switching activity information (VCD file) along with physical design and technology library information is used by SOC Encounter tool [23] to estimate the dynamic IR-drop of the pattern.

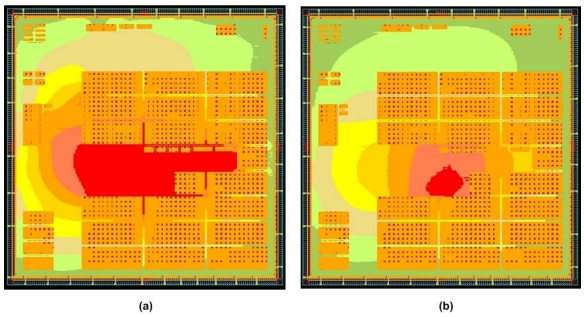

Figure 3: VDD IR-drop plots using SCAP model for two patterns (a) *P1* and (b) *P2*.

Figure 3 shows the IR-drop plots using SCAP model on the VDD network for both of these patterns during launch-to-capture clock cycle. The red region shows the portion of the design where the voltage drop is greater than 10% VDD ($0.18V$). The worst average IR-drop was reported to be $0.28V$ and $0.19V$ for pattern *P1* and *P2*, respectively.

3. EXPERIMENTAL RESULTS

Based on our analysis in previous section, we have seen that a large percentage of the patterns in the test set generated using random-fill option exercise higher switching activity. Also, block *B5* observes higher switching cycle average power which relates to high IR-drop. Ideally, we would like to have isolation logic for block *B5* to avoid switching activity while testing other blocks. For example, to place block *B5* in shift mode and apply constant values to avoid any activity. Since, we do not have any such DFT logic, our major challenge is how we can use the existing ATPG tools capability to generate IR-drop tolerant patterns.

3.1 Pattern Generation

To generate lower IR-drop delay test patterns, an ATPG tool must have some kind of an option to limit the maximum number of faults targeted by a pattern in each block to keep the switching activity lower. However, there is no such option in the tool but Synopsys TetraMax provides three types of don't-care fill options for low power pattern generation: *Case1: fill-0*, causes all don't-care scan cells to be filled with 0's, *Case2: fill-1*, causes all don't-care scan cells to be filled with 1's and *Case3: fill-adjacent*, causes don't-care scan cells to be filled with the value of the first adjacent scan cell with a defined/care value. Note that other previously proposed low power pattern generation techniques could also be used. *Case3* is mostly useful to minimize power usage during scan shifting by reducing signal switching at the expense of higher pattern count. However, in our experiments we are trying to reduce the switching activity between the launch and capture window of the launch-off-capture patterns. We have tried all three don't-care fill options, but in this work we'll only provide the results of *fill-0* option which provided the best results.

Although, we use *fill-0* option to reduce the switching activity, there was another problem which we observed during pattern generation. The number of don't-care bits is very low in the initial set of patterns and increases significantly later. This is because the ATPG tool tries to target most of the faults in the first few generated patterns. Therefore, if we target faults in all the blocks simultaneously, the initial set of generated patterns will still observe high SCAP as they try to detect most of the faults in each block. As a workaround, we provided the ATPG tool with target faults only in a subset of blocks. For example, faults in blocks *B1*, *B2*, *B3* and *B4* can be targeted simultaneously as they observed the least IR-drop. In this case, the generated patterns will have very high don't-care

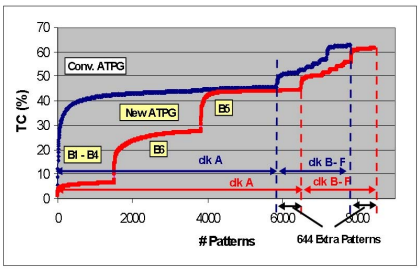

Figure 4: Test coverage curves for conventional ATPG and the new pattern generation procedure.

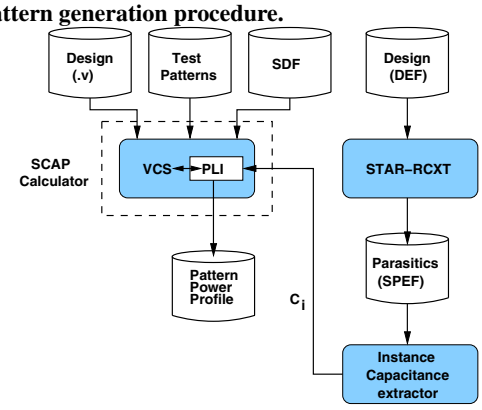

Figure 5: SCAP calculator using Synopsys VCS simulator.

bits in the remaining blocks *B5* and *B6* and the tool will use *fill-0* in these blocks, thereby reducing power dissipation in them. Moreover, this procedure is only applied for the dominant clock domain (*clkA*). A more ideal scenario would be that the ATPG tool provides different fill options for don't-care bits in different blocks. This would allow us to generate patterns in some blocks with random options yet keep the switching activity in other blocks to a minimum.

