iFill: An Impact-Oriented X-Filling Method for Shift- and Capture-Power Reduction in At-Speed Scan-Based Testing

Jia LI1, 2, Qiang XU3, Yu HU1, and Xiaowei LI1*

1Key Laboratory of Computer System and Architecture, ICT, CAS, Beijing, China
2Graduate University of Chinese Academy of Sciences, Beijing, China
3Dept. of Computer Science & Engineering, The Chinese University of Hong Kong
{gracelee, huyu, lxw}@ict.ac.cn, qxu@cse.cuhk.edu.hk

Abstract

In scan-based tests, power consumptions in both shift and capture phases may be significantly higher than that in normal mode, which threatens circuits’ reliability during manufacturing test. In this paper, by analyzing the impact of X-bits on circuit switching activities, we present an X-filling technique that can decrease both shift- and capture-power to guarantees the reliability of scan tests, called iFill. Moreover, different from prior work on X-filling for shift-power reduction which can only reduce shift-in power, iFill is able to decrease power consumptions during both shift-in and shift-out. