

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 part of power-gating technique to reduce leakage power of a design. To the best of our knowledge, this is the first work in open-literature to show a systematic diagnosis method for accurately diagnosing power switches. The proposed diagnosis method utilizes recently proposed DFT solution for efficient testing of power switches in the presence of PVT variation. It divides power switches into segments such that any faulty power switch is detectable thereby achieving high diagnosis accuracy. The proposed diagnosis method has been 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%; and the worst case loss of accuracy is less than 12% when considering VT (Voltage and Temperature) variations.

Index Terms—Sleep transistor, diagnosis, power gating, leakage power management, design for test (DFT).

I. INTRODUCTION AND RELATED WORK

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 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. 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 coarse-grain design style, a number of power-switches are combined to feed a block of logic. When comparing the two design styles, fine-grain design simplifies the incorporation of power-gating through existing EDA tools, but it has higher area overhead and it is more vulnerable to voltage drop fluctuations due to process, voltage and temperature variations [2]. Therefore coarse-grain design style is a more popular design choice in

practice and is the focus of this work. Power-switches are implemented in two power-modes that provide a trade-off 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). DFT solutions for power switches with intermediate power-off mode have been recently proposed [3], [4]. Therefore, this work focuses on power-switches with 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, 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 compliments 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, that monitors the current flow during steady-state condition [7]. The first DFT solution is reported in [9], and is 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 power switches are turned off. This leads to long test time due to the necessity of applying 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 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 [5] (Sec. 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. (1a) 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 time and eliminates the possibility of

Manuscript received 28th Mar, 2012; revised 10th Oct, 2012; accepted 30th Dec, 2012. The authors would like to acknowledge Engineering and Physical Sciences Research Council (EPSRC, UK) for funding this work under grant no. EP/H011420/1.

S. Khursheed, B. M. Al-Hashimi and P. R. Wilson are with the School of Electronics and Computer Science, University of Southampton, UK. (email: 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 at Imperial College London, UK. (email: k.shi11@imperial.ac.uk)

K. Chakrabarty is with the department of Electrical and Computer Engineering, Duke University, USA. (email: krish@ee.duke.edu)

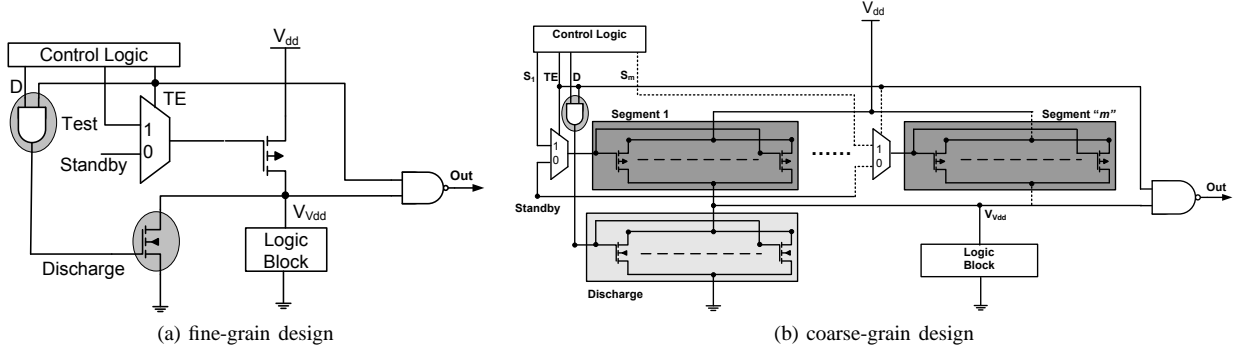


Fig. 1: DFT for testing power-switches [5]

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 higher 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 delay fault caused by power switches, logic gates or interconnects on the faulty paths, and overall negatively affects 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 coarse-grain design style (Fig. 1b), 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 as segment size. For each design, segment size is determined through detailed trade-off analysis between test frequency and 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 in nominal operating conditions and under the influence of process, voltage and temperature variations. It is shown that in the worst-case, loss of diagnosis accuracy is less than 12%. To the best of our knowledge, this is the first work on diagnosis of power switches that shows a systematic diagnosis method for accurately diagnosing power-switches in the presence of process, voltage and temperature variation.