Figure 4 shows the test coverage curves for two pattern sets generated by conventional method and the new pattern generation procedure. It can be noticed that in the new technique for *clkA* domain, the ATPG process was divided into three steps: *Step1:* In this step the ATPG tool was provided with fault list for blocks *B1* through *B4* only, *Step2:* fault set in block *B6* alone targeted and *Step3:* fault set in block *B5* only is targeted. In each step, *fill-0* option was used for don't-care cells. For *clkA*, the new technique generated slightly higher number of patterns (644 extra patterns) compared to conventional random-fill pattern set. For the remaining clock domains, the ATPG is similar in both the methods.

3.2 Pattern Validation

• SCAP Calculator:

To determine the switching cycle average power (SCAP) of each pattern in the transition fault pattern set, we need the following information: 1) the gates switching inside the circuit, 2) output capacitance of each gate and 3) the switching time frame window. Simulation-based techniques can be used to capture the switching activity information in the standard *value change dump (VCD)* format. But, this technique is sufficient only to analyze a very small number of patterns due to the extremely large size of VCD files for large designs.

To overcome this problem, we use programming language interface (PLI) routines during gate-level verilog simulation. The PLI provides a standard interface to the internal data representation of the design during simulation. Figure 5 shows the SCAP calculation

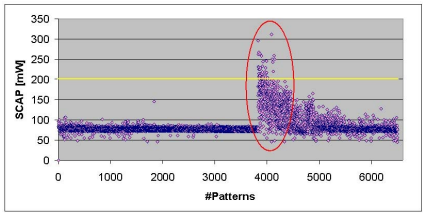

Figure 6: Switching cycle average power (SCAP) measured on VDD network for each pattern in the new test pattern set (*clkA* domain pattern set).

flow. The capacitance per each gate instance is extracted from the RC parasitics file (*Standard parasitics exchange format (SPEF)*) generated using Synopsys STAR-RCXT extraction tool. We have developed a PLI which can be plugged into Synopsys VCS gate level simulator which acts as SCAP calculator during simulation. It reports the SCAP value for VDD and VSS power network for each pattern during the launch-to-capture window in the launch-off-capture transition fault pattern set. The above procedure using PLI interface avoids the VCD file generation for estimation of switching power.

As we noticed that in *clkA* domain, block *B5* experienced very high switching cycle average power for random-fill, here we again measure the SCAP in it for the new test pattern set generated. Figure 6 shows the measured SCAP value in the VDD network of block *B5* for the new test patterns in *clkA* domain only (6490 patterns). It can be seen that the initial patterns (up to approximately 4000 patterns) have very low and nearly the same SCAP value. This is because these patterns target faults in other blocks and *fill-0* option maintains block *B5* in a quiet state with very less switching activity. Another important point to notice is that there is a sudden disturbance and a high switching activity is observed in the later patterns when the ATPG tool targets the faults in block *B5*. This shows that the ATPG tool applies greedy algorithm to target as many faults in block *B5* even with *fill-0* option and it is unaware of the power consumption. However, the number of patterns above the SCAP threshold are extremely low (approximately 57 patterns) using the new pattern generation technique compared to random-fill with an increase of approximately 8% in test pattern count.

• Simulation with IR-drop Effects:

To further validate some of the delay test patterns exercising long paths, we wanted to perform pattern simulation including IR-drop effects. However, this requires transistor-level simulation with the power/ground network parasitics which is not a feasible solution for a large design. In general, presently during test pattern sign-off, the patterns are simulated at the best and worst-case corners. This is either over optimistic or pessimistic as we apply the corner conditions to all the portions of the design which is not the case as seen in Figure 3. In order to take advantage of relatively faster gate-level simulation but still take IR-drop effects into account, we have developed another PLI which can be plugged into Synopsys VCS gate level simulator which modifies the cell delays during a pattern simulation based on the voltage of every instance reported during dynamic IR-drop analysis using SOC Encounter tool for the respective test pattern. The cell delay degradation is calculated by the following formulation:

$$ScaledCellDelay = Delay \times (1 + k\_volt * \Delta V)$$

where *k_volt* is a factor that accounts for non-linear delay scaling model and it is specified in the vendor supplied technology library. Here, we used a value of 0.9 for *k_volt*, which means for a 5% cell

