Delay Test for Diagnosis of Power Switches

Saqib Khursheed, Kan Shi, Bashir M. Al-Hashimi, *Fellow, IEEE*, Peter R. Wilson, *Senior Member, IEEE*, and Krishnendu Chakrabarty, *Fellow, IEEE*

***Abstract*— Power switches are used as a part of the power-gating technique to reduce the leakage power of a design. To the best of our knowledge, this is the first report in open literature to show a systematic diagnosis method for accurately diagnosing power switches. The proposed diagnosis method utilizes the recently proposed design-for-test solution for efficient testing of power switches in the presence of process, voltage, and temperature variation. It divides power switches into segments such that any faulty power switch is detectable, thereby achieving high diagnosis accuracy. The proposed diagnosis method is validated through SPICE simulation using a number of ISCAS benchmarks synthesized with a 90-nm gate library. Simulation results show that, when considering the influence of process variation, the worst case loss of accuracy is less than 4.5%; it is less than 12% when considering VT variations.**

***Index Terms*— Design for test (DFT), diagnosis, leakage power management, power gating, sleep transistor.**

I. Introduction

POWER gating is a low-power design technique to reduce leakage power. It has gained popularity in sub-100-nm CMOS designs, where leakage power is a major contributor to the overall power consumption [1]. It utilizes power switches (also called sleep transistors) to power-down the logic blocks during the idle mode to reduce leakage power consumption [2]. Power switches are implemented as header switches or footer switches. This paper analyzes headers in detail, but the results are equally applicable to footer power switches as well. Power switches are usually implemented in either "fine-grain" or "coarse-grain" design styles. A fine-grain style incorporates a power switch within each standard logic cell with a control signal to switch on/off the power supply of the cell. In the coarse-grain design style, a number of power switches are combined to feed a block of logic. When comparing the two design styles, the fine-grain design simplifies the incorporation of power gating through existing EDA tools, but it has a higher area overhead and is more vulnerable to voltage drop fluctuations due to process, voltage, and temperature (PVT) variations [2]. Therefore, the coarse-grain design style is a more popular design choice in practice and is the focus of this paper. Power switches are implemented in two power modes, which provides a tradeoff between leakage power saving and wake-up time. These include complete power-off mode (higher leakage power saving) and intermediate power-off mode (lower wake-up time). Design-for-test (DFT) solutions for power switches with intermediate power-off mode have been recently proposed [3], [4]. Therefore, this paper focuses on power switches with a complete power-off mode.

Diagnosis is a systematic method to uniquely identify the defect causing malfunction in the circuit. It is critical to silicon debugging, for yield analysis, and for improving subsequent manufacturing cycle [6]–[8]. Recent research has reported a number of DFT solutions to test power switches when considering the two possible type of faults: stuck-open and stuck-short [5], [9]–[11]. Stuck-open fault models a physical scenario where the drain or source of a transistor is disconnected leading to a faulty transistor behavior. Testing such faults require two test vectors. The first test vector drives the output of a transistor to logic high or low, while the second test vector complements the output logic value using each transistor in the pull-up or pull-down network [7]. Stuck-short faults produce a conducting path between V_{dd} and ground and may be detected by a test technique called I_{DDQ} testing which monitors the current flow during a steady-state condition [7]. The first DFT solution was reported in [9], and was used to test power switches in both fine-grain and coarse-grain designs. However, it was highlighted in [10] that this DFT solution suffers from long discharge time when the power switches are turned off. This leads to long test time due to the necessity of applying a slower test clock and may lead to false test (false-fail or false-pass). This problem was addressed in [5] through an effective DFT solution, which added a low-leakage (high V_{th}) discharge transistor segment to the DFT. This is because the discharge transistor is switched off during normal operation of the design, therefore high performance and leaky (low V_{th} or standard V_{th}) transistors are unnecessary. This is why a high V_{th} (low performance and less leaky) nMOS transistor is used as a discharge transistor. It is designed such that I_{on} is maximum and I_{off} is minimum to ensure low leakage through virtual rail DFT during normal operation of the design. See Section III-A and Fig. 4 for more details about discharge transistor design including SPICE simulation results. This DFT solution is shown in Fig. 1, where Fig. 1(a) shows the DFT for a fine-grain design and Fig. (1b) shows that for a coarse-grain design. The DFT proposed in [5] achieves fast test time through balanced charge and discharge times and eliminates

Manuscript received March 28, 2012; revised October 10, 2012; accepted December 30, 2012. Date of publication January 29, 2013; date of current version January 17, 2014. This work was supported by the Engineering and Physical Sciences Research Council, U.K., under Grant EP/H011420/1. The work of K. Chakrabarty was supported in part by the National Science Foundation under Grant CCF-0903392.

S. Khursheed, B. M. Al-Hashimi, and P. R. Wilson are with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (e-mail: ssk@ecs.soton.ac.uk; bmah@ecs.soton.ac.uk; prw@ecs.soton.ac.uk).

K. Shi is with the Electrical and Electronics Engineering Department, Imperial College London, London SW7 2AZ, U.K. (e-mail: k.shi11@imperial.ac.uk).

K. Chakrabarty is with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27705 USA (e-mail: krish@ee.duke.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2013.2239319

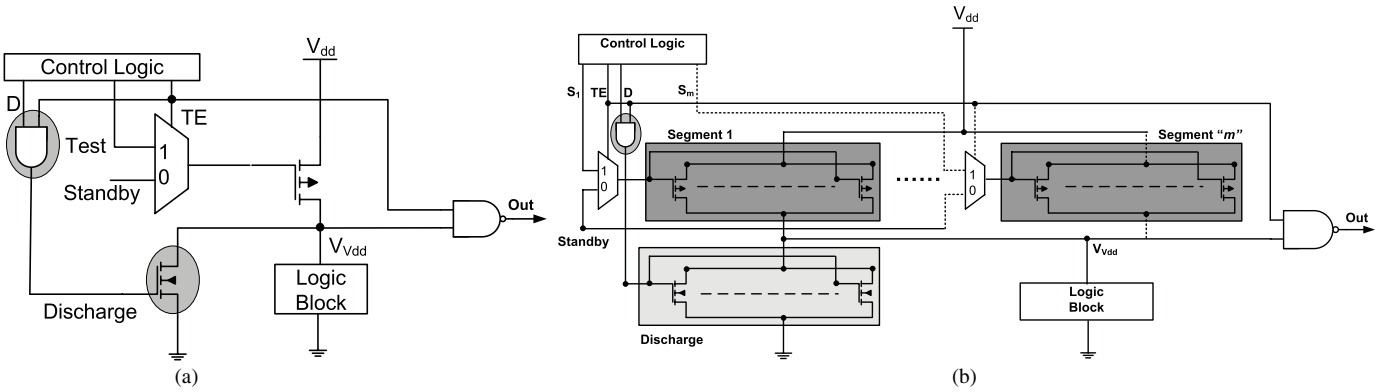


Fig. 1. DFT for testing power-switches [5]. (a) Fine-grain design. (b) Coarse-grain design.

the possibility of a false test. In [11], test and diagnosis of power switches is proposed through delay test by simulating circuit delay at the outputs of functional unit of the design. It is motivated by the fact that the circuit delay increases with a larger number of faulty (stuck-open) power switches. The drawback of such a technique is that it is not possible to locate the exact cause of additional delay leading to IC timing violation, and therefore it is not possible to distinguish between the delay fault caused by power switches, logic gates, or interconnects on the faulty paths, thereby affecting negatively the overall diagnosis accuracy. This clearly motivates the need for an accurate and efficient method for diagnosing power switches.

