This is a copy of the author’s accepted version of a paper subsequently published in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 34 (21), pp. 2013-2024.

It is available online at:

https://dx.doi.org/10.1109/TCAD.2015.2446939

© 2015 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

The WestminsterResearch online digital archive at the University of Westminster aims to make the research output of the University available to a wider audience. Copyright and Moral Rights remain with the authors and/or copyright owners.

Whilst further distribution of specific materials from within this archive is forbidden, you may freely distribute the URL of WestminsterResearch: (http://westminsterresearch.wmin.ac.uk/).

In case of abuse or copyright appearing without permission e-mail repository@westminster.ac.uk
DFT Architecture with Power-Distribution-Network Consideration for Delay-based Power Gating Test

Vasileios Tenentes, Member, IEEE, Saqib Khursheed, Member, IEEE, Daniele Rossi, Member, IEEE, Sheng Yang, Member, IEEE, and Bashir M. Al-Hashimi, Fellow, IEEE

Abstract—This paper shows that existing delay-based testing techniques for power gating exhibit both fault coverage and yield loss due to deviations at the charging delay introduced by the distributed nature of the power-distribution-networks (PDNs). To restore this test quality loss, which could reach up to 67.7% of false passes and 25% of false fails due to stuck-open faults, we propose a design-for-testability (DFT) logic that accounts for a distributed PDN. The proposed logic is optimized by an algorithm that also handles uncertainty due to process variations and offers trade-off flexibility between test-application-time and area cost. A calibration process is proposed to bridge model-to-hardware discrepancies and increase test quality when considering systematic variations. Through SPICE simulations, we show complete recovery of the test quality lost due to PDNs. The proposed method is robust sustaining 80.3% to 98.6% of the achieved test quality under high random and systematic process variations. To the best of our knowledge, this paper presents the first analysis of the PDN impact on test quality and offers a unified test solution for both ring and grid power gating styles.

Index Terms—power gating, dft, power-distribution-network, test quality, grid style, ring style, systematic variations

I. INTRODUCTION

Power gating is a low power design technique for integrated circuits (ICs) that assures the viability of high performance and energy efficient electronic devices at sub-100-nm CMOS technologies [1]. It utilizes transistors as power-switches of logic blocks supply voltage to reduce leakage power and power consumption during periods of inactivity. Power switches are susceptible to defects and their high quality testing is crucial for the efficient low power performance of power-gated ICs, for silicon debugging, for yield analysis and for improving subsequent manufacturing cycle [2]–[5]. Design-for-testability (DFT) is a design technique for assuring the quality of testing of ICs for physical defects during their lifetime from the manufacturing to the field. It consists of fault models that mimic the behavior of physical defects and DFT logic structures that provide the engineering means to apply the tests and collect back their responses.

Power switches are implemented as header or footer switches in either fine-grain or coarse-grain design styles. A fine-grain style incorporates a power switch within each logic cell simplifying power gating synthesis through existing EDA tools [6]. However, the coarse-grain design style is more popular and the focus of this work, since it requires less silicon and offers higher robustness against process variations.

Coarse grain power gating is implemented in two different design styles by deploying either a ring or a grid network of power switches. In ring style [6], power switches are placed at a ring externally to the power-gated block (Figure 1(a)). In grid style [6], [7], power switches are distributed throughout the power-gated region (Figure 1(b)) forming a grid between the power-distribution-networks (PDNs): the supply voltage $V_{dd}$, PDN (SPDN) and the virtual voltage $V_{dd}$, PDN (VPDN). When comparing these two styles [6], the ring is the only option for power gating IP blocks, while the grid style is the only one scalable to large designs and the only option that supports state retention. This paper considers both styles.

Power switches may operate in two low power modes which provide a trade-off between leakage power saving and wake-up time: complete power-off mode (higher leakage power saving) and intermediate power-off mode (lower wake-up time). Recent research has reported a number of DFT solutions to test power switches when considering the stuck-open [8]–[12] and the stuck-short [13], [14] fault models. Stuck-short faults produce a conducting path between $V_{dd}$ and ground and testing against them is crucial to sustain the low power consumption benefits of power gating. Stuck-shorts impact the steady state current at power-off mode and could be detected by an $I_{DDQ}$ based method. Digital-based DFT for monitoring the voltage level of power switches at intermediate mode steady-state have been recently proposed [13], [14]. Stuck-open faults model a defect where the drain or source of a transistor is disconnected. Their testing is crucial for assuring that the power-gated domain will not suffer from small delays due to power-grid IR-drop. In this paper, we target the stuck-
opens through measuring the power-off to power-on delay.

Although previous works have considerably advanced the DFT techniques for power switches, they rely on lumped RC models of the PDNs without considering their distributed nature. It was shown in [16] that, at the grid style power gating, this simplification interacts with the test result and in [17] that could even influence the diagnosis result. In Section II, we consider a distributed model for the RC components of the supply voltage PDN (SPDN), the ground voltage PDN (GPDN) and the virtual voltage PDN (VPDN). We examine both the ring and the grid power gating styles, shown in Figure 1(a) and in Figure 1(b), respectively. Based on this setup, in Section III, we show that the lumped model shortcut used by the state-of-the-art [8], [9], [15] may lead to both fault coverage loss and yield loss that may reach up to 67.7% and 25%, respectively, and we analyze the reasons of this test quality loss. To tackle this problem, Section IV presents a DFT architecture that considers a distributed PDNs model and restores the test quality (fault coverage and yield) at low cost. Its overhead is optimized by an algorithm that offers trade-off flexibility between test-application-time (TAT) and area cost. In Section V we adapt the proposed DFT design method to handle uncertainty and we propose a calibration method from post-silicon measurements that also handles systematic variations. Section VI evaluates the performance and presents the trade-offs of the proposed method and Section VII concludes the paper.