The 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 the 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. 1b 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 power-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 active mode by simulating the voltage at V_{dd} , while feeding transition pulses (high-to-low and low-to-high) to the primary inputs of each design. The number of discharge transistors are 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 the voltage level to reach 90% of V_{dd} and the discharge time is the time it takes 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 has a trade-off between area overhead, test time and precision in identifying faulty transistors [5], [9], [10]. This is shown in Table I, where for each design, we varied segment size from 5 to 30, and evaluated diagnosis accuracy using a fixed test frequency. For a design shown in Fig. 1b, with 5 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 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

TABLE I: Trade-off: Diagnosis Accuracy vs. Segment Size

Design	Total # of PS	Segment Size	Detectable 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%

varies from 5 to 30. The number of detectable power switches are then used to calculate diagnosis accuracy. Results are shown in Table I, as expected, for each design, the number of un-detected faulty power switches increases with segment size leading to reduced diagnosis accuracy. With higher number of power switches per segment (for example, 10 or more in case of C432 and C1908), test time and hardware to control power switches reduces but it also reduces diagnosis accuracy. This is because, with 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, which has to be determined to achieve 100% diagnosis accuracy.

Second important parameter that affects diagnosis accuracy of power switches is 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 determined the falling delay at the output of NAND gate (d_f) in fault-free design, and used that to determine test frequency. Next, we inserted stuck-open faults in each segment and increased test frequency (starting from $\frac{1}{1.4*d_f}$ and step size is 10% of $\frac{1}{d_f}$) until 100% diagnosis accuracy is achieved. That is the test frequency at which even a single faulty power switch is detectable. Test frequency step size is increased from $\frac{1}{1.4*d_f}$ as an illustration, as that showed good correlation between test frequency and segment size in case of C432 design. The

TABLE II: Trade-off: Diagnosis Accuracy vs. Test Frequency

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

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 very high test frequency can potentially lead to yield loss due to excessive power consumption [13], [14]. This is why, in this work we optimize segment size using rated frequency of each design to be used as test frequency. For each design, rated frequency is determined through Synopsys design compiler under timing constraints. This also minimizes the overhead of 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 90-nm STMicroelectronics gate library. Second column shows critical path length, for these designs the number of gates in critical path varies from 9 to 18 gates as in case of C2670 and C3540 respectively. Third column shows the operating frequency of each design as determined by Synopsys Design Compiler. Fourth column shows the total number of power switches needed for each design to achieve $\leq 5\%$ IR-drop target. 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 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 6 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 5 power switches per segment is selected, where it is still possible to determine single faulty power switch. The last column shows total number of segments for each design. In this work, we assumed voltage level of ≤ 0.2 -V as logic-0, and voltage ≥ 0.8 -V as logic-1. This is because delay faults can be computed using signal capture time and logic threshold voltage of observation point (in this case output of NAND gate), shown in Fig. 1. When considering process variation with $\pm 3\sigma$ 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, when rated frequency is lower than maximum achievable frequency, it may increase the number of segments and a multiplexer is needed for each additional segment (Fig. 1b). For example in case of C432 design, last row of Table II, shows 6.4 GHz as 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.

This setup (described above) allows us to detect each segment with one or more faulty power switches in it. However, it is not possible to determine the number of faulty power switches per segment. For example, as shown in Table III,