This paper proposes an efficient delay-test-based diagnosis method for power switches. Using a coarse-grain design style [Fig. 1(b)], it demonstrates how to divide power switches into segments to achieve high diagnosis accuracy with minimum possible hardware overhead. In this paper, the number of power switches per segment is referred to as the segment size. For each design, the segment size is determined through detailed trade-off analysis between the test frequency and the segment size using HSPICE simulations. The proposed diagnosis method capitalizes on optimal segment sizes (per design) to determine the number of faulty power switches in a design. Experiments are conducted on a 90-nm gate library, and the proposed method is analyzed under nominal operating conditions and under the influence of PVT variations. It is shown that, in the worst case, the loss of diagnosis accuracy is less than 12%. To the best of our knowledge, this is the first report on the diagnosis of power switches that shows a systematic diagnosis method for accurately diagnosing power switches in the presence of PVT variation.

The rest of this paper is organized as follows. Section II describes how segment sizing can be exploited for higher diagnosis accuracy. The proposed diagnosis algorithm is presented in Section III. Simulation results are reported in Section IV, and finally Section V concludes this paper.

II. Analysis of Segment Size and Test Frequency

In this section, we analyze two important parameters that affect diagnosis accuracy of power switches: segment size and test frequency. Fig. 1(b) shows a typical coarse-grain power gating design along with its DFT [5]. The DFT consists of a control logic for controlling the test sequence, a multiplexer to enable the test mode, an AND gate to control the discharge transistor segment, and an output NAND gate where fault effect is observed. For this analysis, the logic block consists of ISCAS benchmark designs that are synthesized using a 90-nm ST-Microelectronics gate library. The netlist is converted to SPICE format using Synopsys STAR-RCXT to allow detailed HSPICE analysis. The operating voltage used in this experiment is 1 V and operating temperature is 25 °C. Power switches used in this experiment are high V_{th} pMOS transistors, where each switch has a width of 1.1 μm and length of 150 nm. See [2] for more details on power switch design. Table I shows three benchmarks along with the number of power switches needed for each design. The number of power switches used for each design varies to achieve $\leq$ 5% IR drop target [2], [12]. IR drop is determined in the active mode by simulating the voltage at VS_{Vdd} while feeding the transition pulses (high-to-low and low-to-high) to the primary inputs of each design. The number of discharge transistors is chosen to achieve a balanced charge/discharge time at the V_{dd} when only transistors of one segment are turned on. The charge time is defined as the time it takes for the voltage level to reach 90% of V_{dd}, and the discharge time is the time it

TABLE I
Tradeoff Between Diagnosis Accuracy and Segment Size

Design	Total number of PS	Segment Size	Dectectable Power Switches	Diagnosis Accuracy
C432	30	5	5	100%
		10	9	90.00%
		15	12	80.00%
		30	22	73.30%
C1908	120	5	5	100%
		10	9	90.00%
		15	13	86.70%
		30	24	80.00%
C2670	180	5	5	100%
		10	10	100%
		15	14	93.30%
		30	25	83.30%

TABLE II
TRADEOFF BETWEEN DIAGNOSIS ACCURACY AND TEST FREQUENCY

Design	Segment Size	Falling Delay df (ns)	Test Freq (GHz)	Diagnosis Accuracy
C432	5	0.41	1.74	100%
	10	0.207	3.45	90.00%
			4	100%
	15	0.143	5	80.00%
			5.8	93.30%
			6.4	100%

takes for the voltage level to reach 10% of V_{dd}; see [5] for more details on designing discharge transistors.

In coarse-grain design style, power switches are divided into segments and the number of power switches per segment is a tradeoff between area overhead, test time, and precision in identifying faulty transistors [5], [9], [10]. This is shown in Table I, where for each design, the segment size is varied from 5 to 30, and diagnosis accuracy evaluated using a fixed test frequency. For the design shown in Fig. 1(b), with five power switches per segment, we first simulated the test frequency. The signal "TE" (test enable) is set to 1, "D" (control signal for Discharge Transistors) is set to 0, the power switch is turned on ("S_1" = 0), and the fall time at the output of NAND gate is observed when it reaches 20% of V_{dd} (0.2 V). This fall time is used to determine the test frequency. Using the same test frequency, we determined the maximum number of faulty (stuck-open) power switches per segment, where segment size varieed from 5 to 30. The number of detectable power switches is then used to calculate diagnosis accuracy. The results are shown in Table I. As expected, for each design, the number of undetected faulty power switches increases with segment size, leading to reduced diagnosis accuracy. With a higher number of power switches per segment (e.g., 10 or more in case of C432 and C1908), the test time and hardware to control power switches reduce, but it also reduces diagnosis accuracy. This is because, with a higher number of power switches per segment, the voltage on virtual rail gets to 90% of V_{dd} within specified time hiding faulty power switches. This clearly shows that, for a given design, number of power switches, and test frequency, there is an optimal segment size that has to be determined to achieve 100% diagnosis accuracy.

