

X-Filling for Simultaneous Shift- and Capture-Power Reduction in At-Speed Scan-Based Testing

Jia Li, *Student Member, IEEE*, Qiang Xu, *Member, IEEE*, Yu Hu, *Member, IEEE*, and Xiaowei Li, *Senior Member, IEEE*

Abstract—Power consumption during at-speed scan-based testing can be significantly higher than that during normal functional mode in both shift and capture phases, which can cause circuits' reliability concerns during manufacturing test. This paper proposes a novel X-filling technique, namely “*iFill*”, to address the above issue, by analyzing the impact of X-bits on switching activities of the circuit nodes in the two different phases. In addition, different from prior X-filling methods for shift-power reduction that can only reduce shift-in power, our method is able to cut down power consumptions in both shift-in and shift-out processes. Experimental results on benchmark circuits show that the proposed technique can guarantee the power safety in both shift and capture phases during at-speed scan-based testing.

Index Terms—At-speed scan-based testing, low-power testing, X-filling.

I. INTRODUCTION

THE POWER dissipation of integrated circuits (ICs) in scan-based testing can be significantly higher than that during normal operation [1], [2]. This brings the following problems that threaten the reliability of the circuits under test (CUTs).

Manuscript received October 21, 2008; revised January 24, 2009; accepted March 03, 2009. First published August 11, 2009; current version published June 25, 2010. The work of J. Li, Y. Hu, and X. Li was supported in part by the National Natural Science Foundation of China (NSFC Program) under Grant 60633060, 60803031, and 90607010, and in part by the National High Technology Research and Development Program of China (863 program) under Grant 2007AA01Z107 and 2007AA01Z113, and in part by the National Basic Research Program of China (973 program) under Grant 2005CB321604 and 2005CB321605. The work of Q. Xu was supported in part by the General Research Fund CUHK417406, CUHK417807, and CUHK418708 from Hong Kong SAR Research Grants Council (RGC), in part by NSFC under Grant 60876029, in part by a grant N_CUHK417/08 from the NSFC/RGC Joint Research Scheme, and in part by the National High Technology Research and Development Program of China (863 Program) under Grant 2007AA01Z109. A preliminary version of this paper was published in Proceedings of IEEE/ACM Design, Automation, and Test in Europe (DATE), pp. 1184-1189, 2008.

J. Li is with the Institute of Computing Technology, Chinese Academy of Sciences, and the Graduate University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: gracelee@ict.ac.cn).

Q. Xu is with the CUHK Reliable Computing Laboratory (CURE), Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong. He is also with CAS-CUHK Shenzhen Institute of Advanced Integration Technology, Shenzhen 518055, China (e-mail: qxu@cse.cuhk.edu.hk).

Y. Hu is with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: huyu@ict.ac.cn).

X. Li is with the Key Laboratory of Computer System and Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lxw@ict.ac.cn).

Digital Object Identifier 10.1109/TVLSI.2009.2019980

- 1) The elevated average power consumption adds to the thermal load that must be transported away from the CUT and can cause structural damage to the silicon, bonding wires, or the package.
- 2) The excessive peak power dissipation is likely to cause a large voltage drop that may lead to erroneous data transfer in test mode only, especially in capture phase of at-speed testing, thus invalidating the testing process and leading to unnecessary test yield loss [2]–[4].

It is likely that a CUT's power rating is violated in both shift mode and capture mode in scan tests. A significant amount of research work has been proposed to address this problem in the literature, which can be broadly divided into two categories: design-for-testability (DfT)-based solutions [5]–[19] and software-based solutions [20]–[35]. Generally speaking, DfT-based solutions are more effective for test power reduction by introducing dedicated DfT hardware to suppress switching activities in the CUT. Software-based solutions, on the other hand, usually cannot achieve the same amount of test power reduction as that of DfT-based solutions, but they do not involve any DfT overhead and can be easily integrated into conventional IC design flow. In this work, we focus on one of the most widely-used software-based solutions for test power reduction, which tries to reduce the CUT's switching activities by filling the don't-care bits (i.e., X-bits) in given test cubes intelligently, known as the X-filling technique.

Various X-filling techniques have been proposed in the literature to reduce shift- and/or capture-power in scan-based testing. However, they either target only one type of power consumption (i.e., shift-power reduction [20] or capture-power reduction [21]–[24]) or do not consider the differences of these two types of power consumptions [25]. In addition, prior work on shift-power reduction (e.g., *Adjacent fill* [20]) considers the shift-in power consumption only, which, unfortunately, may lead to excessive power during the shift-out process.

In this paper, by investigating the impact of X-bits on test power consumption, we propose a novel X-filling technique, namely “*iFill*”, to reduce both shift- and capture-power during scan tests. In the proposed approach, first, we try to fill as few as possible X-bits to keep the capture-power under the peak power limit of the CUT to avoid test overkills, and then use the remaining X-bits to reduce shift-power as much as possible to cut down the CUT's average power consumption, so that designers are able to use higher shift frequency and/or increase test parallelism to reduce the CUT's test time and hence cut down the test cost. Moreover, the proposed X-filling technique is able to reduce power consumptions in both shift-in and shift-out

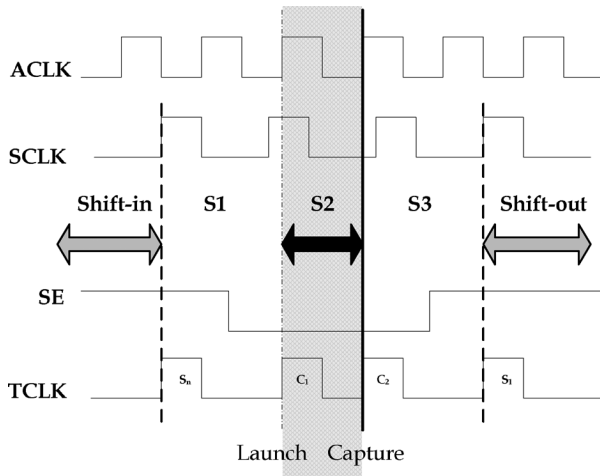


Fig. 1. Timing diagram for launch-on-capture at-speed tests.

processes, thus leading to significant shift-power reduction. To verify the effectiveness of the proposed *iFill* technique, we conduct experiments on various ISCAS'89 and ITC'99 benchmark circuits and the results show that our technique is superior to known techniques in the literature in terms of both shift- and/or capture-power reduction.

The remainder of this paper is organized as follows. Section II presents background of this work and Section III analyzes the impact of X-filling on shift- and capture-power in at-speed scan testing. In Section IV, the proposed *iFill* X-filling procedure is detailed. Experimental results on several benchmark circuits are presented in Section V. Finally, Section VI concludes this paper and lists some future research directions.

II. BACKGROUND

A. Power Consumption in At-Speed Scan-Based Testing

At-speed scan-based tests facilitate to detect speed-related defects of the CUTs and have been widely utilized in the industry in recent years, which typically involve a long low-frequency shift phase and a short at-speed capture phase. There are mainly two types of at-speed scan-based testing approaches: Launch-on-Shift (LoS) and Launch-on-Capture (LoC). LoC scheme is more widely utilized because it does not need the expensive high-speed scan-enable signal required by the LoS scheme. As shown in Fig. 1, there are three types of clock signals in LoC scheme: "SCLK" represents the shift clock signal, under which the test vectors are shift-in/out of the scan chains; "ACLK" is the at-speed clock in the CUT to be applied in the capture phase; "TCLK" is the clock signal that the sequential elements on the scan chain will receive, by MUXing "SCLK" and "ACLK" signals. Typically two capture cycles (c_1 and c_2) are used to detect defects. We denote the initial state of the scan cells and the nodes in combinational portion of the circuit before capture as $S1$. The first capture c_1 launches the state $S2$ into the CUT, while the second capture c_2 store the circuit state $S3$ after the CUT is applied in functional mode.

An example at-speed scan-based testing procedure is shown in Fig. 2. The first test vector "10001" is shifted into the scan

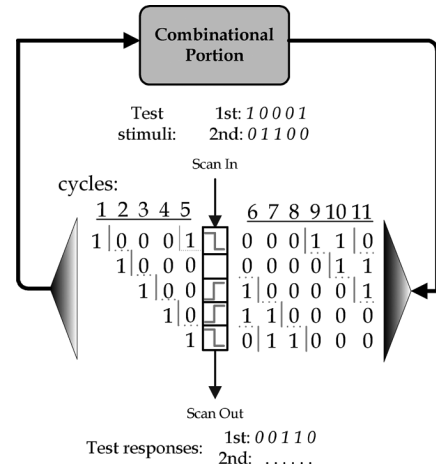


Fig. 2. Transitions in scan-based testing.

chain bit by bit in five clock cycles. After two capture cycles: c_1 and c_2 , the response vector "00110" is captured into the scan chain and shifted out while the next test vector "01100" is scanned in concurrently.

It is important to note that shift and capture phases have different impact on the CUT's power consumption and hence they should be dealt with differently.