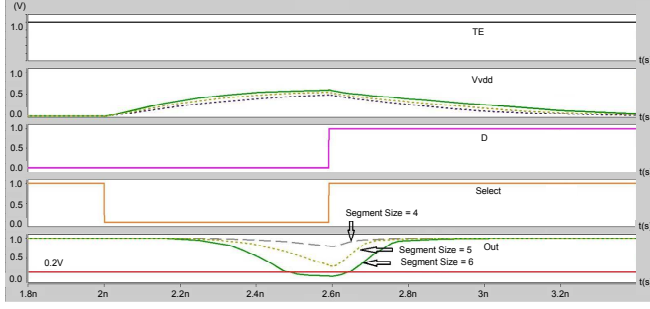


Fig. 2: HSPICE simulation to determine optimal segment size. C7552 benchmark design requires 703 power switches, which are divided into 19 segments, where each segment contains 37 power switches. In case, there are 12 faulty power switches in one segment, this setup allows us to identify that segment, but it is not possible to determine the number of faulty power switches. This issue is addressed by exploiting test frequencies to determine the number of faulty power switches per segment. This is achieved by using 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 slower test frequency. In this work, we have used 3 additional test frequencies, to charge virtual rail by using 25%, 50% and 75% of total power switches in a segment. Slower test frequencies can be used to determine the range of faulty power switches in a segment. This is demonstrated in Fig. 3, f_1 denotes the rated frequency, whereas f_2 , f_3 and f_4 are slower frequencies, and are meant for 75%, 50% and 25% of total power switches per segment respectively. Using the same 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, 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 case of 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 algorithm. As discussed in Sec. II, it is assumed that power switches are di-

TABLE III: Optimized Segment Sizes at rated operating Frequency

Design	Critical Path Length (ns)	Freq (GHz)	Total # 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

TABLE IV: Slower than rated test frequency to determine the number of faulty power switches per segment

Design	Segment Size	Prop. Segment Size	# 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

vided 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 are identified by using 3 additional test frequencies per design. The rated frequency and additional test frequencies per design are shown in Table III and Table IV respectively. The algorithm (Fig. 1) takes as input the netlist, number of segments “ m ”, and test frequencies and returns diagnosis information consisting of 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. 1b). Algorithmic steps to test each segment are shown in lines 6-16, where starting from the first segment ($i = 1$), highest test frequency (f_1) is applied first and 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 are identified, this information is stored on stack as shown by line 12. These steps (lines 6-16) are repeated for all segments and

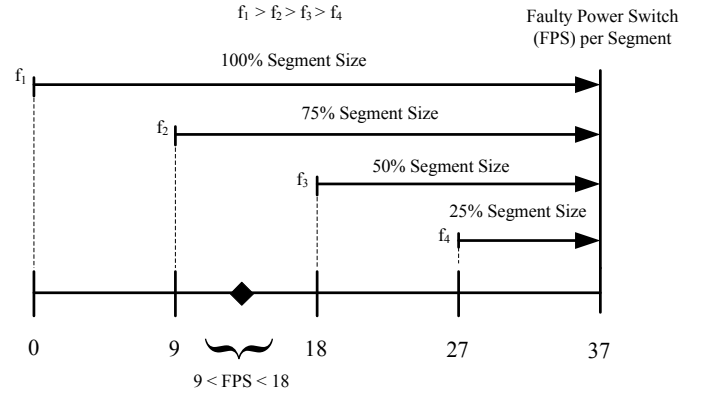


Fig. 3: Slower test frequencies are used to detect the number of faulty power switches

Algorithm 1 Diagnosis Algorithm

Input: (Netlist, m , f_1 , f_2 , f_3 , f_4)
 // Set f_1 to f_4 , to test frequencies corresponding to 100%, 75%, 50% and 25% segment size
 // diagnosis (Fig. 3). “ m ” is total number of segments (Fig. 1b)

Output: Faulty segments with number of faulty power switches in each segment

- 1: $TE = 1$
 // TE is Test Enable (Fig. 1b)
- 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. 1b). Next, turn-off all power switch segments.
- 5: $i = 1$
 // “ i ” points to 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**
 // S_i is fault-free; Discharge virtual rail
- 10: **else**
- 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)
 // Push 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**

the algorithm terminates with complete diagnosis information stored on stack.

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. 1b

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

and assuming two segments $m = 2$. The first test cycle turns-off both power switch segments (Segment 1 and Segment 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 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. 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 # 3 to discharge virtual rail. Finally, the last test cycle turns-off the discharge transistors to test for stuck-short at either of the two 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. 1b). For designs with $m \geq 2$ power switch segments, first test cycle (Table V) should be repeated after applying stuck-open test at each segment, to discharge the voltage at V_{Vdd} and to prepare for the next test cycle.