A second important parameter that affects diagnosis accuracy of power switches is the test frequency, which is discussed next through HSPICE simulation. For example, C432 benchmark design requires 30 power switches to achieve targeted $\leq$ 5% IR drop. For C432, we considered three different segment sizes: 5, 10, and 15. For each segment size, we determine the falling delay at the output of NAND gate (d_f) in fault-free design, and use that to determine test frequency. Next, we insert stuck-open faults in each segment and increase the test frequency (starting from $1/1.4 * d_f$ and with a step size of 10% of $1/d_f$) until 100% diagnosis accuracy is achieved. That is the test frequency at which even a single faulty power switch is detectable. The test frequency step size is increased from $1/1.4 * d_f$ as an illustration, as that showed good correlation between test frequency and segment size in case of C432 design. The results are shown in Table II. As expected, for each segment size, it is possible to achieve 100% diagnosis accuracy at a certain test frequency. Note that using a very high test frequency can potentially lead to yield loss due to excessive power consumption [13], [14]. This is why in this paper we optimize the segment size using the rated frequency of each design to be used as the test frequency. For each design, the rated frequency is determined through the Synopsys design compiler under timing constraints. This also minimizes the overhead of an additional DFT clock only for diagnosis. For each design, using its rated frequency, the optimal segment size is shown in Table III, where each design is synthesized using the 90-nm STMicroelectronics gate library. The second column shows the critical path length, and for these designs the number of gates in the critical path varies from 9 to 18 gates as in case of C2670 and C3540, respectively. The third column shows the operating frequency of each design as determined by Synopsys Design Compiler. The fourth column shows the total number of power switches needed for each design to achieve $\leq$ 5% IR drop target. The fifth column shows the segment size to achieve 100% diagnosis accuracy using the operating frequency as test frequency. It is calculated through an iterative algorithm that increases the number of power switches per segment until the diagnosis accuracy remains unaffected at nominal operating conditions (25 °C, 1 V, and without considering process variation). Fig. 2 shows a snapshot of HSPICE simulation as in case of C432 benchmark design. It can be seen that using six power switches per segment with one faulty (stuck-open) power switch, it is not possible to differentiate between faulty and fault-free design; this is why five power switches per segment is selected, where it is still possible to determine a single faulty power switch. The last column shows the total number of segments for each design. In this paper, we assume a voltage level of $\leq$ 0.2 V as logic-0, and voltage $\geq$ 0.8 V as logic-1. This is because delay faults can be computed using signal capture time and logic threshold voltage of the observation point (in this case the output of the NAND gate), shown in Fig. 1. When considering process variation with $\pm$ 3σ variation effects, logic threshold voltages of all gates in a gate library are within 20%–80% of V_{dd} [15]. This means for a rising transition, logic-1 is guaranteed at $V_{Out} \geq 0.8$ V. Similarly, logic-0 is guaranteed at $V_{Out} \leq 0.2$ V. Note that, when the rated frequency is lower than the maximum achievable frequency, it may increase the number of segments and a multiplexer is needed for each additional segment [Fig. 1(b)]. For example, in case of C432 design, the last row of Table II, shows 6.4 GHz as the maximum frequency to detect 15 power switches per segment, which is significantly higher than the rated frequency of 1.69 GHz (Table III). Therefore, an overhead of additional multiplexer per segment is preferred over additional logic for higher than rated clock frequency generation.

The setup describe above allows us to detect each segment with one or more faulty power switches in it. However, it is not possible to determine their number per segment. For example, as shown in Table III, the C7552 benchmark design requires 703 power switches, which are divided into 19 segments, where each segment contains 37

TABLE III
OPTIMIZED SEGMENT SIZES AT RATED OPERATING FREQUENCY

Design	Critical Path Length (ns)	Freq (GHz)	Total number of PS	Segment Size	Total Segments
C432	0.59	1.69	30	5	6
C1908	0.71	1.41	126	9	14
C2670	0.43	2.33	187	17	11
C3540	0.83	1.2	216	18	12
C7552	0.53	1.89	703	37	19

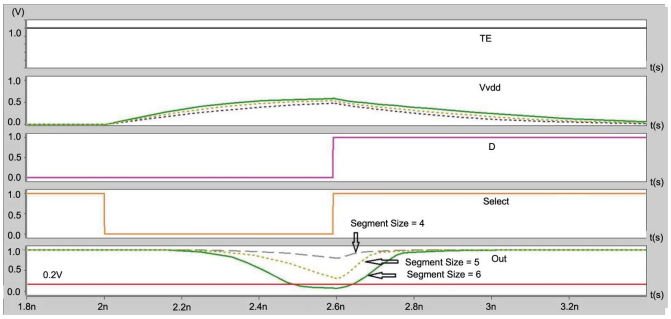


Fig. 2. HSPICE simulation to determine optimal segment size.

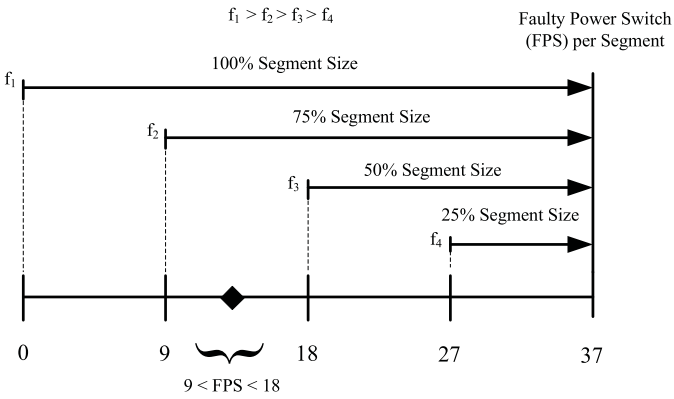


Fig. 3. Slower test frequencies used to detect the number of faulty power switches.

power switches. In case, there are 12 faulty power switches in one segment, and this setup allows us to identify that segment but it is not possible to determine the number of faulty power switches. This issue is addressed by exploiting the test frequencies to determine the number of faulty power switches per segment. This is achieved by using a slower than rated test frequency, such that it is possible to charge virtual rail V_{vdd} by using less number of power switches, i.e., by providing extra charge time by using a slower test frequency. In this paper, we have used three additional test frequencies to charge the virtual rail by using 25%, 50%, and 75% of the total number of power switches in a segment. Lower test frequencies can be used to determine the range of faulty power switches in a segment. This is demonstrated in Fig. 3, where f_1 denotes the rated frequency, and f_2, f_3, and f_4 are lower frequencies and are meant for 75%, 50%, and 25% of total power switches per segment, respectively. Using the same

TABLE IV
SLOWER THAN RATED TEST FREQUENCY TO DETERMINE THE NUMBER OF FAULTY POWER SWITCHES PER SEGMENT

Design	Segment Size	Prop. Segment Size	Number of Faulty PS	Test Freq (GHz)
C432	5	50%	3	0.63
C1908	9	50%	5	0.52
C2670	17	75%	4	1.65
		50%	8	0.95
		25%	12	0.53
C3540	18	75%	4	0.81
		50%	9	0.49
		25%	13	0.23
C7552	37	75%	9	1.25
		50%	18	0.72
		25%	27	0.3

example of C7552 benchmark design with 12 faulty power switches in one segment, it is possible to diagnose the range of faulty power switches with additional clock frequencies, i.e., between 9 and 18 faulty power switches as shown in Fig. 3. Note that it is possible to further narrow diagnosis accuracy using additional test frequencies. Table IV shows the slower than rated test frequencies for each design. In the case of the C7552 benchmark design, the rated test frequency is 1.89 GHz (last row of Table III), and slower than rated frequencies with proportional segment are 1.25 GHz for 75%, 0.72 GHz for 50%, and 0.3 GHz for 25% segment size.