II. STATE-OF-THE-ART & DISTRIBUTED PDNs

Figure 2(a) presents the state-of-the-art DFT architecture for delay-based testing against stuck-open faults on header power switches [8], [9], [15]. The power switches are clustered in m segments-under-test (SUTs) of segment-size L power switches [9]. The test process, shown in Figure 2(b), starts with the initialization phase, during which the control logic fully

discharges the \( V_{dd} \) node by using the discharge transistors [8]. During the application phase, a single SUT \( S_i \) is awakened by the control logic by asserting the sleep signal. Upon the capture moment, the NAND gate logic output is captured at the “test result” flip-flop by asserting the test clock [15], the frequency of which depends on the segment size L. The captured value indicates whether the observation point \( V_{dd} \) was sufficiently charged at the capture moment. Test clock frequency is selected based on the observable charging delay \( M \) of the \( V_{dd} \) point, hereafter referred to as observable wake-up time or simply observation \( M \). This delay is the elapsed time from the start of the application phase to the capture

![Fig. 2](image-url)

![Fig. 3](image-url)

**TABLE I**

<table>
<thead>
<tr>
<th>style &amp; model PDN</th>
<th>ethernet</th>
<th>virtual supply</th>
<th>ground</th>
<th>s38417</th>
<th>virtual supply</th>
<th>ground</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lump</td>
<td>R (Ω) max</td>
<td>C (F) max</td>
<td>count</td>
<td>R (Ω) max</td>
<td>C (F) max</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4.9E-08</td>
<td>9.4E-12</td>
<td>90859</td>
<td>5.9E-23</td>
<td>9.7E-12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.9E-08</td>
<td>1.2E-11</td>
<td>109339</td>
<td>4.1E-23</td>
<td>1.6E-11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.4E-07</td>
<td>1.9E-13</td>
<td>27540</td>
<td>2.0E-07</td>
<td>3.5E-13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8.5E-07</td>
<td>5.5E-12</td>
<td>39255</td>
<td>4.3E-07</td>
<td>3.3E-12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.3E-07</td>
<td>2.3E-12</td>
<td>4.9E-08</td>
<td>1.1E-07</td>
<td>2.1E-12</td>
</tr>
</tbody>
</table>

![Diagram](image-url)
moment, when the transient voltage at the NAND gate output reaches logic-0 value under the fault-free scenario. The voltage level of $\leq 0.2V_{dd}$ is used at the illustrations of the paper. However, this point is within 20%-80% of $V_{dd}$ [18]. Note that based on the PDNs lumped models shown in Figure 2(a), the observable charging delay is computed the same for all SUTs.

We analyzed a large number of benchmarks from the IWLS’05 benchmark suite [19] and selected three representatives: the ethernet, the s38417 and the s38584 benchmark circuits. These circuit comprises 157.5K, 30.5K and 26.9K gate equivalents, respectively, with a gate equivalent corresponding to a two input NAND gate. To generate the RC distributed model of SPDN, GPDN and VPDN, we synthesize the circuits using a 90nm library and operational voltage of $V_{dd} = 1.2V$ for both ring and grid power gating styles using header power switches. The constraint set during the physical synthesis of the PDNs is to achieve $\leq 10\%$ IR drop for the ring and $5\%$ for the grid style using 2048 power switches for the ethernet and 512 power switches for the s38417 and s38584 circuits. This leads to similar power rails size for the two styles. Then, using Synopsys STAR-RCXT, we extract a SPICE model for each style, shown in Figure 3(a) for ring and in Figure 3(b) for grid style, that includes both the nets and the power distribution networks of the design. GPDN is omitted from the Figure for clarity. Table I shows the RC elements information for the distributed and lumped models of the ethernet and the s38417 circuits. For the distributed model the number of R and C elements (count) and their range of value ([min, max]) is shown. The R and C values of the lumped model were computed assuming that the elements of the PDNs are connected in parallel ($C = C_1 + C_2 + \ldots + C_N$ and $1/R = 1/R_1 + 1/R_2 + \ldots + 1/R_N$). Note that the high number of distributed RC elements for the distributed PDNs of the ethernet (more than 350 thousands) imply that their spatial effect should be considered for delay measurements. Thus, we cluster the power switches on both ring (Figure 3(a)) and grid (Figure 3(b)) power gating styles into SUTs according to a layout-driven approach: power switches that are closer to each other are assigned to the same SUT. Finally, we integrate 200 uniformly scattered observation points $D_{ij}$ along the SUTs, shown as dots in Figure 3(a) and 3(b), for monitoring the observable wake-up times during simulations.

In this distributed environment, the wake up time may be measured through any of the observation points $D_{ij}$ on the VPDN (marked $D_{ij}$ nodes in Figure 1(a) and in Figure 1(b)), an option not considered by the lumped model where the observation point is unique. In Section III we show that, when the distributed PDN model of Figure 3 is considered, the observable charging delay $M_{ij}$ depends on the observation point $D_{ij}$ and on the SUT $S_i$. The deviations introduced by these factors negatively affect test quality with both fault coverage and yield loss.

III. Analysis of PDNs Impact on Test Quality

Through SPICE simulations of the distributed model presented in Section II, we analyze the factors affecting the observable charging delay.

A. Dependence of charging delay on segment size $L$

Firstly, we consider the observable charging delay through a single observation point $D_C$, located at the corner of the SUT ($m \times m = 128 \times 16, 32 \times 64$ and $16 \times 128$). We simulate the test process for every SUT of each style and we gather the delays through a single observation point $D_C$. The results are presented in Figure 4 for both the ring (first row) and the grid style (second row), for the three considered segmentation setups. A bar in each graph presents the observed charging delay $M_C$ for every SUT $S_i$. We observe that the charging delay increases significantly as the number of power switches in the circuit decreases. This effect is more pronounced in the grid style, where the charging delay can be as high as $45\%$ compared to the ring style, where the charging delay is $25\%$.

![Fig. 4. The observable charging delay $M_{ij}$ at observation point $D_C$ (right corner of the power domain marked with an arrow) for every SUT $S_i$ on ring and grid power gating styles for three segmentation setups: $L \times m = 128 \times 16, 32 \times 64$ and $16 \times 128$.](image-url)
TABLE II

<table>
<thead>
<tr>
<th>circuit style</th>
<th>ethernet</th>
<th>ring</th>
<th>grid</th>
<th>ethernet</th>
<th>ring</th>
<th>grid</th>
<th>ethernet</th>
<th>ring</th>
<th>grid</th>
</tr>
</thead>
<tbody>
<tr>
<td>(L×m)</td>
<td>M σ%</td>
<td>M σ%</td>
<td>M σ%</td>
<td>(L×m)</td>
<td>M σ%</td>
<td>M σ%</td>
<td>(L×m)</td>
<td>M σ%</td>
<td>M σ%</td>
</tr>
<tr>
<td>128×16</td>
<td>5.9 21.2</td>
<td>2.5 23.5</td>
<td>1.4 51.0</td>
<td>128×16</td>
<td>5.9 21.2</td>
<td>2.5 23.5</td>
<td>1.4 51.0</td>
<td>128×16</td>
<td>5.9 21.2</td>
</tr>
<tr>
<td>64×64</td>
<td>20.1 14.1</td>
<td>9.9 6.2</td>
<td>2.4 42.2</td>
<td>1.1 67.0</td>
<td>2.4 42.2</td>
<td>1.1 67.0</td>
<td>2.4 42.2</td>
<td>1.1 67.0</td>
<td>2.4 42.2</td>
</tr>
<tr>
<td>32×32</td>
<td>58.7 10.6</td>
<td>14.7 7.7</td>
<td>2.7 41.7</td>
<td>1.3 36.1</td>
<td>2.7 41.7</td>
<td>1.3 36.1</td>
<td>2.7 41.7</td>
<td>1.3 36.1</td>
<td>2.7 41.7</td>
</tr>
</tbody>
</table>

Fig. 5. Observable charging delay from various observation points for activated SUT: (a) at the corner and (a) at the middle of the design.

and the observable delay \( M_{ij} \) increases. This result complies with the findings of previous works [8], [9], [15], since SUTs of smaller size \( L \) delay the wake-up time. Consequently, the observable charging delay \( M_{ij} \) depends on the segment size \( L \).

Yet, from Figure 4 we derive that \( M_{ij} \) depends on additional factors that are discussed next.

B. Dependence of charging delay on the activated SUT \( S_i \)

In Figure 4, we observe that the charging delay varies even for a single segmentation setup \( L \times m \). For the ethernet circuit and for segmentation setup of \( L \times M = 128 \times 16 \), the \( M_{ij} \) is in the ranges [3.57ns, 8.14ns] and [1.44ns, 3.59ns] for the ring and grid styles, respectively. From these graphs note that the charging delay \( M_{ij} \) depends on the distance of the SUT to the observation point, as expected. Particularly, it depends on the RC components between the activated SUT \( S_i \) and the observation point \( D_j \). The activation of a SUT \( S_i \) closer to the observation point \( D_j \) causes faster observable wake-up time. The same trend is observed for the rest of segmentation configurations. Consequently, the observable charging delay depends on the specific SUT \( S_i \).

Table II presents the observable delay variations of the benchmarks for both styles. For each circuit, the first column shows the examined segmentation setup \( L \times m \). The observable charging delay is presented with the average value \( M \) between the minimum and maximum values of the range and the relative standard deviation \( \sigma \). Note that for higher SUT sizes \( L \), the charging delay variation increases.

C. Dependence of charging delay on observation point \( D_j \)

Similarly, the observable charging delay of a SUT \( S_i \) depends on the observation point through which it is observed. In Figure 5 we present the observable charging delay when two different SUTs are activated for the grid style segmentation setup of \( L \times m = 64 \times 32 \). The ‘x’ and ‘y’ axis are the location coordinates of the corresponding observation point in the die and the ‘z’ axis is the observable charging delay, when a SUT is activated. The first SUT (Figure 5(a)) is located at the corner of the design and exhibits observable charging delays in the range [8.3ns, 11.2ns] and the other one at the center (Figure 5(b)) in the range [9.5ns, 10.5ns]. Note, as expected, that when observation points are closer to the activated SUT the observable charging delay is lower. Thus, we conclude that both the choice of the observation point \( D_j \) and the activated SUT \( S_i \) impact considerably the observable wake-up time.

D. Test quality degradation

Two scenarios affect the quality of power switches testing. Fault-free power switches may fail the test (false fails) and defective switches may pass the test (false passes). The first ones unnecessarily decrease yield, while the second ones threaten the reliability of the die [4]. Note in Figure 6 how two hypothetical scenarios (dashed lines) with observable charging delay that deviates from the one of the ideal lumped model (solid line) might affect test quality. The dashed line to the left reaches logic-0 too early and is susceptible to false passes, because it might mask faults. On the other hand, the dashed line to the right might reach logic-0 too late and is susceptible to false fails, because it might result to logic-1 even for a fault-free scenario. To evaluate the test quality degradation, we define as test quality \( (TQ) \) the quantity: \( (TQ = 100\% – \text{false passes} – \text{false fails}) \). Then, through fault injections, we gather the false passes, the false fails and TQ results, shown in Table III, for all the investigated
segmentation setups of both styles, when a lumped VPDN model is used with a single observation point and a single capture moment. The $TQ$ is higher for small segment sizes, because the sensitivity of those setups (Table II) to observation point selection is lower. Note that as the size of a SUT increases, the sensitivity of the delay to observation point selection (Table II) increases and the $TQ$ decreases (Table III), rendering previous DFT methods inapplicable for high speed testing of power switches. These results clearly motivate the importance of a DFT architecture that considers the distributed PDNs nature. Therefore, Section IV presents a novel PDN-aware DFT architecture and a method to restore test quality.

IV. PROPOSED PDN-AWARE DFT ARCHITECTURE

In this section we propose a PDN-aware DFT architecture that offers on-chip control of the parameters that affect the deviations of the observable wake-up time in order to restore test quality ($TQ$): the observation point $D_j$ that observes that delay and the SUT $S_i$. To avoid the need for multiple clock frequencies, the proposed DFT utilizes clock gating to generate variable capture moments. Practical heuristics are proposed to scale the DFT design method to large circuits and a compression scheme is proposed that reduces both the area cost of the DFT and the test application time (TAT).