In shift mode, switching activities occur in both sequential elements and combinational logic when adjacent bits in test vectors have different logic values, highlighted by the dash lines in Fig. 2. The shift process in scan-based testing not only dominates the test time of the CUT, but also determines the CUT's accumulated power dissipation. The main objective in shift-power reduction is thus to decrease it *as much as possible*, so that higher shift frequency and/or increase test parallelism can be applied to reduce the CUT's test time and hence cut down the test cost, under the average power constraint of the CUT.

In capture mode, on the other hand, since the duration is very short, it has limited impact on the CUT's accumulated test power consumption. However, because test vectors are usually generated to detect as many faults as possible, the excessive transitions in capture phase (highlighted as red waves in the scan cells in Fig. 2) may cause serious IR-drop and prolong circuit delay, thus causing false rejects (i.e., good chips fail the test) in at-speed tests. Consequently, the main objective in capture-power reduction is to keep it under a safe threshold to avoid test overkill. As long as this requirement is fulfilled, there is *no* need to further reduce capture-power.

B. Overview of Prior Work on Low-Power Testing

Reducing test power by modifying the circuit under test has been proposed by several research groups, including clock gating [5], scan chain segmentation, or combinational logic division [6]–[12], toggling suppression through circuitry insertion [13], [14], power reduction for built-in self-test (BIST) application [15]–[17], scan enable disabling [18], and circuit virtual partitioning [19]. The above solutions are usually quite effective for test power reduction, but they typically involve high DfT cost and are not readily applicable to regular IC design and test flow.

There are also many software-based techniques for test power reduction without incurring any DfT overhead. Test vector re-ordering is quite effective for shift-power reduction [27]. Power-constrained test scheduling is often conducted in core-based testing, in which we carefully select embedded cores that are tested simultaneously according to a given power budget [26], [36]. There are also many approaches that reduce the switching activities of the CUT by taking advantage of the X-bits in given test cubes, e.g., low-power automatic test pattern generation (ATPG) techniques in [28]–[31], the test compression strategy in [32] and [34], and the various X-filling techniques proposed recently in [20]–[25], and [35]. As X-filling techniques do not require to rerun the time-consuming application test pattern generation (ATPG) process and can work together with other DfT-based solutions to further reduce test power (if necessary), they have received lots of attention recently from both academia and industry [20]–[25].

C. X-Filling for Shift- and Capture-Power Reduction

It has been shown that test cubes for industrial circuits contain as much as 95%–98% X-bits, which can be filled with logic “0” or logic “1” freely without affecting the CUT’s fault coverage¹. It should also be noted that, even if the given tests are fully specified, the don’t-care bits can still be identified with techniques such as the one proposed in [37].

X-filling for shift-power reduction tries to generate fewer differences between adjacent scan cells. The so-called *weighted transition metric (WTM)* was proposed in [38] to estimate shift-power caused by these logic value differences. That is, the shift-power in the i th test vector is estimated as follows:

$$WTM_i = \sum_{j=1}^{N-1} (S_{i,j} \oplus S_{i,j+1}) \times j \quad (1)$$

where N is the number of scan cells in the scan chain, $S_{i,j}$ represents the logic value of the j th scan cell in this test vector.

Based on the previous formula, [20] proposed a simple X-filling method that fills X-bits according to the logic values of their adjacent scan cells for shift-in power reduction, namely *Adjacent fill*. For example, if the test vector is “01XX1XX0X”, it will be filled as “01111100”, and WTM of this test vector can be calculated as: $WTM_i = 2 + 8 = 10$, which has the minimum transitions among all the possible X-filling options for the shift-in process. Unfortunately, *Adjacent fill* cannot be applied to shift-out power reduction and may cause serious capture-power problem.

X-filling for capture-power reduction is quite different from the above. The CUT’s switching activities in capture mode is caused by logic value differences in scan cells before and after the capture cycle. Filling one X-bit in the test stimuli may cause several X-bits in test responses to turn into determined logic values (“1” or “0”). Therefore, we need to consider the impact on test responses when filling X-bits in the test stimuli.

Wen *et al.* [21] first addressed the low capture-power X-filling problem. They mainly considered the transitions at the output of

scan flip-flops (SFFs) during X-filling, which, however, does not necessarily have a good correlation with the total capture-power of the whole circuit (i.e., both FFs and combinational gates). Later, in [24], they took the above shortcoming into consideration and introduced a new method to select the X-filling target based on a so-called set-simulation technique, which is proved to be a more effective X-filling method with experimental results on ISCAS’89 circuits. One of the main limitations of [21] and [24] is that their computational time is quite high. This is because: 1) they are incremental filling approaches, that is, they fill the X-bits in the test cubes one by one and 2) forward implications and backward justification are extensively used in their methodologies.

In fact, the complexity of the set-simulation techniques proposed in [24] is quite high and it is difficult, if not impossible, to be applicable for at-speed two-pattern tests in industrial designs. In [23], Remersaro *et al.* developed an efficient probability-based X-filling technique, called *Preferred fill*, which tries to fill all X-bits in the test cube in one step, instead of using incremental fill and logic simulation. Their technique, however, is inherently less effective as the available information for the probability calculation in their single-step filling is quite limited. Also, only transitions at the SFFs are considered while the transitions at logic gates are ignored in their work.

The above X-filling techniques target either shift-power reduction or capture-power reduction, but not both. This is unfortunate, because filling these unspecified bits has impact on both shift- and capture-power. Because the number of X-bits in test cubes is limited but the objectives are different, low shift-power X-filling techniques may result in high capture-power dissipation, and vice versa. As a result, it is necessary to consider both shift- and capture-power reduction during the X-filling process to gain both satisfied test power result during at-speed scan testing.

Remersaro *et al.* [25] takes a fully specified test set as input and generates a new test set with reduced shift-power and capture-power. The authors first identify X-bits in the test set and then fill 50% of the X-bits using *Preferred fill* [23] while the remaining X-bits are filled next using *Adjacent fill* [20]. However, filling half of the X-bits for capture-power reduction and the other half for shift-power reduction is not a very good strategy. This is because, as discussed previously, the shift-power dissipations and the capture-power dissipation should be dealt with differently. The main objective in shift-power reduction is to decrease the average test power dissipation *as much as possible*; while the main duty in capture-power reduction is to keep it under a safe peak power limit.

III. IMPACT ANALYSIS FOR X-FILLING

In this work, we propose to study the impact of different X-bits on the CUT’s shift- and capture-power (namely *S-impact* and *C-impact*, respectively), and use them to guide the X-filling process.

A. Impact of X-Bits on Capture-Power

To make the impact of X-bits in test stimuli on capture transitions more clear, for an at-speed scan test with timing diagram shown in Fig. 1, we first expand the CUT’s combinational logic

¹X-filling does have implications on the CUT’s defect coverage, but it is beyond the scope of this paper.

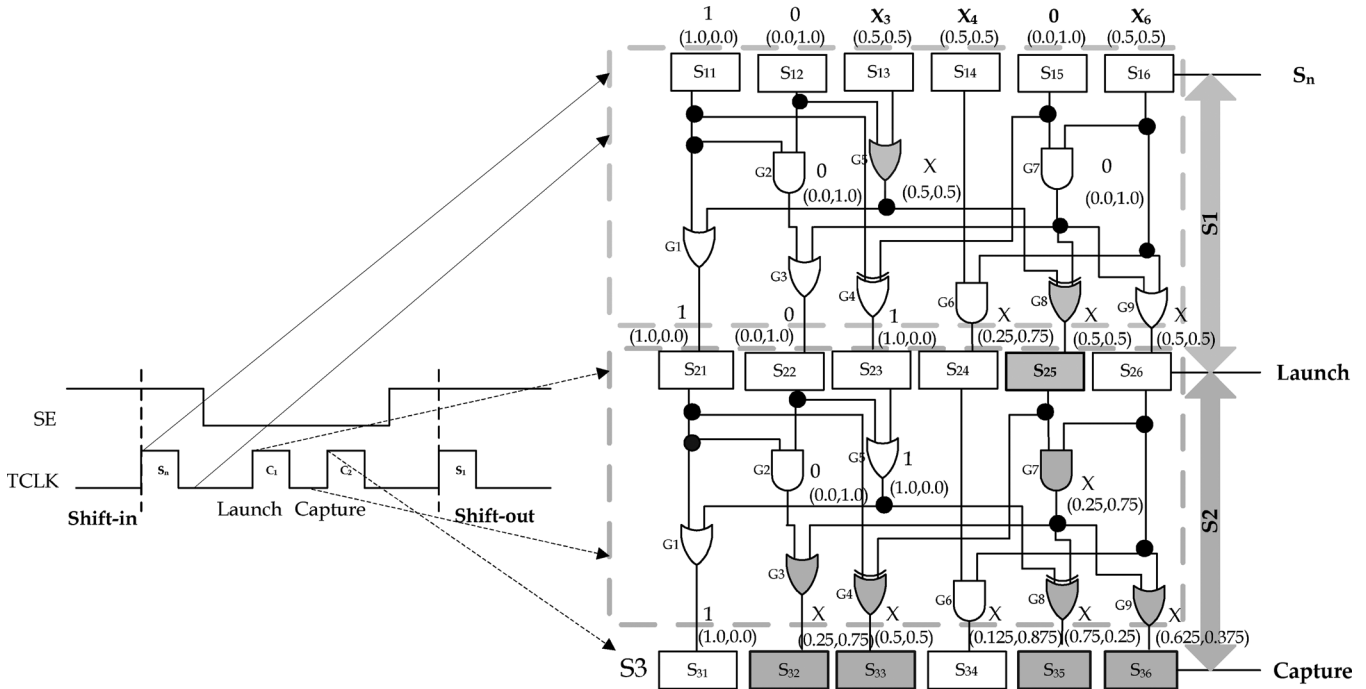


Fig. 3. Example circuit.