III. DIAGNOSIS ALGORITHM

Algorithm 1 shows the proposed diagnosis scheme. As discussed in Section II, it is assumed that power switches are divided into segments using their corresponding rated frequency, such that even a single faulty power switch per segment is detectable. The number of faulty power switches per segment is identified by using three additional test frequencies per design. The rated frequency and the additional test frequencies per design are shown in Tables III and IV, respectively. The algorithm (Fig. 1) takes as input the netlist, number of segments "m," and test frequencies and returns diagnosis information consisting of the location and number of faulty power switches per segment in a design. The algorithm activates one segment ($S_i = 0$, $i \in [1, m]$) during each test cycle, while all others segments are switched off. Diagnosis is carried out through capturing the signal at the output of NAND gate [signal "OUT" in Fig. 1(b)]. The algorithm steps to test each segment are shown in lines 6–16, where starting from the first segment ($i = 1$), the highest test frequency (f_1) is applied first and the response is observed at "OUT." In case OUT $\neq 0$, the segment is identified as faulty, and lower test frequencies are then used to determine the number of faulty power switches for segment i. Once the number (range) of faulty power switches is identified, this information is stored on stack as shown by line 12. These steps (lines 6–16) are repeated for all segments and the algorithm terminates with complete diagnosis information stored on stack.

Algorithm 1 Diagnosis Algorithm

Input: (Netlist, m, f_1, f_2, f_3, f_4)
// *S*et f_1 to f_4, to test frequencies corresponding to 100%, 75%, 50%, and 25% segment size
// *d*iagnosis (Fig. 3). "m" is total number of segments [Fig. 1(b)]
Output: Faulty segments with number of faulty power switches in each segment
1: $TE = 1$
// TE is Test Enable [Fig. 1(b)]
2: $FF_1 = f_1$; $FF_2 = f_2$
// FF is Frequency at which a segment fails, and used to determine number of faulty power switches
3: $D = 1$
// Enable discharge transistor
4: $S_i = 1$; $i \in (1, M)$
// m is total number of segments [Fig. 1(b)]. Next, turn-off all power switch segments.
5: $i = 1$
// "i" points to the first segment
6: **repeat**
7: $TF = f_1$
// TF is Test Frequency
8: $S_i = 0$, $D = 0$
// Turn on only one segment at a time
9: **if** $Out == 0$ **then**default
// S_i is fault-free; Discharge virtual rail
10: **else**default
11: Use Lower Test Frequencies (f_2, f_3 and f_4) to determine the number of faulty power switches
// Stuck-open exists in S_i. Number of faulty power switches are located by FF_1 and FF_2
12: Push(i, FF_1, FF_2)
// *P*ush on Stack failed segment and the number of faulty power switches
13: **end if**
14: $S_i = 1$, $D = 1$
15: $i++$
16: **until** $i \leq m$
17: **return**

A. Test Set for Power-Gated Design

For the proposed DFT, it is necessary to test the discharge transistors and the power switches for two possible faults: stuck-open and stuck-short. This is because a stuck-open fault (transistor drain–source open) in a discharge transistor will result in long discharge time of the power switch, leading to a false test, while a stuck-short fault (transistor drain–source short) will lead to a stuck-at 0 fault at V_{Vdd}. Table V shows the test vectors to test a design using the DFT shown in Fig. 1(b) and assuming two segments $m = 2$. The first test cycle turns off both power switch segments (segments 1 and 2) and turns on the discharge transistors to discharge the voltage at V_{Vdd}. The second test cycle turns on all power switches in segment 1, while power switches in segment 2 and discharge transistors are switched off. This charges up the virtual supply node (V_{Vdd}) through the power switches in segment 1 and is used to test stuck-open on transistors of segment 1 and stuck-short on discharge transistors. The third test cycle turns off both power switch segments and turns on the discharge transistors to discharge the voltage at V_{Vdd} that was charged up in the previous test cycle. It is also used to test stuck-open fault on the discharge transistors. The fourth test cycle is used to test stuck-open on power switches in segment 2 by turning off power switches in segment 1 and discharge transistors, which is followed up test cycle no. 3 to discharge the virtual rail. Finally, the last test cycle turns off the discharge transistors to test for stuck-short at either of the power switch segments. In general, "$(2 * m) + 2$" test cycles are needed to test a design with m power switch segments and a discharge segment using the proposed DFT [Fig. 1(b)]. For designs with $m \geq 2$ power switch segments, first test cycle (Table V) should be repeated after applying the stuck-open test at each segment to discharge the voltage at V_{Vdd} and to prepare for the next test cycle.

TABLE V
TEST PATTERNS FOR TESTING POWER SWITCHES AND DISCHARGE TRANSISTORS USING THE PROPOSED DFT (FIG. 4) ASSUMING TWO SEGMENTS $m = 2$

Test cycle	TE = 1		D	V_{Vdd}	Out		Justification
	S_1	S_2			Fault free	Faulty	
1.	1	1	1	0	1	0	Discharge
2.	0	1	0	1	0	1	Seg. 1 Open *DT short
3.	1	1	1	0	1	0	Discharge DT open
4.	1	0	0	1	0	1	Seg. 2 open DT short
5.	1	1	1	0	1	0	Discharge DT open
6.	1	1	0	0	1	0	*PS Short turn-off DFT

*DT → Discharge transistor; PS → Power switch.

B. Control Logic

Fig. 4 shows the control logic implementation of a generic design with "m" power switch segments. Control logic implementation is based on three observations from the test vector sets shown in Table V. First, note that consecutive odd test cycles (1, 3, 5) are repeated. Second, the input to the discharge transistor segment "D" toggles between logic-1 and logic-0 in each consecutive test cycle. Finally, when testing power switch segments S1 and S2, consecutive even test cycles (2, 4, 6) shift logic-0 to right, starting from the first segment, while all other segment inputs are at logic-1. In total, there are "$(2 * m) + 2$" test cycles, which can be implemented by a state machine consisting of $\log_2(2 * m + 2)$ flip-flops. The first two observations are realized through a flip-flop (C_t) that toggles between logic-0 and logic-1 in consecutive test cycles, and its inverted output is used to control the discharge transistor segment through signal "D." The last observation can be realized by a simple m-bit wide shift right register