A. DFT Design Flow

In Figure 7 we present the design flow of the proposed DFT architecture. It consists of four major steps described below.

1) Physical synthesis and segmentation setup: This step requires the power switches physical location, the distributed SPICE netlist and the segmentation setup $L \times m$. Then, the power switches are clustered in SUTs of size $L$, driven by the layout, as described in Section II and shown in Figure 3. Observation points $D_j$ of the $V_{Vdd}$ are injected following layout driven evenly scattered intervals on the VPDN.

2) Safe threshold computation: In Section II the basic scheme for testing power switches was described according to which the observation cell output is captured during capture moment by a flip-flop (Figure 6). Due to the factors that affect an observation, the capture moment will exhibit some deviation from the focal moment for at least some SUTs $S_i$. Recall from Section II that focal moment is the moment when the output of the NAND gate reaches logic-0. However, not all the deviations are harmful, if they do not affect test quality. Therefore, we introduce the safe threshold ($ST$), a time threshold that represents the maximum acceptable deviation between the focal moment for observation $M_{ij}$ and the capture moment. If ST is honored, neither false passes on single stuck open faults nor false fails are expected. The graph in Figure 8(a) shows how ST can be identified. The data of this graph correspond to the ethernet circuit and the segmentation setup of $L \times m = 128 \times 16$ of grid style (bottom-left graph of Figure 4). In Figure 8(a) the SUT $S_i$ is observed through the observation point $D_C$ which is located very close to the SUT $S_i$ in the layout. The single darked shaded line on the left of this graph shows the transient voltage at the observation point $D_C$ under fault-free scenario, while the other lines belong to the $L$ single-stuck open fault scenarios for every power switch in $S_i$. The results show that the observation of the fault-free scenario is different from the faulty ones. Particularly, the faulty scenarios, as expected, exhibit a higher delay. We refer to this additional delay as skew and we denote it as $H$. The earlier faulty observation exhibits the “lowest skew”. That lowest skew is selected as the safe threshold ST. To justify this selection, three possible capture moments are examined in Figure 8(b). The first moment occurs before the focal moment arriving after $M_{ij}$ delay and is susceptible to false fails, because the output of the observation cell has not reached yet logic-0. Next, we examine the second moment which occurs after the time moment $M_{ij} + ST$ and is susceptible to fault coverage (FC) loss due to false passes, since faulty scenarios have already reached logic-0 in that range. Finally, we examine the third moment which occurs in a time window between the fault-free observation $M_{ij}$ and the moment $M_{ij} + ST$. This capture moment selection does not imply any TQ loss, because the fault-free scenario has already reached logic-0 and the faulty scenarios arrive after the capture moment. Consequently, if a ‘logic-0’ value is captured at the
flip-flop, then the SUT is fault-free, while if a ‘logic-1’ value is captured, then it suffers from at least one fault.

It is worth noting that ST, which is the lowest delay skew LH of single stuck-open faults from the fault-free scenario, varies for every SUT \( S_i \) and observation point selection \( D_j \) (\( ST_{ij} \)). For large designs, fault simulating all the SUTs, even for single stuck-open faults, might lead to a formidable number of fault simulations. Therefore, to reduce the number of required fault simulations for the safe threshold ST computation, the following heuristics are proposed:

**Heuristic** \( h_1 \): Fault simulate the farthest to the observation point power switch as faulty. From Figure 8(a) derives that the least skewed curve to the left belongs to the scenario where the faulty power switch is far from the observation point.

**Heuristic** \( h_2 \): Focus only at the observation point which is closer to a SUT \( S_i \). In Figure 9 the LH for two SUTs of the ethernet circuit at the grid style segmentation setup \( L \times m = 32 \times 64 \) are presented for all possible single stuck open faulty scenarios. One SUT is located at the corner of the design (Figure 9(a)) and the other one is located at the center of the design (Figure 9(b)). The ‘x’ and ‘y’ axes denote the location of the observation point in the layout and the ‘z’ axis the lowest skew. In both cases the lowest skew is observed through the observation points closer to the activated SUT.

**Heuristic** \( h_3 \): Fault simulate the SUTs at the areas with the lowest IR-drop. Even with the two heuristics \( h_1 \) and \( h_2 \) every SUT should be fault simulated. Consider the graphs in Figure 9. The absolute value of the observed skew at the corner (Figure 9(b)) SUT is lower compared to the SUT at the corner (Figure 9(a)) for the grid style architecture. The reason is that the SUT at the corner is located at an area with lower IR-drop and it is more tolerant on faulty power switches compared to another SUT at an area with higher IR-drop. For the same reason, SUTs at areas with lower IR-drop (Figure 5(b)) exhibit less deviations on the observable charging delay compared to those at areas with higher IR-drop (Figure 5(a)).

In our experiments we used this heuristic to obtain a global minimum ST for all SUTs and observation points. Note that this selection is pessimistic and a safe threshold per SUT and minimum ST for all SUTs and observation points. Note that compared to those at areas with higher IR-drop (Figure 5(a)).

Besides, we observe in Figure 9 that the lowest skew is exhibited very close to the SUT. A minimum distance constraint, for example excluding an observation point to be assigned to the SUT it belongs, between the observation point and the selected SUT leads to \( 1.5 \times \) to \( 2 \times \) larger safe threshold values. This feature benefits the robustness of the proposed method and reduces the observable charging delay deviations due to random noise on the RC parasitics of the PDNs.