portion into two time-frames (as shown in Fig. 3). $S1$ and $S2$ denote the state of the circuit nodes before the launch and capture cycles, respectively; while $S3$ shows the final test responses after capture. As can be observed from this figure, the CUT is only applied at-speed in the capture cycle (i.e., launch of $S2$ and the unloading of $S3$ are not applied at-speed) in LoC scheme. Consequently, only the switching activities in this at-speed capture cycle (caused by the CUT state converts from $S1$ to $S2$) may cause test overkills in at-speed testing. We therefore consider transitions in this at-speed capture cycle only when modeling the impact of filling an X-bit in the test stimuli (e.g., X_i associated with SFF_i) on the CUT's capture-power dissipation.

In Fig. 3, (P_1, P_0) denotes the probability for a circuit node (i.e., SFFs and logic gates) to be logic "1" or logic "0". For each X-bit in $S1$, its (P_1, P_0) is initialized as $(0.5, 0.5)$. The probabilities for internal circuit nodes are calculated based on the CUT's logic structure [23]. For example, for a two-input AND gate, e.g., G_6 in Fig. 3, its two inputs are both X-bits with probability value of $(0.5, 0.5)$, according to the property of AND gate, probabilities of being logic "1" and "0" of the gate's output line (P_{1O}, P_{0O}) should be

$$P_{1O} = P_{1I_1} \times P_{1I_2} = 0.5 \times 0.5 = 0.25$$

$$P_{0O} = 1 - P_{1O} = 1 - 0.25 = 0.75$$

Similar computations can be applied to all types of gates in the CUT. We can thus obtain (P_1, P_0) value for each circuit node in $S1$ and $S2$, and the final test responses in $S3$.

The filling order for X-bits has a significant impact on the capture-power of the CUT, as shown in [21] and [24]. While it is possible to calculate the impact of different X-bits more accurately based on the above probability analysis, this value needs to be updated after filling each X-bit, which incurs extremely

high computational complexity. As a result, in our method, we propose to model the impact of an X-bit in a much simpler format using its fan-out information only. That is, generally speaking, an SFF with larger fan-out logic network cause more circuit transitions than an SFF with smaller one. Based on this observation, in this paper, we calculate $C - impact_i$ simply as the fan-out SFFs and logic gates that do not have determined logic values in the capture cycle. For example, in Fig. 3, among X_3 's fan-out logic network in the capture cycle, S_{25} , G_7 , G_3 , G_4 , G_8 , and G_9 are all X-bits, which means they are likely to make transitions, and hence we calculate its capture impact to be the total number of these node as: $C - impact_3 = 6$. Compared to the sophisticated method to calculate an X-bit's $C - impact$ (namely $X - score$ in their work) in [24], our method does not need to conduct the time-consuming set-simulation task and greatly reduce computational effort.

It is important to note that the above $C - impact$ is used to determine the filling order of X-bits for capture-power reduction. During the actual filling process for each X-bit, we still need to use the probability information (P_1, P_0) to decide whether to fill the X-bit with logic "0" or logic "1" (detailed in Section IV-A).

B. Impact of X-Bits on Shift-Power

During scan shift phase, while the test stimuli are shifted into the scan cells, previous test responses are shifted out concurrently. Therefore, we first identify those sequential elements in $S3$ that are possibly affected (turned into determined logic values from X-bits) when filling an X-bit in $S1$ (denoted as $S3_{affected}^i$), again, by tracing its fan-out logic network. For the example circuit in Fig. 3, when filling X_3 , $S3_{affected}^3 = \{S_{32}, S_{33}, S_{35}, S_{36}\}$ are affected with possible transitions during the shift-out process. We only consider X-bits affected in $S3$ because they are the final test responses

that need to be shifted-out for observation, while $S2$ is only an intermediate state.

Shift-power for a test vector depends not only on the number of transitions in it but also on their relative positions. Consider a particular X-bit i residing at position p_i on a scan chain sc with length $l_{sc,i}$, we propose to calculate the impact of filling X-bit X_i in $S1$ on shift-power as

$$S - impact_i = p_i + \sum_{j \in S3_{affected}^i} (l_{sc,j} - p_j) \quad (2)$$

where the first part and the second part of the above equation denote the impact of X_i on shift-in power and the one on shift-out power, respectively.

Similarly, the above S -*impact* is used to determine the filling order of X-bits for shift-power reduction. We also use the probability information (P_1, P_0) to decide whether to fill the X-bit with logic “0” or logic “1” during the actual filling process for each X-bit (detailed in Section IV-B).

IV. PROPOSED *iFill* X-FILLING TECHNIQUE

In this section, we detail our proposed *iFill* X-filling solution, including C -filling for capture-power reduction, S -filling for shift-power reduction, and the overall algorithm to achieve both shift- and capture-power reduction objectives. In addition, we show how to improve the runtime efficiency of the proposed solution.

A. C -Filling for Capture-Power Reduction

As discussed earlier, it is not necessary to reduce capture-power as much as possible. Instead, we only need to control it to be smaller than a safe threshold and we wish to use as few as possible X-bits to achieve this objective, so that the remaining X-bits can be filled for shift-power reduction. Therefore, we need to fill X-bits with higher C -*impact* earlier.

The transition probability for a logic node in the capture cycle when filling X_i is calculated as follows:

$$TP_i = P'_{1i} \times P_{0i} + P'_{0i} \times P_{1i} \quad (3)$$

where P'_{1i} (P'_{0i}) is its probability to be “1” (“0”) in $S1$, and P_{1i} (P_{0i}) is its probability to be “1” (“0”) in $S2$. For example, in Fig. 3, G_7 has “0” as its logic value in $S1$, so its P'_{17} is 0.0, and its P'_{07} is 1.0; in $S2$, since the logic value of G_7 becomes “X”, according to the calculation of signal probabilities of its fan-in ports, its P_{17} and P_{07} are 0.25 and 0.75, respectively. Therefore, $TP_7 = 0 \times 0.75 + 1 \times 0.25 = 0.25$.

The *Capture Transition Probability* (CTP) of the CUT caused by filling an X-bit X_i in test stimuli can be calculated as sum of transition probabilities of X-bits in its fan-out part

$$CTP_i = \sum_{j \in \text{fan-out}_{X_i}} TP_j. \quad (4)$$

Then we can decide the logic value to be filled for the target X-bit, which will cause less CTP in fan-out portion of this scan cell. The decision flow is shown in Fig. 4.

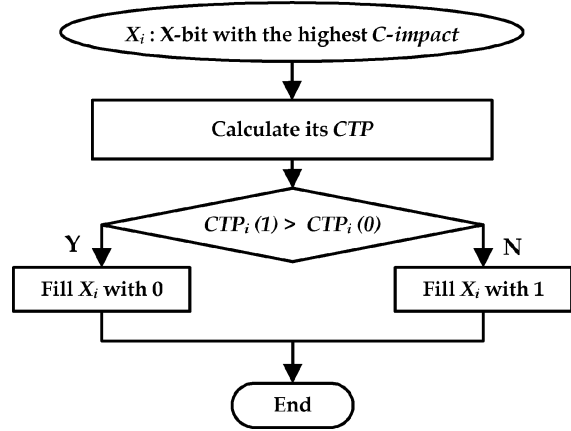


Fig. 4. X-filling flow for capture-power reduction.

For example in Fig. 3, first, we will choose the X-bit has the largest C -*impact*, X_3 to fill. Because S_{25} , G_7 , G_3 , G_4 , G_8 , and G_9 are logic nodes having undetermined values in fan-out portion of X_3 in the capture cycle, based on (4), we can calculate its CTP as

$$\begin{aligned} CTP_3(1) &= TP_{S_{25}}(1) + TP_{G_7}(1) + TP_{G_3}(1) + TP_{G_4}(1) \\ &\quad + TP_{G_8}(1) + TP_{G_9}(1) \\ &= 1 + 0.5 + 0.5 + 1 + 0.5 + 0.5 = 4 \\ CTP_3(0) &= TP_{S_{25}}(0) + TP_{G_7}(0) + TP_{G_3}(0) + TP_{G_4}(0) \\ &\quad + TP_{G_8}(0) + TP_{G_9}(0) \\ &= 0 + 0 + 0 + 0 + 1 + 0.5 = 1.5 \end{aligned}$$

$TP_i(1/0)$ represents the transition probability of the logic node i in fan-out logic structures of X_3 . From the above calculation, $CTP_3(0)$ is lower than $CTP_3(1)$, which means filling X_3 with logic “0” will cause fewer possible transitions in the CUT in the capture cycle when compared to filling it with logic “1”, therefore we should fill it accordingly. Note that, different from previous methods [21]–[23] that try to reduce the Hamming Distance between $S1$ and $S2$ through filling X-bit in the test stimuli with the same logic value as the logic value of this scan cell in $S2$ (“1” in this case), our method considers directly the switching activities in all the nodes of the circuit caused by transitions in scan cells. This strategy is proven to be more effective in our experiments.