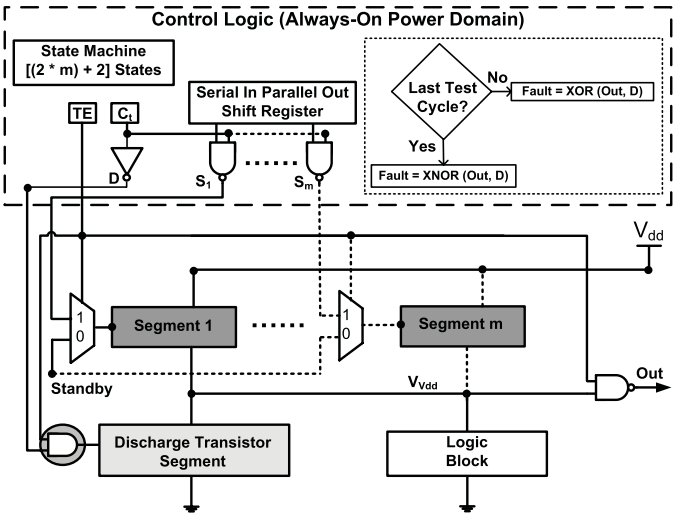


Fig. 4. Hardware implementation of control logic of a design with "m" power switch segments.

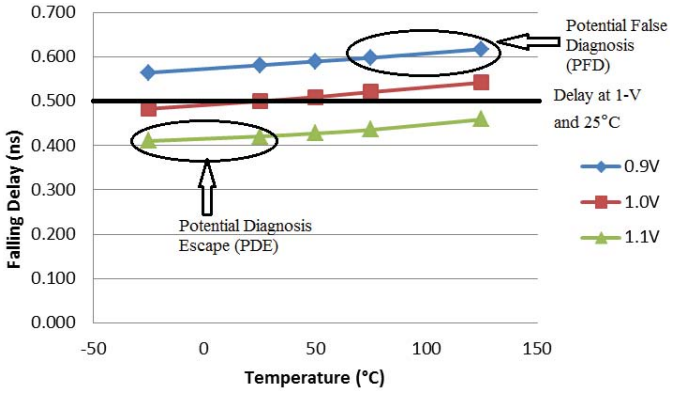


Fig. 5. Effect of VT variations on the diagnosis of power switches.

TABLE VI
POTENTIAL DIAGNOSIS ESCAPES

Design	Test Freq. (GHz)	Segment Size	Number of Undetected PS		Diag. Acc. at PDE	
			1.1 V V_{dd}			
			−25 °C	25 °C	−25 °C	25 °C
C432	1.69	5	0	0	100%	100%
C1908	1.41	9	1	1	88.9%	88.9%
C2670	2.33	17	2	2	88.2%	88.2%
C3540	1.2	18	2	1	88.9%	94.4%
C7552	1.89	37	4	2	89.2%	94.6%

(serial in, parallel out). To distinguish between faulty and fault-free values, note that the first "$(2 * m) + 1$" test cycles (all other than the last test cycle), the fault-free value is the same as input "D," and only for the last test cycle the fault-free value is "$\overline{D}$." This observation can be exploited to differentiate between faulty and fault-free values for each test cycle and requires only very small hardware overhead, which includes one XOR gate, one XNOR gate, and a comparator to determine the last test cycle. Therefore, the total hardware cost of the control logic implementation is $(1+ \text{m} + \log_2(2 \times m + 2))$ flip-flops, a comparator, m NAND gates, one inverter, one XOR, and one XNOR gate. The proposed solution is scalable and can be parallelized to reduce test time, as different power domains can be tested in parallel.

IV. Simulation Results

In this section, we first analyze the proposed diagnosis algorithm in the nominal scenario (without considering the effect of process variation) and then under the influence of process variation. In particular, we investigate the effect of two operating points that can potentially lead to loss of diagnosis accuracy, referred to as potential diagnosis escape (PDE) and potential false diagnosis (PFD). These two operating points are shown in Fig. 5. PDE refers to the operating point where, due to faster signal transition (than at 1.0 V, 25 °C), it is possible that a faulty power switch remains undetected by the diagnosis algorithm. PFD refers to the operating point where, due to slower signal transition (than at 1.0 V, 25 °C), it appears as if there is a defective power switch in a segment though it is fault-free. The results shown in Fig. 5 are generated using three operating voltages (0.9, 1.0, and 1.1 V; 10% variation of nominal V_{dd}) and, at each operating voltage, the delay is simulated at five operating temperatures (−25 °C–125 °C, with a step size of 25 °C) using the C432 benchmark design (Table III). Fall delay is simulated at the output of NAND gate ["OUT," Fig. 1(b)] using HSPICE. It can be seen (Fig. 5) that the fall delay is minimum at (1.1 V, −25 °C) and maximum at (0.9 V, 125 °C). This is because the transistor delay reduces as the operating voltage increases and it reduces further at lower temperatures [16]. This means that, when operating at 1.1 V and low temperatures, it is possible that fault effect is masked out by reduction in signal transition delay, leading to what is called "diagnosis escape." Similarly, when operating at 0.9 V and high temperatures, it is possible that the diagnosis algorithm incorrectly diagnoses ("false diagnosis") a design as faulty when it is actually fault-free, due to slow signal transition at this operating point. It should be noted that the segment size of each design, shown in Table III, is calculated at a given test frequency when operating at 1.0 V and 25 °C. This is why the diagnosis accuracy is 100% for all designs, when operating at 1.0 V and 25 °C, when considering both stuck-open and stuck-short faults. We conducted two sets of experiments to analyze the effect of PVT variations on the accuracy of proposed diagnosis algorithm, which are discussed next.

A. Nominal Scenario Under VT Variations

When considering the nominal scenario (without process variation), we evaluate the proposed algorithm at two operating points PDE and PFD, respectively, as shown in Fig. 5. To evaluate diagnosis accuracy at PDE, HSPICE simulations are conducted at 1.1 V V_{dd}, at two different operating temperatures: −25 °C and 25 °C. These points are selected because they are likely to show the highest number of diagnosis escapes (Fig. 5). To determine diagnosis accuracy, we insert faulty (stuck-open) power switches per segment, and increase the

TABLE VII
POTENTIAL FALSE DIAGNOSIS

Design	Test Freq. (GHz)	Segment Size	Number of Faulty PS at PFD		Diag. Acc. at PFD	
			0.9 V V_{dd}			
			75 °C	125 °C	75 °C	125 °C
C432	1.69	5	0	0	100%	100%
C1908	1.41	9	0	0	100%	100%
C2670	2.33	17	1	1	94.1%	94.1%
C3540	1.2	18	0	0	100%	100%
C7552	1.89	37	1	2	97.3%	94.6%

number of faulty power switches until the fault is detectable. For each of the two temperature settings, we report the number of undetectable power switches and the resulting diagnosis accuracy. The results are shown in Table VI. It can be seen that there are no escapes for one design (C432), leading to 100% diagnosis accuracy. Diagnosis accuracy reduces with a higher number of power switches per segment as in case of other designs, and C7552 has the highest number of diagnosis escapes as it has the largest segment size. As expected, when operating at 1.1 V V_{dd}, reduction in accuracy is higher at -25 °C (up to 12%) than at 25 °C.