B. Control Logic

Fig. 4 shows control logic implementation of a generic design with “ m ” power switch segments. Control logic implementation is based on three observations from test vectors set shown in Table V. First, note that consecutive odd test cycles (1, 3, 5) are repeated. Second, note that the input to discharge transistor segment “D” toggles between logic-1 and logic-0 in each consecutive test cycle. Finally, note that when testing power switch segments (S_1 and S_2), 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

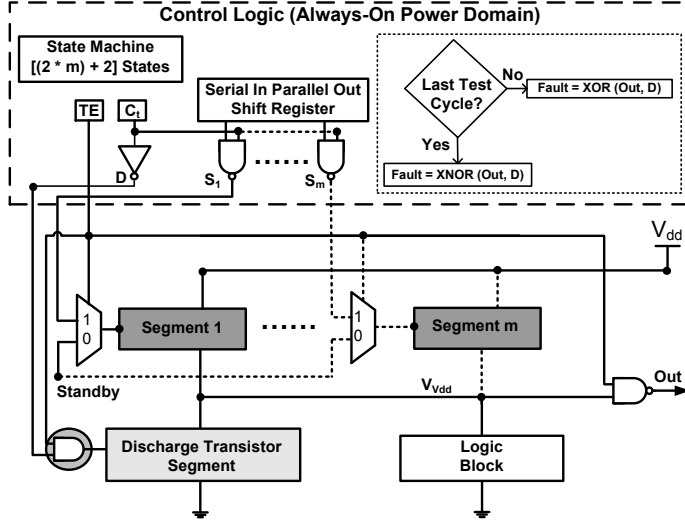


Fig. 4: Hardware Implementation of Control Logic of a design with “m” power switch segments.

by a state machine consisting of $\log_2(2 * m + 2)$ flip-flops. The first two observations are realized through a flip-flop (C_1) 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 (Serial In Parallel Out). To distinguish between faulty and fault-free values, note that first “ $(2 * m) + 1$ ” test cycles (all other than the last test cycle), fault-free value is the same as input “D”, and only for the last test cycle fault-free value is “ \bar{D} ”. This observation can be exploited to differentiate between faulty and fault-free values for each test cycle requiring very small hardware overhead, which includes one XOR gate, one XNOR gate, and a comparator to determine the last test cycle. Therefore, total hardware cost of control logic implementation is $(1 + m + \log_2(2 * m + 2))$ flip-flops, a comparator, m Nand gates, one inverter, one XOR and one XNOR gate. The proposed solution is scalable and it 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 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 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 1.0V, 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 1.0V, 25°C), it appears as if there is a defective power switch in a segment, when it is fault-free. The

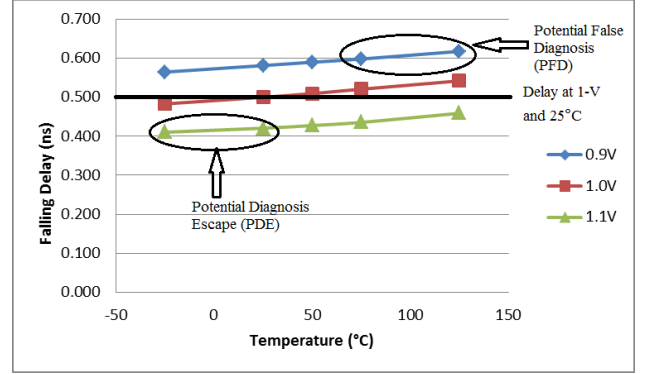


Fig. 5: Effect of VT variations on diagnosis of power switches

TABLE VI: Potential Diagnosis Escapes

Design	Test Freq. (GHz)	Segment Size	# 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%

results shown in Fig. 5 are generated using three operating voltage (0.9-V, 1.0-V, 1.1-V; 10% variation of nominal V_{dd}) and at each operating voltage, the delay is simulated at five operating temperatures (-25°C to 125°C, with step size of 25°C) using C432 benchmark design (Table III). Fall delay is simulated at the output of NAND gate (“OUT”, Fig. 1b) 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 operating voltage increases and it reduces further at lower temperatures [16]. This means 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 diagnose (“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 1.0V and 25 °C. This is why the diagnosis accuracy is 100% for all designs, when operating at 1.0V and 25 °C, when considering both stuck-open and stuck-short faults. We conducted two sets of experiments to analyze the effect of process, voltage and temperature variations on the accuracy of proposed diagnosis algorithm, which are discussed next.

A. Nominal Scenario under VT variations

When considering nominal scenario (without process variation), we evaluate the proposed algorithm at two operating points PDE (Potential Diagnosis Escapes) and PFD (Potential False Diagnosis) 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 inserted faulty (stuck-open) power switches per segment, and the number of faulty power switches is increased until the fault is detectable. For each of the two temperature settings, we report the number of undetectable power switches and 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 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 .

Potential false diagnosis (PFD) is simulated next by changing the operating point to take into account the effect of slow signal transition on diagnosis accuracy. In these operating conditions, accuracy of diagnosis algorithm is evaluated without using any faulty power switch in the design. In this case, the design is operating 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 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 higher number of power switches per segment, as in case of C7552, where 2 power-switches are incorrectly diagnosed to be faulty. These results (Table VI and Table VII) clearly show that diagnosis accuracy is affected by the combined effect of voltage and temperature variations and designs with 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 segment size (10 or less), but that will increase 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

TABLE VII: Potential False Diagnosis

Design	Test Freq. (GHz)	Segment Size	# 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%

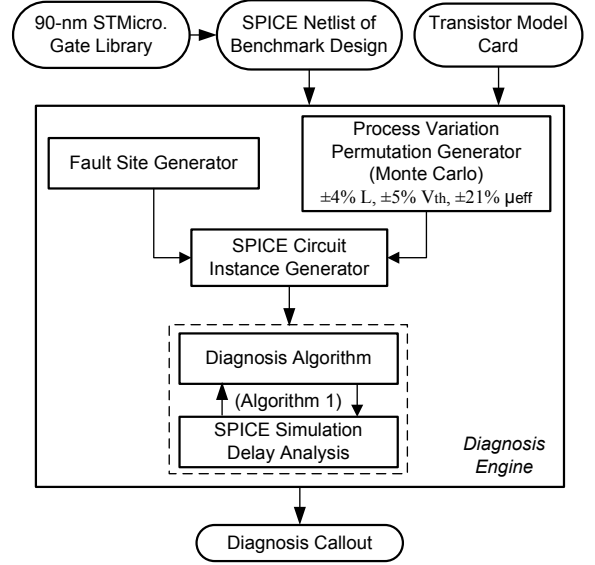


Fig. 6: Simulation setup to analyze the effect of process variation on diagnosis accuracy of the proposed algorithm.

with m power switch segments and a discharge segment using the DFT shown in Fig. 1b.

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). It takes as input transistor model card and SPICE netlist of each benchmark design, generated through Synopsys design compiler and Synopsys STAR-RCXT using 90-nm ST-Microelectronics 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]. This study has recognized three transistor parameters as leading sources of process variation, which include: gate length (L), threshold voltage (V_{th}), and effective mobility (μ_{eff})¹. These parameters follow 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, 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

¹Mobility varies due to 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%

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) is used to create a transistor-level 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 NAND gate (“Out”, Fig. 1b) for all benchmark designs. In case of false diagnosis, we simulated fault-free transition delay under the influence of $\pm 3\sigma$ 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 is likely to show highest percentage of diagnosis escapes. Both experiments are conducted at 1.0V 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. First column lists the benchmark designs, second column shows the voltage at the output of NAND gate (“Out”, Fig. 1b) in nominal scenario (without process variation), 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 majority of designs and its contribution is up to 1.2%, when considering all designs. Fig. 7 shows 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 work, we assumed 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