B. S -Filling for Shift-Power Reduction

Prior work on X-filling for shift-power reduction (e.g., [20]) considers shift-in power only. This is unfortunate, because filling these unspecified bits has impact on both shift-in and shift-out power consumption.

To model the shift transition probability caused by logic differences between X_i and its adjacent scan cells in the test stimuli, we calculate the *shift-in transition probability* ($SITP$) caused by filling this X-bit as follows:

$$\begin{aligned} SITP_i &= (P_{1S_{i-1}} \times P_{0S_i} + P_{0S_{i-1}} \times P_{1S_i}) \times (i - 1) \\ &\quad + (P_{1S_i} \times P_{0S_{i+1}} + P_{0S_i} \times P_{1S_{i+1}}) \times i \quad (5) \end{aligned}$$

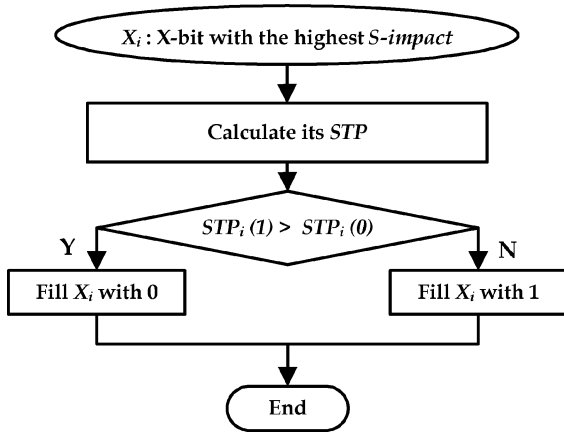


Fig. 5. X-filling flow for shift-power reduction.

where $P_{1s}(P_{0s})$ represents the probability of X_i to be 1 (0), i is the position this X-bit resides, which relates to transition number it may cause during the shift-in.

The calculation of the *shift-out transition possibility* (SOTP) caused by filling X_i is quite similar, which is

$$\begin{aligned}
 \text{SOTP}_i = & \sum_{j \in \text{fan-out}(X_i)} \\
 & \times [(P_{0R_{j-1}} \times P_{1R_j} + P_{1R_{j-1}} \times P_{0R_j}) \\
 & \times (l_{sc,j} - j + 1) + (P_{0R_j} \times P_{1R_{j+1}} + P_{1R_j} \times P_{0R_{j+1}}) \\
 & \times (l_{sc,j} - j)] \quad (6)
 \end{aligned}$$

where j ranges among all the X-bits affected by X_i , and $P_{1r}(P_{0r})$ represents the probability of the response X-bit X_j to be “1” (“0”). It should be noted that these X-bits in test responses can be in different scan chains.

Now we can determine the total *shift transition probability* (STP) when filling X_i with “1” and “0”, respectively. It can be simply calculated as the sum of its SITP and SOTP

$$\begin{aligned}
 \text{STP}_i(1) &= \text{SITP}_i(1) + \text{SOTP}_i(1) \\
 \text{STP}_i(0) &= \text{SITP}_i(0) + \text{SOTP}_i(0). \quad (7)
 \end{aligned}$$

To reduce shift-power, we should fill X_i with the logic value that will cause fewer transitions in both shift-in and shift-out phases. As shown in Fig. 5, if $\text{STP}_i(1) < \text{STP}_i(0)$, it means filling X_i with logic “1” is likely to generate less circuit transitions during the scan shift phase than that of filling it with logic “0”, so we should fill X_i with logic “1”. Otherwise, we should fill X_i with logic “0”.

For example, consider filling X_6 in Fig. 6 (the X-bit with the maximum *S-impact* in current filling step in this test vector), its SITP(1/0) after X-filling is

$$\begin{aligned}
 \text{SITP}_6(1) &= (0.5 \times 0 + 0.5 \times 1) \times 5 + (1 \times 0 + 0 \times 1) \times 6 \\
 &= 2.5 \\
 \text{SITP}_6(0) &= (0.5 \times 1 + 0.5 \times 0) \times 5 + (0 \times 0 + 1 \times 1) \times 6 \\
 &= 8.5.
 \end{aligned}$$

For the test responses part, X_6 affects the 6th and 7th X-bit in scan chain 1 and the 22nd X-bit in scan chain 2, suppose the

lengths of these two scan chains are both 50. The SOTP(1/0) values for filling X_6 with “1”/“0” are

$$\begin{aligned}
 \text{SOTP}_6(1) &= (1 \times 1 + 0 \times 0) \times (50 - 5) + (0 \times 0 + 1 \times 1) \\
 & \times (50 - 6) + (1 \times 1 + 0 \times 0) \times (50 - 7) \\
 & + (0.75 \times 1 + 0.25 \times 0) \times (50 - 21) \\
 & + (0 \times 0.5 + 1 \times 0.5) \times (50 - 22) = 167.75 \\
 \text{SOTP}_6(0) &= (1 \times 0 + 0 \times 1) \times (50 - 5) + (1 \times 0.25 + 0 \times 0.75) \\
 & \times (50 - 6) + (0.75 \times 1 + 0.25 \times 0) \\
 & \times (50 - 7) + (0.75 \times 0 + 0.25 \times 1) \times (50 - 21) \\
 & + (1 \times 0.5 + 0 \times 0.5) \times (50 - 22) = 64.5.
 \end{aligned}$$

Therefore, STP(1/0) after filling X_6 should be

$$\begin{aligned}
 \text{STP}_6(1) &= \text{SITP}_6(1) + \text{SOTP}_6(1) = 2.5 + 167.75 \\
 &= 170.25
 \end{aligned}$$

$$\text{STP}_6(0) = \text{SITP}_6(0) + \text{SOTP}_6(0) = 8.5 + 64.5 = 73.$$

From the calculation result we can see that, although filling X_6 with logic “1” (if using *Adjacent fill* [20]) will cause fewer transitions in the shift-in phase, it may cause significantly higher transitions in the shift-out phase. To better consider the impact of filling this X-bit on shift-power dissipation, we should fill X_6 with logic “0” instead.

C. Overall Flow

The objective of the proposed *iFill* X-filling technique for simultaneous shift- and capture-power reduction is to keep the capture transitions under threshold and reduce shift transitions as much as possible. To meet this target, we proposed the overall flow as outlined in Fig. 7. First, we try to conduct *S-filling* to use all the X-bits in test vectors for shift-power reduction and check whether the capture-power violates the constraint after the *S-filling* process. If it does, we need to re-load the initial test cube, and fill one X-bit with the highest *C-impact* value for capture-power reduction. After filling every X-bit for capture-power reduction, the test vector will be updated, and we will apply *S-filling* procedure one more time to fill the remaining X-bits and then the capture-power will be checked again to see whether this test vector still has capture-power violation. When there is no power violation, we have completed filling the vector; otherwise, *C-filling* procedure will be called again to reduce capture transitions. The above steps iterate themselves until there is no peak power violation or all X-bits have been utilized to reduce capture-power. If the capture transitions still violates the limit after all X-bits have been filled, this test pattern need to be discard. After X-filling for all the test patterns in give test set, new test patterns need to be generated for the faults the test pattern violating the capture power limit covered.

During the *S-filling* (*C-filling*) process, we always try to fill the X-bit with the highest *S-impact* (*C-impact*) value first. This incremental X-filling approach results in improved test power consumption when compared to the single-step filling approach such as *Preferred fill* [23] at the cost of higher computational time. While the proposed solution does not require to conduct backward justification as in [24], its computational complexity may still be too high for industrial circuits. In the following