PFD is simulated next by changing the operating point to take into account the effect of slow signal transition on the diagnosis accuracy. Under these operating conditions, the accuracy of the diagnosis algorithm is evaluated without using any faulty power switch in the design. In this case, the design operates at 0.9 V V_{dd} and we have considered two operating temperatures, 75 °C and 125 °C. These selected points are likely to show the effect of false diagnosis (Fig. 5). The results are shown in Table VII. As expected, diagnosis accuracy reduces at the higher temperature; when comparing the results for all designs at these two temperature settings, diagnosis accuracy is better at 75 °C than at 125 °C. Diagnosis accuracy is affected by the number of power switches per segment, and it reduces with a larger number of power switches per segment, as in the case of C7552, where two power switches are incorrectly diagnosed to be faulty. These results (Tables VI and VII) clearly show that diagnosis accuracy is affected by the combined effect of voltage and temperature variations and that designs with a higher number of power switches per segment (17 or more) are more likely to be affected than designs with smaller segment sizes. This means higher diagnosis accuracy across all these operating conditions and designs is possible by reducing the segment size (10 or less), but that will increase the diagnosis time due to increase in total number of segments (per design), where only one segment is tested at a time. In general, "$(2*m)+2$" test cycles are needed to test a design with m power switch segments and a discharge segment using the DFT shown in Fig. 1(b).

B. Process Variation

Fig. 6 shows the simulation setup for analyzing the effect of process variation on diagnosis escapes and false diagnosis using the proposed diagnosis algorithm (Algorithm 1).

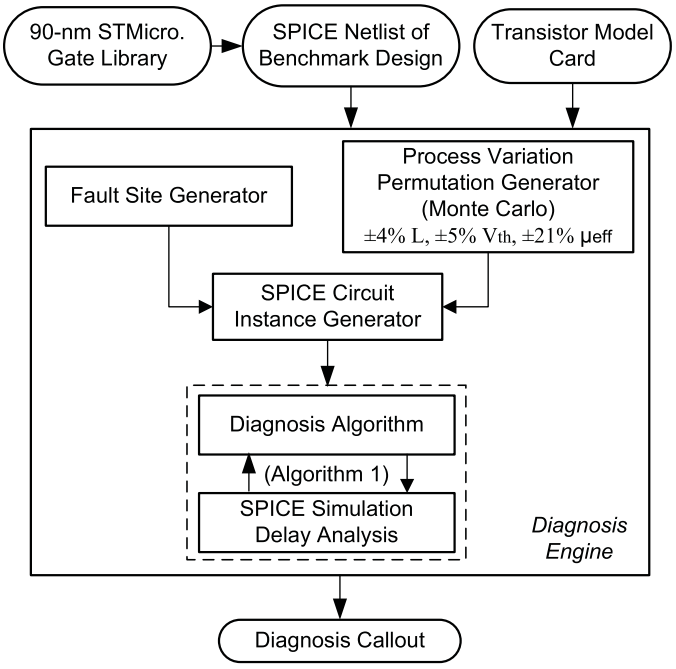


Fig. 6. Simulation setup to analyze the effect of process variation on the diagnosis accuracy of the proposed algorithm.

It takes as input the transistor model card and SPICE netlist of each benchmark design, generated through Synopsys design compiler and Synopsys STAR-RCXT using 90-nm ST-Microelectornics gate library. The output of simulation flow is marked as diagnosis callout, which specifies the location and number of faulty power switches. The simulation engine has four main blocks as shown in Fig. 6. The effect of process variation is incorporated by the process variation permutation generator, which uses the results reported in a recent study to incorporate the effect of process variation [17]. That study recognized three transistor parameters as the leading sources of process variation: gate length (L), threshold voltage (V_{th}), and the effective mobility (μ_{eff}).[1] These parameters follow a Gaussian distribution ($\pm 3\ \sigma$ variation) with standard deviations of 4% for L, 5% for V_{th}, and 21% for μ_{eff}. Negligible spatial correlation is found between these parameters, i.e., they can be treated as independent random variables following Gaussian distribution [17]. Note the parameter fluctuations (correlated or otherwise) do not imply that these parameters are independent; for example, as L decreases, V_{th} also decreases, and this effect is also known as V_{th} roll-off [16]. In total, 600 permutations per design are generated through Monte Carlo simulation. The number of permutations are based on a recent study, which shows that the probability of generating a unique logic fault follows the law of diminishing returns, as it reduces significantly after 500 permutations [15]. Fault-site generator is used to insert a faulty power switch at a random location in the design. This randomly inserted (power switch) fault and process variation permutation (generated through Monte Carlo) are used to create a transistor-level

[1]Mobility varies because of variation in effective strain in a strained silicon process [17].

TABLE VIII
EFFECT OF PROCESS VARIATION ON FALSE DIAGNOSIS

Design	Nominal Scenario Voltage (mV)	Process Variation Voltage (mV)			% False Diagnosis
		Max	Min	Avg	
C432	48.9	84.7	31.2	49.8	0%
C1908	66.9	123	48	68	0%
C2670	99.2	206	58.3	102.8	0.3%
C3540	112.7	230.1	77.4	118.1	0.17%
C7552	120.3	267.6	76.6	124.8	1.2%

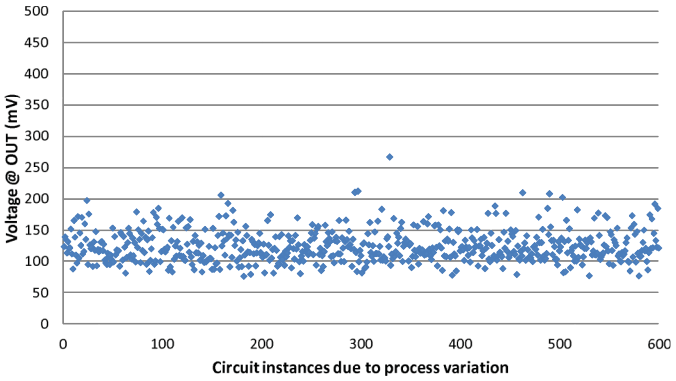


Fig. 7. Instances of false diagnosis in case of C7552 benchmark design over 600 process variation permutations.