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%

faults in a randomly selected segment, in each of 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, second column shows the observed voltage at the output of NAND gate (“Out”, Fig. 1b) in nominal scenario, without considering the effect of process variation. 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 rest of the cases it is less than 1%. Fig. 8 shows 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 effect of process, voltage and temperature variation. This observation can be exploited to further reduce the effect of process, voltage and temperature 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 because of two reasons. Firstly, power switches are designed to reduce leakage power, which is why these are long channel transistors with $W=1.1\text{-}\mu\text{m}$ and $L=150\text{-nm}$. It is well-known that the effect

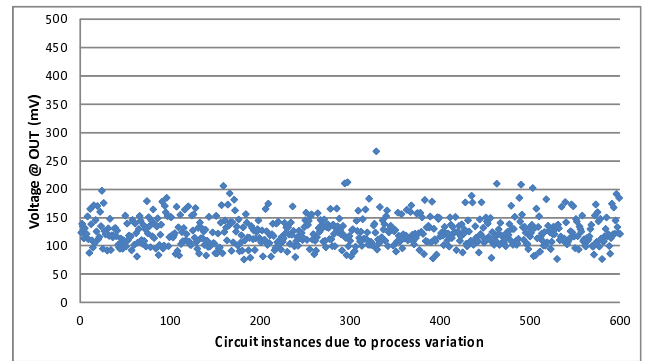


Fig. 7: Instances of false diagnosis in case of C7552 benchmark design over 600 process variation permutations.

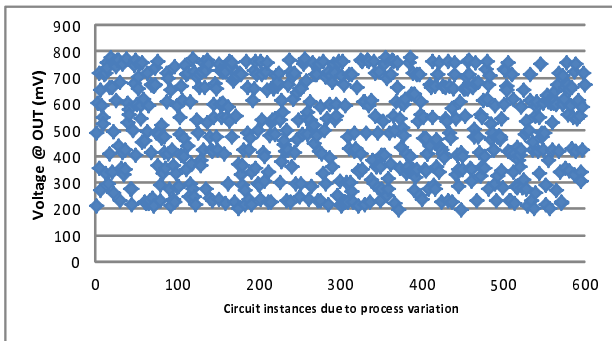


Fig. 8: Instances of diagnosis escapes in case of C7552 benchmark design over 600 process variation permutations.

of process variation is smaller on long-channel devices. See [2] for details. Secondly, the DFT setup shown in Fig. 1b allows explicit testing of power switches, and it is further facilitated by dividing 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

This work has 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 divides power switches into segments and uses transition delay test to achieve very high diagnosis accuracy. The diagnosis method is validated through SPICE simulation using a number of ISCAS benchmarks synthesized with a 90-nm gate library. Experimental results show that under nominal operating conditions (at 1.0-V, 25°C, and without considering process variation), it achieves nearly 100% accuracy. In case of VT (Voltage and Temperature) variations, the worst-case loss of accuracy is less than 12%, and finally under the influence of process variation, the worst case loss of accuracy is 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," *Proceedings of the IEEE*, Vol. 91, No. 2, Feb. 2003.
- [2] M. Keating, D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual for System-on-Chip Design*. Springer, 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 *European Test Symposium (ETS)*, May 2011.
- [4] Z. Zhang, X. Kavousianos, Y. Tsiatouhas, and K. Chakrabarty, "A bist scheme for testing and repair of multi-mode power switches," in *On-Line Testing Symposium (IOLTS)*, July 2011.
- [5] S. Khursheed, S. Yang, B. Al-Hashimi, X. Huang, and D. Flynn, "Improved dft for testing power switches," in *European Test Symposium (ETS)*, 2011 16th IEEE, May 2011.
- [6] J. Waicukauski and E. Lindbloom, "Failure Diagnosis of Structured VLSI," *IEEE Design & Test of Computers*, Vol. 6, No. 4, Aug. 1989.
- [7] M. Abramovici, M. Breuer, and A. Friedman, *Digital Systems Testing and Testable Designs*. IEEE Press, 1998.
- [8] S. Khursheed, B. Al-Hashimi, S. Reddy, and P. Harrod, "Diagnosis of multiple-voltage design with bridge defect," *IEEE Trans. on Computer-Aided Design*, Vol. 28, No. 3, Mar. 2009.
- [9] S. Kumar Goel, M. Meijer, and J. de Gyvez, "Testing and diagnosis of power switches in SOCs," in *ETS*, May 2006.
- [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. Springer, Nov. 2009, Chp. 10.
- [11] H.-H. Huang and C.-H. Cheng, "Using clock-vdd to test and diagnose the power-switch in power-gating circuit," in *Proceedings IEEE VLSI Test Symposium (VTS)*, May 2007.
- [12] K. Shi, Z. Lin, and Y.-M. Jiang, "A power network synthesis method for industrial power gating designs," in *ISQED*, Mar. 2007.
- [13] P. Girard, "Survey of low-power testing of vlsi circuits," *Design & Test of Computers, IEEE*, Vol. 19, No. 3, 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. Springer, Chp. 2, Nov. 2009.
- [15] S. Zhong, S. Khursheed, and B. Al-Hashimi, "A fast and accurate process variation-aware modeling technique for resistive bridge defects," *IEEE Trans. on CAD*, Vol. 30, No. 11, Nov. 2011.
- [16] BSIM4.6.4, Manual, Univ. of California, Berkeley, Mar. 2012. [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. on Semiconductor Manufacturing*, Vol. 22, No. 1, Feb. 2009.