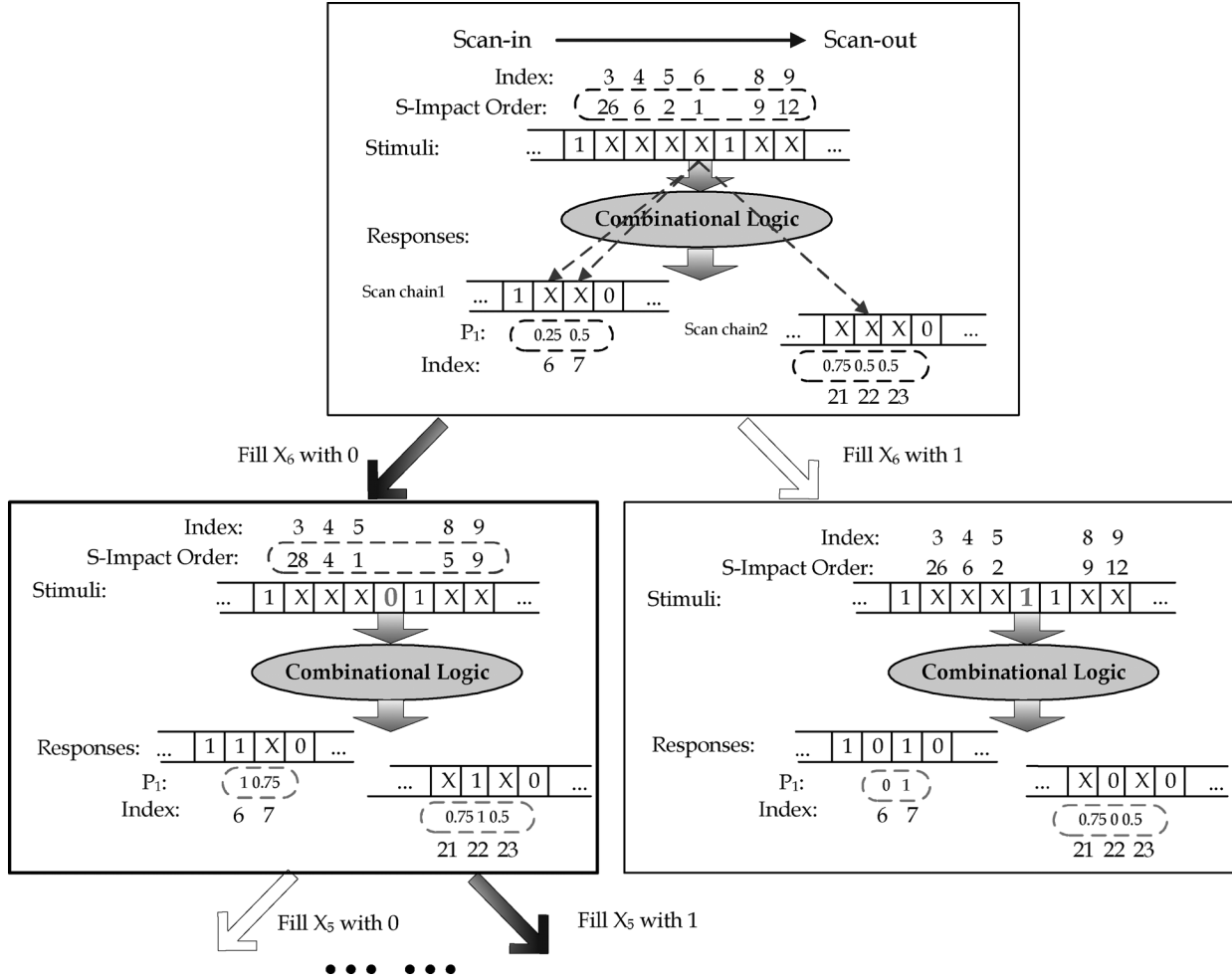


Fig. 6. Fill X-bits for shift-power reduction.

subsection, we show how to improve the runtime efficiency of the proposed method.

D. Runtime Enhancement

From the previous section we can see that the computational complexity of the proposed flow for one test vector should be $O(N^2)$ in the worst case, where N is the number of X-bits in this test vector, when there are many X-bits in the test vector, the computational time of the above solution can be quite high, so we introduce several techniques to tackle this problem.

1) *C-Filling for Multiple X-Bits at Once:* According to the flow in Fig. 7, every time after conducting *C-filling* for an X-bit, *S-filling* is called for the remaining X-bits again, which involves a significant amount of computation effort. Therefore, we would like to reduce the number of the above iterations by filling multiple X-bits in the *C-filling* procedure at once. Finding an appropriate number of X-bits to fill each time, however, is a tricky problem. If this number is too small, the number of iterations for *S-filling* cannot be efficiently reduced; if it is too large, on the other hand, some X-bits might be unnecessarily wasted for capture-power reduction in the *C-filling* procedure.

Since the main purpose of *C-filling* is to keep capture transitions in the CUT under a threshold (say, $\text{Thres}\%$ of all the

N_{gates} logic nodes), we can estimate the number of X-bits necessary for *C-filling* by checking the difference between the target transition count and the current one. Suppose the number of transitions after the i th pass of *S-filling* is N_{trans}^i . We have this difference N_{excs} as

$$N_{\text{excs}} = N_{\text{trans}}^i - N_{\text{gates}} \times \text{Thres}\%. \quad (8)$$

Generally speaking, there is linear correlation between the transition count of the logic nodes in the combinational portion and that of the scan cells. Therefore, in the following *C-filling* pass, we determine the number of X-bits to be filled as

$$N_{\text{fill}}^i = N_{\text{sff}} \times \left(\frac{N_{\text{excs}}}{N_{\text{gates}}} \right). \quad (9)$$

Based on the above, we maintain a list of those unfilled X-bits and sort them based on their *C-impact* value in non-increasing order. For each *C-filling* process, we choose the top N_{fill}^i X-bits and fill them accordingly. The *C-impact* values are updated before every *C-filling* process. In our experiments, we find that N_{fill}^i can converge quickly after a few passes of *S-filling* and *C-filling*, which verifies the efficiency of the proposed solution.

2) *Combining S-Filling and Adjacent Fill:* The advantage of *S-filling* compared to *Adjacent fill* [20] is the consideration for

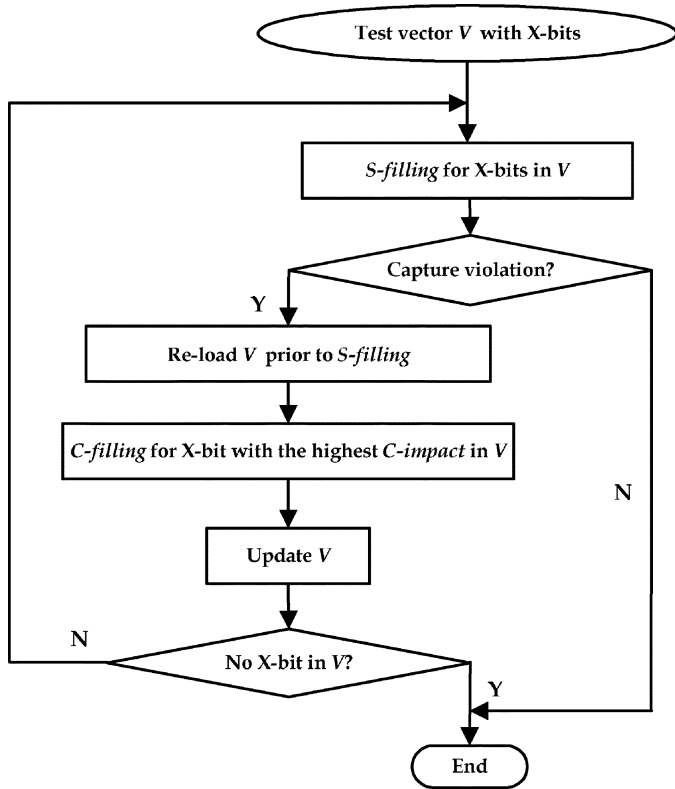


Fig. 7. Overall flow.

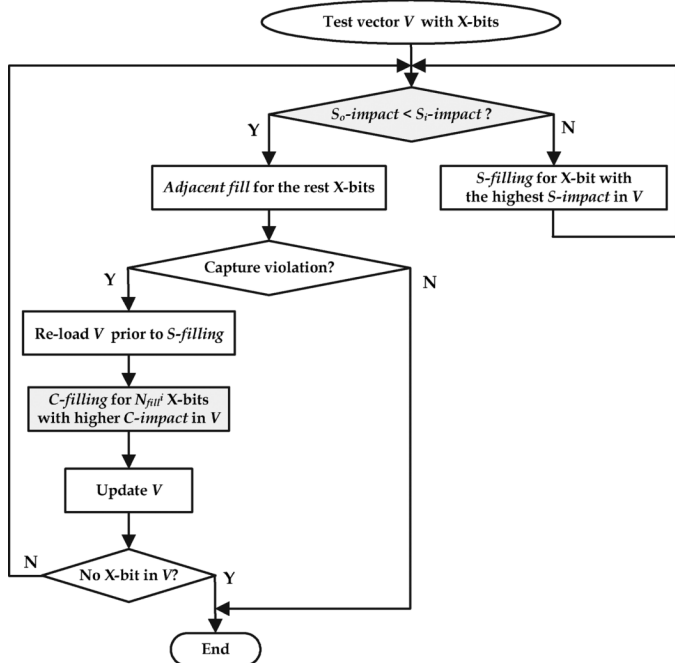


Fig. 8. Runtime enhancement of the initial overall flow.

shift-out power dissipation, but this is associated with a high computational cost.