TABLE IX
EFFECT OF PROCESS VARIATION ON DIAGNOSIS ESCAPES

Design	Nominal Scenario Voltage (mV)	Process Variation Voltage (mV)			% Diagnosis Escapes
		Max	Min	Avg	
C432	293.3	827.4	181.7	484.7	0.2%
C1908	304.1	840.2	236.5	476.7	0%
C2670	279.7	817.3	162.6	453.3	0.5%
C3540	202.1	629.8	171.9	398.4	4.5%
C7552	226.2	775.9	198.5	505.7	0.3%

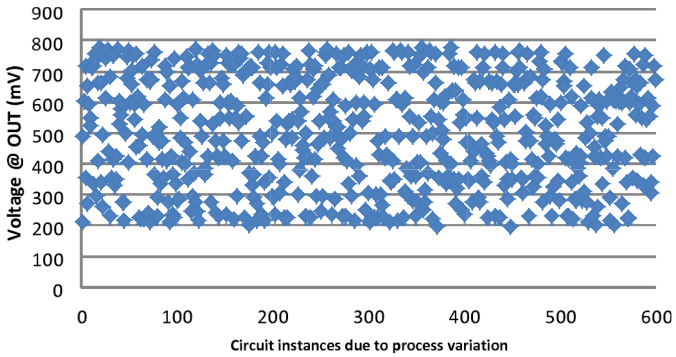


Fig. 8. Instances of diagnosis escapes in case of C7552 benchmark design over 600 process variation permutations.

SPICE circuit instance, which is fed to the diagnosis algorithm (Algorithm 1), which in turn provides the number and location of faulty power switches, as shown in Fig. 6.

We used this setup to conduct two experiments to analyze the effect of process variation on false diagnosis and diagnosis escapes by simulating transition delay at the output of the NAND gate ["Out," Fig. 1(b)] for all benchmark designs. In case of false diagnosis, we simulated fault-free transition delay under the influence of $\pm$ 3σ parameter variation, i.e., without inserting any fault. In case of diagnosis escapes, we inserted one stuck-open fault randomly per circuit instance to determine the accuracy of the proposed diagnosis algorithm. One stuck-open fault was inserted, as that was likely to show highest percentage of diagnosis escapes. Both experiments were conducted at 1.0 V and 25 °C. Table VIII shows the results of simulating false diagnosis under the influence of process variation, without inserting any fault in the design. The first column lists the benchmark designs; the second column shows the voltage at the output of NAND gate ["Out," Fig. 1(b)] in nominal scenario (without process variation); and the third column shows the maximum, minimum, and average voltage values at the output of NAND gate when considering the effect of process variation. The last column shows the overall percentage of false diagnosis observed. It can be seen that the percentage of false diagnosis is negligible for most designs, and its contribution goes up to 1.2% when considering all designs. Fig. 7 shows the detailed simulation results of C7552 benchmark design under the influence of process variation, as C7552 has the highest number of false diagnosis. It can be seen that only in 7 out of 600 instances the voltage is above 0.2 V, which is marked as false diagnosis. In this paper, we assume a voltage level of ≤ 0.2 V as logic-0, and voltage ≥ 0.8 V as logic-1. The proposed diagnosis method can be easily adjusted to match other voltage levels to correspond to logic (high and low) levels (for example, voltage ≤ 0.1 V as logic-0), through segment size adjustment of each design.

In the second experiment, to evaluate the effect of process variation on diagnosis escapes, we inserted $\leq 25\%$ stuck-open faults in a randomly selected segment in each of the 600 circuit instances of each design generated to simulate the effect of process variation (Fig. 6). The results are shown in Table IX, where for each design the second column shows the observed voltage at the output of NAND gate ["Out," Fig. 1(b)] in nominal scenario, without considering the effect of process variation. The third column shows the maximum, minimum, and average voltage values observed at the output of NAND gate when considering the effect of process variation. The last column shows the percentage of diagnosis escape for each design over 600 permutation instances. It can be seen that this percentage is small (up to 4.5% as in case of C3540), and in the rest of the cases it is less than 1%. Fig. 8 shows the detailed simulation results of C7552 benchmark design under the influence of process variation. It can be seen that only in very few (0.3%) instances, the voltage is below 0.2 V, which is marked as diagnosis escape. In general, when considering all designs, those with smaller segment sizes (< 10; Table III), as in case of C432 and C1908, show minimum diagnosis escapes and false diagnosis when considering the effects of PVT variation. This observation can be exploited to further reduce the effect of PVT variations on diagnosis escapes and false diagnosis. From this experiment, we conclude that

process variation has little effect on diagnosis accuracy of the proposed method. This is due to two reasons: first, power switches are designed to reduce leakage power, which is why these are long-channel transistors with $W = 1.1$ μm and $L = 150$ nm. It is well known that the effect of process variation is smaller on long-channel devices. See [2] for details. Second, the DFT setup shown in Fig. 1(b) allows explicit testing of power switches, and it is further facilitated by dividing the power switches into segments and testing one segment at a time. This approach is different from available techniques that simulate logic circuit delay at primary outputs or scan outputs (implicit testing) to test and diagnose power switches using high-switching-activity test patterns.

V. Conclusion

We demonstrated an efficient diagnosis method to identify the location and number of faulty power switches in a design. It utilizes an efficient DFT solution for testing power switches. The proposed method divided power switches into segments and uses the transition delay test to achieve very high diagnosis accuracy. The diagnosis method was validated through SPICE simulation using a number of ISCAS benchmarks synthesized with a 90-nm gate library. Experimental results showed that, under nominal operating conditions (at 1.0 V, 25 °C, and without considering process variation), it could achieve nearly 100% accuracy. In case of VT variations, the worst case loss of accuracy was less than 12%, and finally under the influence of process variation, the worst case loss of accuracy was less than 4.5%. Our continued work on this topic includes a low-cost online test strategy for power switches including discharge transistors to improve their in-field reliability.

References

[1] K. Roy, S. Mukhopadhyay, and H. Mahmoodi-Meimand, "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," *Proc. IEEE*, vol. 91, no. 2, pp. 305–327, Feb. 2003.

[2] M. Keating, D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual for System-on-Chip Design*. New York, USA: Springer-Verlag, 2007.

[3] Z. Zhang, X. Kavousianos, Y. Luo, Y. Tsiatouhas, and K. Chakrabarty, "Signature analysis for testing, diagnosis, and repair of multi-mode power switches," in *Proc. Eur. Test Symp.*, May 2011, pp. 13–18.

[4] Z. Zhang, X. Kavousianos, Y. Tsiatouhas, and K. Chakrabarty, "A bist scheme for testing and repair of multi-mode power switches," in *Proc. On-Line Test. Symp.*, Jul. 2011, pp. 115–120.