3) Observation points selection: The Algorithm 1 in Figure 7 selects a minimum observation points set \( OP \) so as for every SUT \( S_i \) to be at least one \( OP_j \), the safe threshold of which is honored. This condition is referred to as compatibility and discussed below. The observable charging delay \( M_{ij} \) is computed for every pair of SUT \( S_i \) and candidate observation point \( D_j \) through simulations of the distributed model. Also, the TQ is affected by the capture moment. Therefore, the proposed architecture offers control over the selection of the capture moment by using the rising edges of the system clock. Hence, for every pair \((S_i, D_j)\) with observable charging delay \( M_{ij} \), the system clock rising edges are evaluated for honoring the ST. If at least one rising edge honors ST, then the pair \((S_i, D_j)\) is marked as compatible. The clock edges that honor the ST are denoted as skip cycles \( c_{ij} \). Note that minimum set of observation points OP must contain at least one compatible \( OP_j \) for every SUT \( S_i \) to avoid TQ loss. This is guaranteed by iteratively applying the following two criteria that minimize the number of observation points \( |OP| \) (and consequently area cost) and TAT:

\( C_1 \): Select the set of observation points \( D_j \) with the most compatible SUTs in \( S \).

\( C_2 \): Among those \( D_j \) selected by criterion \( C_1 \), select the one that requires the minimum average number of skip cycles \( \bar{c}_{ij} \) for all its compatible SUTs.

Any new observation point selection follows these criteria and its compatible SUTs are dropped from set \( S \). The algorithm terminates when the set \( S \) is empty. If the designer has set the more observation points parameter, MOP, to a value greater than the minimum number of observation points \( |OP| \), the algorithm selects MOP number of observation points. This property offers a trade-off between area cost and TAT that will be shown in Section VI.

Example: Figure 10 presents a case of three SUTs \( S_1, S_2 \) and \( S_3 \) and three candidate observation points \( D_1, D_2 \) and \( D_3 \). The clock edges honoring the safe threshold \( ST_{ij} \) between a SUT and an observation point are shaded. Applying criterion \( C_1 \) on this example leads to the selection of both observation points \( D_1 \) and \( D_2 \), since these are compatible with all the SUTs, while observation point \( D_3 \) is compatible only with \( S_2 \). Next, criterion \( C_2 \) is applied according to which the observation point with the minimum average number of skip cycles per SUT is selected. This criterion targets the minimization of
Fig. 11. Proposed DFT architecture.

TAT. For $D_1$ that value is $\overline{c_{ij}} = (3 + 23 + 5)/3 = 10.3$ and for $D_2$ is $\overline{c_{ij}} = (12 + 7 + 24)/3 = 14.3$. Consequently, $D_1$ is selected and added at set OP. Next, all compatible to $D_1$ SUTs are dropped from set $S$. This action results in an empty set $S$, since in this example $D_1$ is compatible with all the SUTs. Consequently, the algorithm ends. If the parameter $MOP$ was set to $MOP = 2$, an additional observation point would be selected. In this example, $D_3$ would also be selected, since it qualifies for the first criterion on the empty set $S$ and it requires the less average number of clock cycles per compatible SUTs, since it is $\overline{c_{ij}} = 2/1 = 2$. Now, SUT $S_2$ can be tested spending fewer skip cycles through $D_3$. Thus, $MOP$ offers a trade-off between area cost and TAT.

4) Test and Control Data Generation: After the selection of the set $OP$, the test generation process assigns every SUT $S_i$ to a compatible observation point $OP_j$. Since a SUT may be compatible to more than one observation points from the set $OP$, it selects the one with the minimum skip cycles $c_{ij}$ in order to reduce TAT. In the example of Figure 10 SUTs $S_1$ and $S_3$ are assigned to observation point $OP_1 = D_1$ and SUT $S_2$ to observation point $OP_2 = D_3$. Note that each triplet of SUT $S_i$, observation point $OP_j$ and skip cycles $c_{ij}$ is required to be stored on-chip in order to control the DFT logic. In Section VI-E we show that this area cost can become overwhelming, especially for setups with large SUTs number $m$. To limit this overhead, we apply the following process that compresses these data using the Run-Length (RL) compression [20] and requires minimum decompression logic. The basic idea is that many SUTs $S_i$ require the activation of the same $OP_j$ since $OP$ set is selected to be minimum. We compress this correspondence (pairs $S_i$, $OP_j$) using the RL code and we store it on-chip in a register file OP-REG. Specifically, each entry OP-REG[$j$] stores the number of successive SUTs that require the activation of observation point $OP_j$. Next note from Figure 10 that the skip cycles $c_{ij}$ that honor the safe threshold $ST$ is not a unique value but a range of compatible successive skip cycles. In this example the skip cycles $c_{ij} = 5$ suffice for testing all the SUTs. Therefore, each pair $(S_i, OP_j)$ is assigned to the compatible skip cycle value $c_{ij}$ with the most occurrences at all SUTs. In Section VI-E, we show that this action favors the RL compression efficiency. These control data are stored at the register file C-REG. Each entry C-REG[k] stores the skip cycles $c_{ij}$ and the number of successive pairs that require the same skip cycles. In Section VI-E, we demonstrate the compression efficiency of this technique.

An alternative to reduce the compression scheme complexity is to omit the compression of the observation points assignment (OP-REG) and repeat every test from every observation point with the cost of TAT. However, part of the TAT spend could be restored after the post silicon calibration that is presented in Section V-B by reordering the observation points according to the number of their compatible SUTs.

Note that the above method requires the system clock frequency to evaluate and select the observation points. However, it is not affected by after speed-binning clock-frequency selection, because manufacturing testing is conducted before speed binning and the testing is applied using a clock frequency provided by the PLL generator of the die [21].

B. Architecture

The proposed DFT architecture is shown in Figure 11. It consists of four major blocks:

**Test Observation Logic (TOL):** This block generates the capture edge and activates an observation point $OP_j$ out of a set of minimum observation points $OP$. Each SUT requires a different capture edge and $OP_j$ for 100% TQ. This unit latches system clock as long as the capture signal is zero and the multiplexer OP-MUX selects the appropriate observation point $OP_j$ indicated by the opselect value. A flip-flop stores the test result, when capture is asserted.