Therefore, if the impact of X-bits on shift-out power is less significant than that on shift-in power, we can simply use *Adjacent fill* for shift-power reduction, thus cutting down the computational time. Based on this observation, we divide the *S-im-*

TABLE I
STATISTICAL INFORMATION FOR EXPERIMENTAL CIRCUITS

circuit	#sff	#gate	#pattern	X%	#ori vios
s5378	179	3042	125	72.83%	2
s9234	211	5883	137	64.99%	3
s15850	534	10546	115	74.17%	0
s13207	638	8872	115	84.44%	1
s38417	1636	24167	258	85.59%	1
s38584	1426	21175	168	77.23%	3
b20	490	8875	350	73.24%	0
b21	490	9259	369	74.02%	0
b22	735	14282	373	73.87%	0
b17	1415	22645	435	89.17%	0
b18	2988	63045	571	90.17%	0
b19	5736	126415	824	92.90%	0

fact for every X-bit X_i into two parts: $S_i - \text{impact} = p_i$ and $S_o - \text{impact} = \sum (l_{sc,j} - p_i)$, represented for the impact of this X-bit on shift-in power and shift-out power, respectively. The proposed *S-filling* (see Section IV-B) will be conducted only for those X-bits having higher $S_o - \text{impact}$ value than their $S_i - \text{impact}$ value. The other X-bits will be filled by *Adjacent fill*.

With the above methods, the flow of the enhanced algorithm is shown in Fig. 8 and its computational time can be controlled at an acceptable level for most circuits.

V. EXPERIMENTAL RESULTS

To evaluate the effectiveness and efficiency of the proposed *iFill* solution, we conduct experiments on several ISCAS'89 and ITC'99 benchmark circuits on a 2 GHz PC with 1 GB RAM. The transition delay fault test sets for these circuits are generated by a commercial application test pattern generation (ATPG) tool. The experimental setup for these circuits are listed in Table I, including the number of scan cells ($\#sff$), the number of logic nodes ($\#gate$), the number of test patterns ($\#pattern$), and the percentage of X-bits ($X\%$).

The peak power constraint for the CUT's transitions for ISCAS'89 and ITC'99 circuits are set as 20% and 10% of the total logic nodes, respectively. That is, fewer than 20% (10%) of the logic nodes in the CUT are allowed to make transitions during capture to avoid IR-drop (ground bounce) related problems.² There are already some capture violations in several initial test cubes before X-filling, listed under "*#ori vios*", we get this data by counting the number of logic nodes that already have transitions during the at-speed capture cycle before the X-bits in the test cube are filled, and these violations cannot be avoided by any X-filling methods.

A. Experiments on Shift-Power Reduction

Table II compares the shift-power reduction for *Original*, *Adjacent fill* [20], and the proposed *S-filling* method when all X-bits are utilized for shift-power reduction, in terms of WTM[38] including both shift-in and shift-out phases. The shift-power reduction percentage of *S-filling* compared to *Adjacent fill* are listed under "*Red*".

²The peak power constraints used in our experiments are decided by the observation of switching densities of test patterns for these two sets of benchmark circuits. In practice, this peak power constraint should be given by the designer of the CUT.

TABLE II
COMPARISON OF SHIFT-POWER REDUCTION RATIO

circuit	<i>Original</i>	<i>Adjacent fill</i>	<i>S-filling</i>	<i>Red</i>
s5378	13476	5851	5081	13.2%
s9234	22268	14749	13052	11.5%
s15850	136262	74854	54790	26.8%
s13207	204830	102154	76469	25.1%
s38417	1275445	337658	322086	4.6%
s38584	1027407	610803	518060	15.2%
b20	112572	30033	17729	41.0%
b21	112107	29047	16861	42.0%
b22	249328	62668	37195	40.6%
b17	969649	102612	73124	28.7%
b18	4354455	561176	345991	38.3%
b19	16097456	1783038	1283776	28.0%
<i>Average</i>	2047938	309554	230351	26.3%

We can see that although [20] can already reduce the shift-power significantly when compared to the “*Original*” test patterns where X-bits are filled randomly by the test pattern generator, *S-filling* can still achieve significant shift-power reduction over *Adjacent fill*. This is expected because [20] fills X-bits for shift-in power reduction only, which may result in excessive shift-out power dissipation. By considering the shift transition probability in both test stimuli and test responses with some computational complexity overhead, the shift-power of the circuit with *S-filling* can be reduced dramatically, especially for larger circuits. Consequently, our technique is able to reduce the accumulated thermal effects during test, and hence enables higher shift frequency and/or more test parallelism for test cost reduction.

B. Experiments on Capture-Power Reduction

Table III compares the capture-power reduction of the proposed *C-filling* procedure with that of *Preferred fill* [23] (we do not compare with [24] because it only has results for stuck-at tests), when all X-bits are utilized to control capture transitions. “*Cap. sffs*”, “*Ave. Cap.*”, “*Peak Cap.*”, and “*#vio.s*” represent the number of capture transitions on SFFs, the average and peak number of capture transitions on all logic nodes, and the number of test vectors that have capture transition violations in the capture cycle, respectively. and the reduction of these transitions of our *C-filling* procedure compared to *Preferred fill* are listed under “*Red. sffs*”, “*Red. Ave.*”, and “*Red. Peak*”.

From this table, we can observe that, while the *Preferred fill* can significantly cut down the capture transitions compared to “*Original*” test patterns, the proposed *C-filling* can further achieve significant capture-power reduction compared to *Preferred fill*, as shown under “*Red. sffs*”, “*Red. Ave.*”, and “*Red. Peak*”. Among these three capture transitions reduction values, we can see that, the proposed *C-filling* procedure usually can achieve higher average capture transitions in all the circuit nodes, which has more correlation to the capture power value than capture transitions in the scan cells. The maximum capture transition is also better controlled by *C-filling*, which facilitates to reduce the number of test patterns that violate the capture power threshold. These results prove the effectiveness of using *C – impact* to guide the X-filling process for capture-power reduction, by which we can use *C-filling* to control capture

transitions of the test patterns that may violate the capture power threshold with fewer X-bits.

C. Experiments on Simultaneous Shift- and Capture-Power Reduction

Table IV compares the proposed holistic X-filling technique for simultaneous shift- and capture-power reduction against the method proposed in [25], which proposed to use half of the X-bits for capture-power reduction with *Preferred fill* [23] while the remaining X-bits are filled using *Adjacent fill* [20] for shift-power reduction. The average and peak shift- and capture- transition count and the corresponding reduction ratio of the proposed “*iFill*” against the method in [25] are listed under “*iFill*”, “[25]”, and “*Red.*” below “*Ave. Shift*”, “*Peak Shift*”, “*Ave. Cap.*”, and “*Peak Cap.*”, respectively. The number of test patterns that violates the capture transition count limit (“*#vio.s*”) of “*iFill*” and “[25]” are also listed below “*#vio.s*”.

From Table IV, the proposed *iFill* technique generally achieves significant peak capture-power reduction when compared to [25], and thus *iFill* causes much fewer capture-power violations. This is expected as: 1) [25] used *Preferred fill* for capture power reduction, while the proposed *iFill* technique outperforms *Preferred fill* on capture violation control and 2) *iFill* can use more than half of the X-bits for capture-power reduction, if necessary. We can also observe that the average capture transition of *iFill* is similar to [25]. This is because, as discussed early, the objective of capture-power reduction is to keep it under the safe limit. As long as this objective is achieved, there is no need to further reduce it.

We can also see that the average shift-power reduction of the proposed *iFill* is much higher than that of [25] because: 1) the proposed *iFill* uses the *S-filling* procedure to reduce shift-power, which takes both shift-in and shift-out transitions into account during X-filling, while [25] utilizes *Adjacent fill* that can only reduce shift-in power and 2) the proposed *iFill* can control the capture power under the threshold with fewer X-bits by the *C-filling* procedure, while [25] has only half of the X-bits for shift-power reduction.

The peak shift-power reduction ratio of these two techniques are similar, and *iFill* may lead to higher peak shift-power than [25] in some circuits. This is mainly because there are some patterns in the test set that only contain very few X-bits, which limits the efficiency of X-filling techniques on shift-power reduction. Moreover, for test patterns with very high capture-power, *iFill* is likely to use most of the X-bits to reduce capture transitions (i.e., less X-bits are used for shift-power reduction), while [25] still fills half of the X-bits using *Adjacent fill*.

D. Experiments on Computational Time of the Proposed X-Filling Technique

Despite of its effectiveness on test power reduction, the runtime of the proposed *iFill* may be very high, which limits its scalability for large industrial design. To solve this problem, we also perform the enhanced algorithm proposed in Section IV-D to verify its runtime and shift- and capture-power reduction efficiency.