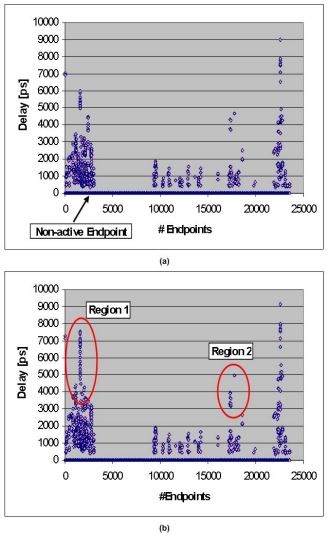

Figure 7: Path delay variation of a test pattern in two cases: (a) no IR-drop effects and (b) scaled cell delays due to IR-drop effects.

voltage decrease ($\Delta V = 0.1V$), the cell delay increases by 9%.

Figure 7 shows the delay observed at each endpoint of the design for a single test pattern in two cases: *Case1:* no IR-drop effect and *Case2:* scaled cell delays due to IR-drop effects. The pattern was selected with most faults being tested in block *B5* (circled region in Figure 6) but with the SCAP value below the threshold limit. An observation point at the end of a path (in our case, scan flip-flop) is referred to as an *endpoint*. An endpoint which does not observe a transition, referred to as *non-active*, is represented with zero delay. It can be seen that the delay observed by a certain number of scan flip-flops (*Region 1*) has increased (upto 30% in some cases). This is because the gates in the input logic cone of these scan flip-flops observed high IR-drop and the delay scaling factor was high for them. Also, we noticed that these flops were in the higher IR-drop region in block *B5*. Also, the delay observed by some endpoints has reduced *(Region 2)*. This is because we measure the path delay observed at each endpoint based on the reference clock signal reaching the respective endpoint. The clock reaches different endpoints at different times due to clock skew and cell delay scaling due to IR-drop effects. Therefore, if the clock signal reaching the capture flip-flop is delayed relatively to the clock signal of launch flip-flop due to IR-drop effects, then the path delay measured at the capture flip-flop decreases.

This kind of pattern simulation is more practical and gives a better insight of the endpoints and paths effected by IR-drop in each pattern. However, the above analysis is still very computationally expensive and it requires two simulations, one to generate VCD file for dynamic IR-drop analysis and then use the reported instance voltages for the next simulation including cell delay degradation effects. Hence, we prefer to apply this technique for only a limited set of patterns or to debug any pattern which is identified to fail due to IR-drop effects.

4. CONCLUSION

In this paper, we have presented a new model called SCAP to measure the impact of a transition delay fault test pattern on supply noise and its negative effect on performance. The proposed model and pattern generation procedure were implemented on an industrial-sized SOC design. The comparison between the new pattern set and and the one generated using conventional ATPG shows that the supply noise significantly decreases in a cost of slight increase in pattern count. The proposed procedure uses existing commercial ATPGs and wrapper is added around them to generate new IR-drop tolerant pattern set.

5. ACKNOWLEDGEMENTS

This work was supported in part by Semiconducto Research Corporation under contract SRC 2006-TJ-1455 and 2007-TJ-1587.

6. REFERENCES

[1] J. Saxena, K. M. Butler, J. Gatt, R. Raghuraman, S. P. Kumar, S. Basu, D. J. Campbell, J. Berech, "Scan-Based Transition Fault Testing - Implementation and Low Cost Test Challenges," in Proc. *International Test Conference (ITC'02)*, pp. 1120 - 1129, Oct. 2002.

[2] X. Lin, R. Press, J. Rajski, P. Reuter, T. Rinderknecht, B. Swanson and N. Tamarapalli, "High-Frequency, At-Speed Scan Testing," *IEEE Design & Test of Computers*, pp. 17-25, Sep-Oct 2003.

[3] M. Nourani, M. tehranipoor and N. Ahmed, "Pattern Generation and Estimation for Power Supply Noise Analysis," in proc. *VLSI Test Symo. (VTS'05)*, pp. 439-444, 2005.

[4] R. Senthinatharr and J. L. Prince, *Simultaneous Switching Noise of CMOS Devices and Systems*, Kluwer Academic Publishers, 1994.

[5] K. Chakrabarty, "Test scheduling for core-based systems using mixed-integer linearprogramming," in proc. *IEEE Trans. on Computer-Aided DEsign of Integrated Circuits and Systems*, Vol. 19, No. 10, 2000, pp. 1163-1174.