[5] S. Khursheed, S. Yang, B. Al-Hashimi, X. Huang, and D. Flynn, "Improved dft for testing power switches," in *Proc. 16th IEEE Eur. Test Symp.*, May 2011, pp. 7–12.

[6] J. Waicukauski and E. Lindbloom, "Failure diagnosis of structured VLSI," *IEEE Design Test Comput.*, vol. 6, no. 4, pp. 49–60, Aug. 1989.

[7] M. Abramovici, M. Breuer, and A. Friedman, *Digital Systems Testing and Testable Designs*. Piscataway, NJ, USA: IEEE Press, 1998.

[8] S. Khursheed, B. Al-Hashimi, S. Reddy, and P. Harrod, "Diagnosis of multiple-voltage design with bridge defect," *IEEE Trans. Comput.-Aided Design*, vol. 28, no. 3, pp. 406–416, Mar. 2009.

[9] S. K. Goel, M. Meijer, and J. de Gyvez, "Testing and diagnosis of power switches in SOCs," in *Proc. Eur. Test Symp.*, May 2006, pp. 145–150.

[10] M. Kassab and M. Tehranipoor, "Test of power management structures," in *Power-Aware Testing and Test Strategies for Low Power Devices*, P. Girard, N. Nicolici, and X. Wen, Eds. New York, USA: Springer-Verlag, Nov. 2009, ch. 10.

[11] H.-H. Huang and C.-H. Cheng, "Using clock-vdd to test and diagnose the power-switch in power-gating circuit," in *Proc. IEEE VLSI Test Symp.*, May 2007, pp. 110–118.

[12] K. Shi, Z. Lin, and Y.-M. Jiang, "A power network synthesis method for industrial power gating designs," in *Proc. 8th Int. Symp. Qual. Electron. Design*, Mar. 2007, pp. 362–367.

[13] P. Girard, "Survey of low-power testing of VLSI circuits," *IEEE Design Test Comput.*, vol. 19, no. 3, pp. 80–90, May–Jun. 2002.

[14] S. Kundu and A. Sanyal, "Power issues during test," in *Power-Aware Testing and Test Strategies for Low Power Devices*, P. Girard, N. Nicolici, and X. Wen, Eds. New York, USA: Springer-Verlag, Nov. 2009, ch. 2.

[15] S. Zhong, S. Khursheed, and B. Al-Hashimi, "A fast and accurate process variation-aware modeling technique for resistive bridge defects," *IEEE Trans. Comput.-Aided Design*, vol. 30, no. 11, pp. 1719–1730, Nov. 2011.

[16] *BSIM4.6.4, Manual*. (2012 Mar.) [Online]. Available: http://www-device.eecs.berkeley.edu/~bsim/Files/BSIM4/BSIM464.zip

[17] W. Zhao, F. Liu, K. Agarwal, D. Acharyya, S. Nassif, K. Nowka, and Y. Cao, "Rigorous extraction of process variations for 65-nm CMOS design," *IEEE Trans. Semicond. Manuf.*, vol. 22, no. 1, pp. 196–203, Feb. 2009.

Saqib Khursheed received the B.E. degree in computer engineering from NED University, Karachi, Pakistan, the M.Sc. degree from King Fahd University, Dhahran, Saudi Arabia, and the Ph.D. degree in electronics and electrical engineering from the University of Southampton, Southampton, U.K., in 2001, 2004, and 2010, respectively.

He is currently a Senior Research Fellow with the School of Electronics and Computer Science, University of Southampton. From 2005 to 2007, he was a Lecturer with the King Fahd University of Petroleum and Minerals, Dhahran. His current research interests include design, test, reliability, and yield improvement of low-power, high-performance, multicore designs and 3-D ICs.

Kan Shi received the B.E. degree in automation from the Beijing Institute of Fashion Technology, Beijing, China, and the M.S. degree (Distinction) from the School of Electronics and Computer Science, University of Southampton, Southampton, U.K., in 2010 and 2011, respectively. He is currently pursuing the Ph.D. degree with the Electrical and Electronics Engineering Department, Imperial College London, London, U.K.

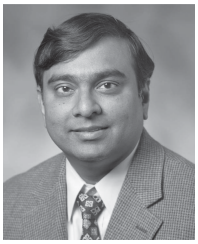
His current research interests include numerical analysis, high-performance and energy-efficient circuit design, and optimization of numerical algorithms.

Bashir M. Al-Hashimi (M'99–SM'01–F'09) received the B.Sc. degree (First Class) in electrical and electronics engineering from the University of Bath, Bath, U.K., and the Ph.D. degree from York University, York, U.K., in 1984 and 1989, respectively.

He was involved in research in the microelectronics design industry. In 1999, he joined the School of Electronics and Computer Science, University of Southampton, Southampton, U.K., where he is currently a Professor of computer engineering, the Director of the Pervasive Systems Center, the ARM Professor of Computer Engineering, the Co-Director of the ARM-ECS research centre, and the Associate Dean of Research with the Faculty of Physical and Applied Sciences. His current research interests include methods, algorithms, and design automation tools for low-power design and test of systems-on-chip and embedded computing systems.

Peter R. Wilson (M'99–SM'06) received the B.Eng. (Hons.) degree in electrical and electronic engineering from Heriot-Watt University, Edinburgh, Scotland, the M.B.A. degree from Edinburgh Business School, Edinburgh, and the Ph.D. degree from the University of Southampton, Southampton, U.K., in 1988, 1999, and 2002, respectively.

He is currently a Reader of electronic and electrical engineering with the School of Electronics and Computer Science, University of Southampton. His current research interests include the modeling of magnetic components in electric circuits, power electronics, renewable energy systems, integrated circuit design, VHDL-AMS modeling and simulation, and development of electronic design tools.



Krishnendu Chakrabarty (M'92–SM'00–F'08) received the B.Tech. degree from the Indian Institute of Technology Kharagpur, Kharagpur, India, in 1990, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, MI, USA, in 1992 and 1995, respectively.

He is currently a Professor of electrical and computer engineering with Duke University, Durham, NC, USA. He is the Chair Professor of Software Theory with Tsinghua University, Beijing, China, a Visiting Chair Professor of Computer Science and Information Engineering with National Cheng Kung University, Tainan City, Taiwan, and a Guest Professor with the University of Bremen, Bremen, Germany. He has led projects on wireless sensor networks, embedded systems, and real-time operating systems. His current research interests include testing and design-for-testability of integrated circuits, digital microfluidics, biochips, and cyberphysical systems, optimization of digital print and enterprise systems.

Prof. Chakrabarty was a recipient of the National Science Foundation Early Faculty (CAREER) award, the Office of Naval Research Young Investigator Award, the Humboldt Research Fellowship from the Alexander von Humboldt Foundation, Germany, and several Best Papers Awards at IEEE conferences.