TABLE III
COMPARISON OF CAPTURE-POWER REDUCTION RATIO

circuits	Original				Preferred fill				C-filling				Red. sffs	Red. Ave.	Red. Peak
	Cap. sffs	Ave. Cap.	Peak Cap.	#vio.s	Cap. sffs	Ave. Cap.	Peak Cap.	#vio.s	Cap. sffs	Ave. Cap.	Peak Cap.	#vio.s			
s5378	89	794	985	125	53	429	782	7	27	232	734	4	49.1%	45.9%	6.1%
s9234	77	1357	1860	113	35	765	1482	12	30	563	1405	3	14.3%	26.4%	5.2%
s15850	192	1762	2138	1	66	665	1857	0	53	452	1752	0	19.7%	61.5%	5.7%
s13207	237	1881	2323	110	167	1137	1911	3	61	438	1890	2	63.5%	32.0%	1.1%
s38417	446	5537	7053	239	250	2186	6065	5	78	921	5963	3	68.8%	57.9%	1.7%
s38584	709	5164	5506	168	207	1609	5313	9	169	1162	5107	8	18.4%	27.8%	3.9%
b20	65	1332	2548	223	48	706	2251	21	23	319	1457	0	52.1%	54.8%	35.3%
b21	66	1374	2878	235	45	614	1840	16	21	308	1651	0	53.3%	49.8%	10.3%
b22	98	2057	3914	213	74	1016	2936	5	28	463	2130	0	62.2%	54.4%	27.5%
b17	77	2358	3765	34	23	1365	2648	0	18	576	1778	0	21.7%	57.8%	32.9%
b18	156	3473	6092	132	179	1492	5982	0	75	622	2940	0	58.1%	58.3%	50.9%
b19	280	6638	11485	96	308	2083	8366	0	52	530	6290	0	61.0%	47.1%	24.8%
Average	208	2811	4212	140.8	121	1172	3453	6.5	53	549	2758	1.7	47.0%	50.1%	17.1%

TABLE IV
COMPARISON OF SHIFT- AND CAPTURE-POWER AGAINST [25]

circuits	Ave. Shift			Peak Shift			Ave. Cap.			Peak Cap.			#vio.s	
	[25]	iFill	Red.	[25]	iFill	Red.	[25]	iFill	Red.	[25]	iFill	Red.	[25]	iFill
s5378	7035	6469	8.0%	14282	15807	-10.7%	673	573	14.9%	1028	734	28.6%	78	4
s9234	15380	14187	7.8%	24772	24522	1.0%	1101	1123	-2.0%	1601	1405	12.2%	50	3
s15850	68466	57753	15.6%	153416	161287	-5.1%	1257	1097	12.7%	1881	1982	-5.4%	0	0
s13207	112644	93847	16.7%	234947	216330	7.9%	1440	1644	-14.2%	2084	1890	9.3%	6	2
s38417	446968	362935	18.8%	1446189	1509120	-4.4%	3170	4066	-28.3%	6283	5963	5.1%	6	3
s38584	567544	549621	3.2%	1153205	1167678	-1.3%	3902	3857	1.2%	5331	5039	5.5%	58	8
b20	32739	19959	39.0%	116850	116850	0.0%	1028	790	23.2%	2532	1457	42.5%	197	0
b21	31346	18583	40.7%	111196	106720	4.0%	1032	784	24.0%	2082	1651	20.7%	178	0
b22	69803	38908	44.3%	249827	241746	3.2%	1535	1080	29.6%	3417	2130	37.7%	186	0
b17	132013	73124	44.6%	1045650	953738	8.8%	1563	1221	21.9%	2727	1778	34.8%	5	0
b18	745235	345991	53.6%	4707262	4671691	0.8%	1992	1729	13.2%	5344	2940	45.0%	0	0
b19	2203572	1283776	41.7%	17236904	16916330	1.9%	3052	3185	-4.4%	8468	6290	25.7%	0	0
Average	369395	238763	27.8%	2207875	2175152	0.5%	1812	1762	7.7%	3565	2802	21.2%	63.7	1.7

TABLE V
IMPROVED RUNTIME OF THE PROPOSED X-FILLING TECHNIQUE AND ITS CAPTURE- AND SHIFT-POWER

circuit	Ave.Shift	Ave.Cap.	#vios	T_{tg}	T_{X-fill}
s5378	7032	539	4	0.23	9.578
s9234	14728	991	3	0.39	36.984
s15850	57753	1396	0	0.81	24.109
s13207	88624	1097	2	2.04	115.141
s38417	365375	3890	3	2.26	373.344
s38584	484339	3390	8	2.34	189.276
b20	30097	1127	0	137.47	20.515
b21	28951	1046	0	95.92	21.968
b22	63208	1377	0	131.25	15.672
b17	102616	1993	0	416.32	46.594
b18	561176	2558	0	796.54	133.062
b19	1783038	4310	0	1741.10	400.110
Average	298911	1976	1.7	277.223	115.529

Table V shows the results of the enhanced X-filling procedure on average shift-power (“Ave.Shift”) in term of WTM, average capture-power (“Ave.Cap.”) in term of transition count of circuit nodes, runtime of the proposed X-filling procedure (“ T_{X-fill} ”) and runtime of test pattern generation (“ T_{tg} ”).

From this table and the results from Table IV we can see that, although we may have higher average capture transitions than the original flow, it has no impact on reducing capture-power violations. At the same time, we cannot achieve the same amount

of shift-power reduction because fewer X-bits are used for this objective, especially for the shift-out phase. Nevertheless, by reducing the number of iterations for *S-filling* and *C-filling* and the number of calculations for *S-impact* and *C-impact*, the running time of the proposed X-filling technique in Fig. 8 is significantly reduced when compared to the original procedure. In particular, it is much shorter than the runtime needed for the test pattern generation process (T_{tg}).

VI. CONCLUSION AND FUTURE WORK

This paper presents an effective and efficient impact-oriented X-filling method, namely “*iFill*”, which is able to keep the CUT’s capture-power within its peak power rating while reduce the CUT’s shift-power as much as possible. Another contribution of the proposed technique is that it is able to cut down power consumptions in both shift-in and shift-out processes. Experimental results on ISCAS’89 and ITC’99 benchmark circuits demonstrate its effectiveness.

In this work, all X-bits in given test cubes are used for test power reduction. In practice, however, these X-bits may be used for other purposes (e.g., test compression). We plan to take the above into consideration in our future work to develop holistic X-filling solutions that are able to fulfill different needs.