[6] Y. Huang, et. al, "Resource Allocation and Test Scheduling for Concurrent Test of Core-Based SOC Design," in proc. *IEEE Asian Test Symposium (ATS'01)*, pp. 265-270, 2001.

[7] J. Savir, "Skewed-Load Transition Test: Part I, Calculus," in Proc. *Int. Test Conf. (ITC'92)*, pp. 705-713, 1992.

[8] J. Savir and S. Patil, "On Broad-Side Delay Test," in Proc. *VLSI Test Symp. (VTS'94)*, pp. 284-290, 1994.

[9] B. Dervisoglu and G. Stong, "Design for Testability: Using Scanpath Techniques for Path-Delay Test and Measurement," in Proc. *Int. Test Conf. (ITC'91)*, pp. 365-374, 1991.

[10] X. Liu and M. Hsiao, "Constrained ATPG for Broadside Transition Testing," in Proc. *IEEE Int. Symp. on Defect and Fault Tolerance in VLSI Systems (DFT'03)*, pp. 175-182, 2003.

[11] Z. Zhang, S. M. Reddy, and I. Pomeranz, On Generating Pseudo-Functional Delay Fault Tests for Scan Designs, in proc. *IEEE Intl. Symposium on Defect and Fault Tolerance in VLSI Systems (DFT'05)*, pp. 398405, 2005.

[12] N. Ahmed, C.P. Ravikumar, M. Tehranipoor and J. Plusquellic, "At-Speed Transition Fault Testing With Low Speed Scan Enable," in proc. *IEEE VLSI Test Symposium (VTS'05)*, pp. 42-47, 2005

[13] Y.-S. Chang, S. K. Gupta and M. A. Breuer, "Analysis of Ground Bounce in Deep Sub-Micron Circuits," in Proc. *IEEE VLSI Test Symposium (VTS'97)*, pp. 110-116, 1997.

[14] H. H. Chen and D. D. Ling, " A Power Supply Noise Analysis Methodology for Deep Submicron VLSI Chip Design," in Proc. *CM/IEEE Design Automation Conference (DAC'97)*, pp. 638-643, 1997.

[15] L. Zheng, B. Li, and H. Tenhunen, "Efficient and Accurate Modeling of Power Supply Noise on Distributed On-Chip Power Networks," in Proc. *Int. Symposium on Circuits and Systems (ISCAS'00)*, pp. 513-516, 2000.

[16] E. Liau and D. Landsiedel, "Automatic Worst Case Pattern Generation Using Neural Networks & Genetic Algorithm for Estimation of Switching Noise on Power Supply Lines in CMOS Circuits," in Proc. *European Test Workshop (ETW'03)*, pp. 105 -110, 2003.

[17] J. Wang, Z. Yue, X. Lu, W. Qiu, W. Shi, D. M. H. Walker, "A Vector-based Approach for Power Supply Noise Analysis in Test Compaction," in Proc. *Int. Test Conf. (ITC'05)*, 2005.

[18] A. Kokrady and C.P. Ravikumar, "Static Verification of Test Vectors for IR-drop Failure," in Proc. *Int. Conf. on Computer-Aided Design (ICCAD'03)*, pp. 760-764, 2003.

[19] A. Krstic, Y. Jiang and K. Cheng, "Pattern Generation for Delay Testing and Dynamic Timing Analysis Considering Power-Supply Noise Effects," in *IEEE Transactions on CAD*, vol. 20, nNo. 3, pp. 416-425 March 2001.

[20] N. Ahmed, M. Tehranipoor and V. Jayaram, "A Novel Framework for Faster-than-at-Speed Delay Test Considering IR-Drop Effects," to appear in *Int. Conf. on Computer-Aided Design (ICCAD'06)*, 2006.

[21] J. Saxena, K. M. Butler, V. Jayaram, S. Kundu, N. V. Arvind, P. Sreeprakash and M. Hachinger, "A Case Study of IR-Drop in Structured At-Speed Testing," in Proc. *International Test Conference (ITC'03)*, pp. 1098 - 1104, Oct. 2003.

[22] Synopsys Inc., "User Manuals for SYNOPSYS Toolset Version 2005.09," Synopsys, Inc., 2005.

[23] Cadence Inc., "User Manuals for Cadence Encounter Tool set Version 2004.10," Cadence, Inc., 2004.

[24] http://crete.cadence.com, 0.18μm standard cell GSCLib library version 2.0, Cadence, Inc., 2005.