**TOL Controller (TOLC):** This block is responsible for generating the control signals opselect and capture for the TOL unit. First, the Observation Point Controller (OPC) generates on-chip the opselect signal to control the activation of a single observation point for a particular SUT $S_i$. The compressed data for the pairs $S_i$ and $OP_j$ are stored in a register file (OP-REG). Each register stores the opselect value and the number of successive SUTs that use it. Secondly, the Capture Edge Controller (CEC) generates on-chip the capture signal that controls the clock gating of the system clock. A counter counts down $c_{ij}$ clock rising edges, denoted by skip cycles, before asserting capture. The skip cycles $c_{ij}$ correspondence for each pair $S_i$ and $OP_j$ is also stored compressed in a C-REG register file. The C-REG contents are serially loaded with the skip cycles recomputed by the post-silicon calibration process, which is presented in Section V.

**Observation Cells (OCs):** The NAND observation cells, shown as an oval shape in Figure 11, are attached on a minimum set of observation points $OP$ selected by the algorithm
of Section IV-A3 that achieve 100% TQ. Voltage monitoring alternatives like those reported in [22] can be deployed. **Finite State Machine (FSM):** An FSM that assures the stuck-open faults test process coordination, according to the application scheme described in Section II.

V. HANDLING UNCERTAINTY AND CALIBRATION

Uncertainty on the observable charging delay of both fault-free and faulty scenarios impacts test quality. Some uncertainty sources, like device process variations, can be considered by the design, while others, like the inadequate analog characterization of digital technologies [23], [24], are not experienced before the circuit has been manufactured. In this section we propose practical solutions to handle modeled uncertainty during DFT design (Section V-A) and unmodeled uncertainty during manufacturing testing with calibration from post-silicon measurements (Section V-B). We also demonstrate a test quality enhancement method (Section V-C) that considers systematic variations [25]–[27].

A. Handling uncertainty during DFT

We adapt the safe threshold computation and the observation points selection to handle uncertainty. We assume 20% threshold voltage ($V_{th}$) and oxide thickness ($t_{ox}$) variations for all CMOS devices of the circuit. In Figure 12, we present the result of four Monte Carlo simulations with 500 permutations each, considering fault-free and single stuck-open faulty scenarios for two segment sizes $L = 8$ (Figure 12(a)) and $L = 16$ (Figure 12(b)) of the s38584 benchmark circuit. The ‘x’-axis is charging delay ranges and the ‘y’-axis reports the number of occurrences (bar) and their probability density functions (PDFs) (lines). The dark-shaded $PDF(0) = [M_{ij}^0, \sigma_{ij}^0]$ belongs to the fault-free scenarios and the light-shaded $PDF(1) = [M_{ij}^1, \sigma_{ij}^1]$ to the faulty ones. Note that RC parasitics are considered by the proposed PDN-aware DFT (Section IV) and that process variations could affect both PDN parasitics and the CMOS devices. However, PDN parasitics variations have negligible impact on the charging delay compared to CMOS devices variations, especially when the minimum distance between the SUT and the assigned observation point is constrained.

In Figure 12(a), the $PDF(0)$ of the fault-free scenarios charging delay and the $PDF(1)$ of the faulty scenarios do not overlap. This case is identified by the variations-aware safe threshold $V_{ST} = (M_{ij}^1 - 3 \times \sigma_{ij}^1) - (M_{ij}^0 + 3 \times \sigma_{ij}^0)$. Since it is $V_{ST} > 0$, the overlap of the two PDFs is negligible. Note that a SUT, although it may be free from stuck-opens, it may suffer from parametric faults with charging delay similar to that of a stuck open scenario. However, we target stuck-opens and we consider the variation of the process parameters as fault-free by selecting the fault-free observable charging delay to be $M_{ij} = M_{ij}^0 + 3 \times \sigma_{ij}^0$. Any parametric faults with higher charging delay than $M_{ij}$ are testable by this selection and a lower fault-free charging delay $PM_{ij}$ (shown in Figure 12) could be selected for parametric faults detection. After setting the fault-free charging delay $M_{ij}$, the observation points selection algorithm (Section IV-A3) is applied with threshold $ST = V_{ST}$. Heuristics $h_1$, $h_2$ and $h_3$ could be used to identify the combination of observation point $D_j$ and SUT $S_i$ with the maximum $\sigma_{ij}$ values for $V_{ST}$ computation.

In Figure 12(b), the charging delays PDFs of the fault-free and the faulty scenarios overlap. This is also indicated by the negative $V_{ST}$ value. The overlapping region belongs to fault-free scenarios, which exhibit high charging delay, due to process variations, that masks single stuck-open faults with low charging delay. A possible ad-hoc solution to avoid the overlapping region is the selection of a smaller segment size. However, a large SUT could be considered tolerant to stuck-open faults, if a faulty switch does not induce a higher charging delay than that of a fault-free SUT. Therefore, the fault-free charging delay is set at $M_{ij} = M_{ij}^0 + 3 \times \sigma_{ij}^0$. This way yield loss is avoided in exchange for fault coverage loss. However, a lower fault-free charging delay could be assigned for critical SUTs that are spatially closer to the longest paths. Also, since the safe threshold is negative, the observation points selection algorithm (Section IV-A3) is modified to select the clock edge that is closer to the $M_{ij}$ value. Next, we present the proposed calibration method to handle uncertainty due to unmodeled variations (Section V-B) and we present an approach to enhance test quality by handling systematic variations when $V_{ST} < 0$ (Section V-C).

B. Calibration for bridging model-to-silicon gaps

The proposed DFT method relies on SPICE simulations in order to estimate charging delays. However, due to model-to-silicon discrepancies [23], [24], SPICE simulations might be inaccurate compared to actual hardware measurements. Unconsidered PDN parasitics is a possible model-to-silicon discrepancy that could affect the PDN voltage level and the test quality. Therefore, we propose the post-silicon calibration process shown in Figure 13. During this process, the calibration signal (Figure 11) exposes observation points output to the oscilloscope of the Automated Test Equipment (ATE) that collects the charging delay measurements. After the collection of an adequate measurements number, the post-silicon $PDF(0) = [M_{ij}^0, \sigma_{ij}^0]$ of the fault-free scenarios is
computed through regression analysis. Next, the calibration configuration (CF), which is the set of skip cycles variables $c_{ij}$ (Section IV-A3) for every group of SUTs is recomputed, by deploying the observation points selection algorithm (Section IV-A3) with the available on-chip observation points. The DFT is configured with CF, when it is set at test mode.

C. Handling of systematic variation

A practical approach is presented that increases the test quality of the proposed method under systematic variations (SVs) for large segment sizes with $V_{ST} < 0$. Spatial correlated variations on the CMOS devices and parasitics of the PDNs are common between both inter-die and intra-die variability [25]–[27]. Systematic variations are explored by analysis of variance (ANOVA) [26], [27], a random to systematic variation decomposition technique by ranking candidate effects as possible sources of variation. For example, the wafer shown in Figure 13 is shaded according to the edge effect and the center effect [27], two common sources of systematic spatial variability. The flow of this process is shown with dashed lines in Figure 13 together with the calibration flow. The calibration process jointly considers the collected measurements and the systematic variations to evaluate the calibration configuration. The systematic variations are used for clustering the SUTs and a calibration configuration is computed per cluster according to the discrepancies between each cluster’s charging delay mean. An example of this approach on spatial variations is demonstrated in Section VI-B.

Finally, note that the proposed method does not stress the chip at the corner cases of its thermal envelope [28] and temperature variability during testing is expected very low. However, if systematic temperature variations are observed during manufacturing testing by temperature sensors, they can be handled similarly to the case of systematic process variations by conducting clustering of charging delays based on discrepancies between the means of temperature variations.

VI. SIMULATION RESULTS

In this section, we evaluate, through SPICE simulation, the performance of the proposed DFT architecture (Figure 11). We analyzed a large number of benchmarks from the IWLS’05 benchmark suite [19] and selected three representatives to present: the ethernet (the largest of the IWLS’05), the s38417 and the s38584 circuits (the largest of the ISCAS’89 benchmarks included in the IWLS’05 suite). We consider the distributed PDNs model (Figure 3 and Table I) for both ring and grid power gating styles and various power switches segmentation setups. Also, for various parameters of the flow (Figure 7), we show the available trade-offs on test-application-time (TAT) and area cost. The operating frequency of all benchmarks was set at $f = 1$GHz.

A. Test quality and TAT evaluation

First, we present the results of the proposed method for various segmentation setups $L \times m$ for both ring and grid power gating styles for the three examined benchmarks. The parameter MOP (More Observation Points) is set to zero in order to trigger the selection of a minimum set of observation points. In Table IV, we present the area cost, in number of observation points $\{OP\}$, the size in flip-flops of the register file OP-REG that stores the observation point selection control data and the size of the register file C-REG that stores the skip cycles control data. The TAT and the TQ (without process variations) of the proposed method is also presented. The columns ‘R’ and ‘G’ belong to the ring and grid style, respectively. For the ethernet benchmark, the selected observation points number is in the range [1, 10] and [1, 9] for the grid and ring styles, respectively. The register files size (OP-REG + C-REG) is also very low, in the range [30, 82] and [33, 167] flip-flops for the grid and ring style, respectively. For all benchmarks and segmentation setups, the proposed method restores $TQ$ to 100% when process variations are omitted.

We highlight that the power switches are tested per SUT of $L$ number of power switches each and the TAT reported in Table IV was computed using $L$ (segment size) discharge transistors of equal size with the power switches. Hence the TAT of $m$ number of SUTs and $\overline{M}$ average charging time per SUT is bounded by $TAT = m \times 2 \times \overline{M}$, counting also for the discharging time through the discharge transistors [8]. Note that for the ethernet, 8 discharge transistors, of equal size with the power switches, are enough to reduce the discharge time by 95%. Thus, we conclude that a small amount of discharge transistors can efficiently reduce the TAT for discharging the
circuit. Note that discharge transistors are already considered by practical solutions for testing power supplies [22].

B. Test quality evaluation under process variations

Next, we consider 5%, 10% and 20% process variations at the $V_{th}$ and $t_{ox}$ of the CMOS devices and we conduct Monte Carlo simulation of 500 permutations each. The variation for the $V_{th}$ is shown in Figure 14(a). We also considered a case of systematic variations, shown in Figure 14(b). In that case we assume that the random variations of 20% has been decomposed to three components of systematic variations with 10% random variation each which is a possible outcome of analyzing, using ANOVA variation decomposition methods [26], [27], the spatial edge and center effects of wafer-to-die variability [27]. The dies in the wafer in Figure 13 have been shaded according to these effects. Figure 14(c) depicts the test quality loss ($TQ$ loss) = (false passes) + (false fails) of the examined cases after applying the calibration method (Section V) under process variations. The yield loss noticed was less than 0.6%. However, it suffers from false passes in the range of [5%, 40.5%] when $RV = 20\%$ based on the segment size $L$. Next we evaluate the ability of the proposed method to handle systematic variations. The results for 20% systematic process variation (case Figure 14(b)) are shown in Figure 14(c) with label “SV=20% of 3×RV=10%”. The TQ loss (false passes) for that case drops (from [5%, 40.5%]) to the range of [1.4%, 19.7%], which is attributed to the efficiency of the calibration method to handle systematic variations. The TQ improvement compared to the PDN-unaware testing is in the range of 81.1% to 90% based on the segment size $L$, even when the PDN-unaware is evaluated with 0% process variations.

Note that the presented results consider process variations also at the NAND observation cells. Omitting the process variations only at the observation cells, decreases the deviations of the charging delays by 50%. Therefore, voltage monitoring alternatives, like those reported in [22], could be deployed as observation cells for more robust measurements.

C. Storage requirements and compression evaluation

Next, we evaluate the selection of RL compression for compressing the tests (triplets $S_i$, $D_j$, $c_{ij}$). Figure 15 depicts the storage requirements in number of flip-flops for storing the tests of the ethernet benchmark in uncompressed (triplets $S_i$, $D_j$, $c_{ij}$) and compressed (register files size C-REG+OP-REG) form for various setups $L \times m$. Note that although only the SUTs number $m$ is shown on x-axis, the size of a segment $L$ is given by $L = 2048/m$. As the SUTs number $m$ increases from $m = 8$ to $m = 128$, the uncompressed storage requirements increase from 80 to 1792 flip-flops. However, when RL compression is used, the compressed storage requirements increase slowly from 30 to 82 flip-flops in the range $m = 8$ to $m = 32$. For higher SUTs number $m$ ($m = 64$ and $m = 128$), they decrease to 30 and 37 flip-flops, respectively. This is attributed to the higher observable charging delay of smaller SUT sizes $L (L = 2048/m)$ that increases the range of compatible successive skip cycles, introduced in Section IV-A4. Therefore for higher SUTs number $m$, it is more frequent to select the same skip cycles $c_{ij}$ between SUTs, which results to higher compression ratio. The achieved compression ratio of the RL compression $R = (uncompressed\ storage\ req.)/(compressed\ storage\ req.)$ is shown in Figure 15. It is always $R > 1$ and increases from $R = 2.67 \times$ to $R = 48.43 \times$ less storage requirements, while the SUTs number increases, clearly showing the efficiency of the RL compression for this type of data.
TABLE V
AREA COST RESULTS

<table>
<thead>
<tr>
<th>circuit</th>
<th>size (ge)</th>
<th>ring area (ge)</th>
<th>area (%)</th>
<th>grid area (ge)</th>
<th>area (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ethernet</td>
<td>157.5K</td>
<td>1015</td>
<td>0.64</td>
<td>643</td>
<td>0.41</td>
</tr>
<tr>
<td>s38417</td>
<td>30.5K</td>
<td>490</td>
<td>1.61</td>
<td>454</td>
<td>1.49</td>
</tr>
<tr>
<td>s38584</td>
<td>26.9K</td>
<td>611</td>
<td>2.27</td>
<td>435</td>
<td>1.62</td>
</tr>
</tbody>
</table>

D. Area cost evaluation

The proposed DFT architecture has the following area cost: area = (storage) + (control logic) + |OP| + |OP| × 1 MUX. Table V presents the area cost for both ring and grid styles for the setups of Table IV with the highest hardware cost. Area cost is given in gate equivalents in column “area (ge)”. Column “area (%)” presents the relative area cost compared to the area cost of the circuit which is reported in column “size (ge)”. For example, for the ethernet benchmark of the grid style, the maximum storage requirements after compression (register files size OP-REG+C-REG) is obtained for the setup \(L \times m = 64 \times 32\) and it is 82 flip-flops. Similarly, the highest control logic hardware cost for the ethernet is 280 gate equivalents. Specifically, this cost includes 4 counters for RL-decompression (CEC+OPC blocks), the SUTs counter, the FSM that coordinates the test application and a shift register of size \(m\) for asserting the sleep signal of the SUT. One counter is for addressing the C-REG register file and it is of \(log_2(C-REG)\) bits size. The other one is for counting down the successive SUTs that require the same skip cycles \(c_{ij}\) and its size is bounded by \(log_2(m)\). Similarly, another counter is required for addressing the OP-REG register file and it is of \(log_2(OP-REG)\) bits size. The last counter is for counting down the successive SUTs that require the same observation point \(OP_j\) and its size is also bounded by \(log_2(m)\). Finally, the observation logic requires \(|OP|\) observation cells and it is in the range \([1, 10]\) NAND gates for the ethernet circuit. Therefore, for the case of the ethernet, the proposed method leads to 55%-68% area overhead compared to the state-of-the-art [15], which, however, is 0.41% the design size. The worst area overhead is 2.27% for the ring style of the s38584 circuit which is the smaller benchmark. Note that the relative area cost of the proposed method drops as the size of the circuit increases, clearly showing its scalability to large designs. We conclude that the proposed method achieves to restore TQ with very low hardware overhead.

E. Trade-off between TAT and area cost

Next, Figure 16 presents a trade-off between hardware overhead and TAT for more observation points \(MOP = 3, 4, \ldots 16\) for the ethernet grid style setup of \(L \times m = 32 \times 64\). These values trigger the selection of more than the minimum \(|OP| = 2\) observation points. Both the TAT improvement and the hardware overhead are presented compared to the minimum observation points selection \(|OP| = 2\). While the \(|OP|\) increases from 2 to 16, the storage requirements fluctuate in the range \([30, 44]\) flip-flops, which is very low. Meanwhile, for \(MOP = 16\), TAT decreases by 17% compared to the case of \(|OP| = 2\), clearly indicating that more observation points can be spared for less TAT, a trade-off observed in all simulations. Finally, note in Figure 16, how the value of \(MOP = 10\) is marked, as a pareto point, that minimizes the TAT with the minimum additional storage requirements.

VII. CONCLUSIONS

We showed that delay-based testing of power switches must consider a distributed model for the PDNs in order to avoid fault coverage loss and yield loss. To tackle this problem, we proposed a new PDN-aware DFT architecture (Figure 11), which is suitable for both ring and grid power gating styles. The DFT design flow (Figure 7) consists of practical heuristics (Section IV-A2) for scaling fault simulation requirements and an algorithm (Section IV-A3) that optimizes multiple objectives: test quality, TAT and area cost. The proposed method handles uncertainty (Section V-A) and can be calibrated (Section V-B) from post-silicon measurements. An approach to improve test quality when systematic variations are considered was also demonstrated (Section V-C). The simulation results show that the test quality which was lost due to PDNs is fully recovered (Table IV) and that 83.3% to 98.6% of the restored test quality is robust under process variations (Figure 14). A trade-off between area cost and TAT (Figure 16) has also been demonstrated. Finally, the proposed DFT requires minimum area cost (Table V) of less than 0.42% percent for a design with 157.5K gate.

ACKNOWLEDGMENTS

This work was supported by the Engineering and Physical Sciences Research Council, U.K., under Grant EP/K000810/1. The work of Dr. S. Khursheed was supported by the Department of Electrical Engineering and Electronics, University of Liverpool, U.K. The authors would like to thank Prof. Sudhakar M. Reddy (Iowa University) for his constructive feedback.

REFERENCES