Saqib Khursheed received the B.E. degree in Computer Engineering from NED University, Pakistan, in 2001, the M.Sc. degree in Computer Engineering from King Fahd University (KFUPM), Saudi Arabia, in 2004, and the Ph.D. degree in Electronics and Electrical Engineering from University of Southampton, U.K., in 2010.

From 2005 to 2007, he served as a Lecturer with KFUPM. Currently he is working as a Senior Research Fellow in the School of Electronics and Computer Science, University of Southampton. He is

interested in all issues related to design, test, reliability and yield improvement of low-power, high-performance, multi-core designs and 3D ICs.



Kan Shi received the B.E. degree in automation from BIFT, China in 2010, the M.S. degree with distinction from the School of Electronics and Computer Science, University of Southampton, U.K. in 2011. He is currently pursuing his Ph.D. degree with the Electrical and Electronics Engineering department at Imperial College 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 (with 1st-class classification) in Electrical and Electronics Engineering from the University of Bath, UK, in 1984 and the Ph.D. degree from York University, UK, in 1989. Following this he worked in the microelectronics design industry and in 1999, he joined the School of Electronics and Computer Science, Southampton University, UK, where he is currently a Professor of Computer Engineering and Director of the Pervasive Systems Center. He is ARM Professor of Computer Engineering,

and Co-Director of the ARM-ECS research centre. He is Associate Dean (Research) of the Faculty of Physical and Applied Sciences, University of Southampton, UK. His 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.) in Electrical and Electronic Engineering from Heriot-Watt University, Edinburgh, Scotland, in 1988; an M.B.A from the Edinburgh Business School, Scotland in 1999, and Ph.D. from the University of Southampton, UK, in 2002.

Dr Wilson is currently a Reader in Electronic and Electrical Engineering at the School of Electronics and Computer Science, University of Southampton, UK. His current research interests include modeling of magnetic components in electric circuits, power electronics, renewable energy systems, integrated circuit design, VHDL-AMS modeling and simulation, and the 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, in 1990, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, in 1992 and 1995, respectively. He is now Professor of Electrical and Computer Engineering at Duke University. He is also a Chair Professor in Software Theory at Tsinghua University, Beijing, China, a Visiting Chair Professor of Computer Science and Information Engineering at National Cheng Kung University in Taiwan, and a Guest Professor

at University of Bremen in Germany. Prof. Chakrabarty is 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. Prof. Chakrabarty's current research projects include: testing and design-for-testability of integrated circuits; digital microfluidics, biochips, and cyberphysical systems; optimization of digital print and enterprise systems. In the recent past, he has also led projects on wireless sensor networks, embedded systems, and real-time operating systems.