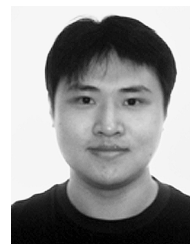
REFERENCES

- [1] P. Girard, X. Wen, and N. A. Touba, "Low-power testing," in *System-on-Chip Test Architectures: Nanometer Design for Testability*, L.-T. Wang, C.E. Stroud, and N.A. Touba, Eds. San Francisco, CA: Morgan Kaufmann, 2007, ch. 7.
- [2] P. Girard, "Survey of low-power testing of VLSI circuits," *IEEE Des. Test Comput.*, vol. 19, no. 3, pp. 80–90, May-Jun. 2002.
- [3] J. Saxena, K. M. Butler, V. B. Jayaram, and S. Kundu, "A case study of IR-drop in structured at-speed testing," in *Proc. IEEE Int. Test Conf. (ITC)*, Oct. 2003, pp. 1098–1104.
- [4] J. Wang, D. M. H. Walker, A. Majhi, B. Kruseman, G. Gronthoud, L. E. Villagra, P. v. d. Wiel, and S. Eichenberger, "Power supply noise in delay testing," in *Proc. IEEE Int. Test Conf. (ITC)*, 2006, p. 17.3.
- [5] S. Bhunia, H. Mahmoodi, D. Ghosh, S. Mukhopadhyay, and K. Roy, "Low-power scan design using first-level supply gating," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 13, no. 3, pp. 384–395, Mar. 2005.
- [6] L. Whetsel, "Adapting scan architectures for low power operation," in *Proc. IEEE Int. Test Conf. (ITC)*, Oct. 2000, pp. 863–872.
- [7] K. J. Lee, T. C. Huang, and J. J. Chen, "Peak-power reduction for multiple-scan circuits during test application," in *Proc. IEEE Asian Test Symp. (ATS)*, Nov. 2000, pp. 453–458.
- [8] P. M. Rosinger, B. M. Al-Hashimi, and N. Nicolici, "Scan architecture with mutually exclusive scan segment activation for shift- and capture-power reduction," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 23, no. 7, pp. 1142–1153, Oct. 2004.
- [9] K. J. Lee, S. J. Hsu, and C. M. Ho, "Test power reduction with multiple capture orders," in *Proc. IEEE Asian Test Symp. (ATS)*, 2004, pp. 26–31.
- [10] Y. Bonhomme, P. Girard, C. Landrault, and S. Pravossoudovitch, "Power driven chaining of flip-flops in scan architectures," in *Proc. IEEE Int. Test Conf. (ITC)*, Oct. 2002, pp. 796–803.
- [11] Y. Bonhomme, P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "Efficient scan chain design for power minimization during scan testing under routing constraint," in *Proc. IEEE Int. Test Conf. (ITC)*, 2003, pp. 488–493.
- [12] J. Li, Y. Hu, and X. Li, "A scan chain adjustment technology for test power reduction," in *Proc. IEEE Asian Test Symp. (ATS)*, 2006, pp. 11–16.
- [13] R. Sankaralingam and N. A. Touba, "Inserting test points to control peak power during scan testing," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Syst. (DFT)*, 2002, pp. 138–146.
- [14] S. Sharifi, J. Jaffari, M. Hosseinabady, A. Afzali-Kusha, and Z. Navabi, "Simultaneous reduction of dynamic and static power in scan structures," in *Proc. Des., Autom., Test Eur. (DATE)*, 2005, pp. 846–851.
- [15] S. Gerstendorfer and H. J. Wunderlich, "Minimized power consumption for scan-based BIST," in *Proc. IEEE Int. Test Conf. (ITC)*, 1999, pp. 77–84.
- [16] P. Girard, L. Guiller, C. Landrault, and S. Pravossoudovitch, "Circuit partitioning for low power BIST design with minimized peak power consumption," in *Proc. IEEE Asian Test Symp. (ATS)*, 1999, pp. 89–94.
- [17] N. Z. Basturkmen, S. M. Reddy, and I. Pomeranz, "A low power pseudo-random BIST technique," in *Proc. Int. Conf. Comput. Des. (ICCD)*, 2002, pp. 468–473.
- [18] R. Sankaralingam, B. Pouya, and N. A. Touba, "Reducing power dissipation during test using scan chain disable," in *Proc. IEEE VLSI Test Symp. (VTS)*, 2001, pp. 319–324.
- [19] Q. Xu, D. Hu, and D. Xiang, "Pattern-directed circuit virtual partitioning for test power reduction," in *Proc. IEEE Int. Test Conf. (ITC)*, 2007, p. 25.2.
- [20] K. M. Butler, J. Saxena, A. Jain, T. Fryars, J. Lewis, and G. Hetherington, "Minimizing power consumption in scan testing: Pattern generation and DFT techniques," in *Proc. IEEE Int. Test Conf. (ITC)*, Oct. 2004, pp. 355–364.
- [21] X. Wen, Y. Yamashita, S. Kajihara, L.-T. Wang, K. K. Saluja, and K. Kinoshita, "On low-capture-power test generation for scan testing," in *Proc. IEEE VLSI Test Symp. (VTS)*, May 2005, pp. 265–270.
- [22] X. Wen, K. Miyase, S. Kajihara, T. Suzuki, Y. Yamato, P. Girard, Y. Ohsumi, and L. T. Wang, "A novel scheme to reduce power supply noise for high-quality at-speed scan testing," in *Proc. IEEE Int. Test Conf. (ITC)*, 2007, p. 25.1.
- [23] S. Remersaro, X. Lin, Z. Zhang, S. Reddy, I. Pomeranz, and J. Rajski, "Preferred fill: A scalable method to reduce capture power for scan based designs," in *Proc. IEEE Int. Test Conf. (ITC)*, Santa Clara, CA, 2006, p. 32.2.
- [24] X. Wen, K. Miyase, T. Suzuki, Y. Yamato, S. Kajihara, L. T. Wang, and K. K. Saluja, "A highly-guided X-filling method for effective low-capture-power scan test generation," in *Proc. Int. Conf. Comput. Des. (ICCD)*, 2006, pp. 251–258.
- [25] S. Remersaro, X. Lin, S. M. Reddy, I. Pomeranz, and J. Rajski, "Low shift and capture power scan tests," in *Proc. Int. Conf. VLSI Des.*, 2007, pp. 793–798.
- [26] V. Iyengar and K. Chakrabarty, "Precedence-based, preemptive, and power-constrained test scheduling for system-on-a-chip," in *Proc. IEEE VLSI Test Symp. (VTS)*, Marina del Rey, CA, May 2001, pp. 368–374.
- [27] V. Dabholkar, S. Chakravarty, I. Pomeranz, and S. M. Reddy, "Techniques for minimizing power dissipation in scan and combinational circuits during test application," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 17, no. 12, pp. 1325–1333, Dec. 1998.
- [28] S. Wang and S. K. Gupta, "ATPG for heat dissipation minimization during test application," *IEEE Trans. Comput.*, vol. 47, no. 2, pp. 256–262, Feb. 1998.
- [29] T. C. Huang and K. J. Lee, "An input control technique for power reduction in scan circuits during test application," in *Proc. IEEE Asian Test Symp. (ATS)*, 1999, pp. 315–320.
- [30] R. Sankaralingam and N. A. Touba, "Controlling peak power during scan testing," in *Proc. IEEE VLSI Test Symp. (VTS)*, 2002, pp. 153–159.
- [31] W. Li, S. M. Reddy, and I. Pomeranz, "On test generation for transition faults with minimized peak power dissipation," in *Proc. ACM/IEEE Des. Autom. Conf. (DAC)*, 2004, pp. 504–509.
- [32] A. Chandra and K. Chakrabarty, "Test data compression for system-on-a-chip using Golomb codes," in *Proc. IEEE VLSI Test Symp. (VTS)*, 2000, pp. 113–120.
- [33] J.-L. Yang and Q. Xu, "State-sensitive X-filling scheme for scan capture power reduction," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 27, no. 7, pp. 1338–1343, Jul. 2008.
- [34] J. Li, X. Liu, Y. Zhang, Y. Hu, X. Li, and Q. Xu, "On capture power-aware test data compression for scan-based testing," in *Proc. Int. Conf. Comput.-Aided Des. (ICCAD)*, 2008, pp. 67–72.
- [35] J. Li, Q. Xu, Y. Hu, and X. Li, "iFill: An impact-oriented X-filling method for shift- and capture-power reduction in at-speed scan-based testing," in *Proc. Des., Autom., Test Eur. (DATE)*, 2008, pp. 1184–1189.
- [36] R. M. Chou, K. K. Saluja, and V. D. Agrawal, "Scheduling tests for VLSI systems under power constraints," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 5, no. 2, pp. 175–184, Jun. 1997.
- [37] K. Miyase and S. Kajihara, "XID: Don't care identification of test patterns for combinational circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 23, no. 2, pp. 321–326, Feb. 2004.
- [38] R. Sankaralingam, R. R. Oruganti, and N. A. Touba, "Static compaction techniques to control scan vector power dissipation," in *Proc. IEEE VLSI Test Symp. (VTS)*, 2000, pp. 35–40.



Jia Li (S'09) received her Ph.D. degree in Computer System and Architecture from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2009. She is currently working as a post-doc with Tsinghua University, Beijing, China.

Her research interest includes the field of low power testing, and testing of network-on-chip.



Qiang Xu (M'06) received the Ph.D. degree in electrical and computer engineering from McMaster University, Hamilton, ON, Canada, in 2005.

Since 2005, he has been an Assistant Professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. He leads the CUHK Reliable Computing Laboratory (*CURE Lab.*). His research interests range from test and debug of system-on-a-chip integrated circuits to fault tolerance and reliable computing. He has published more than 40 technical papers in these areas.

Dr. Xu was a recipient of the Best Paper Award in 2004 IEEE/ACM Design, Automation and Test in Europe Conference (DATE). He is a member of the ACM SIGDA and the IEEE Computer Society. He has served as a technical program committee member for a number of conferences on VLSI design and testing.



Yu Hu (M'06) received the B.S., M.S., and Ph.D. degrees, all in electrical engineering, from the University of Electronic Science and Technology, Chengdu, China, in 1997, 1999, and 2003, respectively.

She is currently an Associate Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. Her research interests generally include architectural-level and circuit-level design-for-reliability, especially fault diagnosis and fault tolerance techniques.

Dr. Hu is a member of IEICE and CCF.



Xiaowei Li (SM'04) received his B.Eng. and M.Eng. degrees in Computer Science from Hefei University of Technology, China, in 1985 and 1988, respectively, and his Ph.D. degree in Computer Science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), in 1991.

He was born in 1964 in China. From 1991 to 2000, he was an Assistant Professor and an Associate Professor (since 1993) in the Department of Computer Science, Peking University, China. During 1997 and 1998, he was a Visiting Research Fellow in the Department of Electrical and Electronic Engineering, the University of Hong Kong. During 1999 and 2000, he was a Visiting Professor in the Graduate School of Information Science, Nara Institute of Science and Technology, Japan. He joined the ICT, CAS as a Professor in 2000. He is now the deputy director of the Key Lab. of Computer System and Architecture, CAS. He is a senior member of IEEE.

Dr. Li's research interests include VLSI Testing, design for testability, design verification, software testing, dependable computing, wireless sensor networks. He has participated in more than 20 research projects in these areas. He has co-published over 150 papers in academic journals and international conference, hold 21 patents and 29 software copyrights. Dr. Li served as Chair of CCF (China Computer Federation) Technical Committee on Fault Tolerant Computing since 2008. He co-initiated the first China Test Conference (CTC) in 2000, and served as the Program Committee Co-Chair for CTC00 and CTC02, General Co-Chair for CTC06. He served as IEEE Asian Pacific Regional TTTC (Test Technology Technical Council) Vice Chair since 2004. He served as the Steering Committee Vice-chair of IEEE Asian Test Symposium (ATS) since 2007, and served as the Program Committee Co-Chair for ATS2003 and General Co-Chair for ATS2007. He also served as the Steering Committee Chair of IEEE Workshop on RTL and High Level Testing (WRTLTL), and served as the General Chair for WRTLTL2003 and WRTLTL2007. In addition, he serves on the Technical Program Committee of several IEEE and ACM conferences, including VTS, DATE, ASP-DAC, PRDC, etc. He also serves as member of editorial board of JCST, JOLPE, JETTA, etc.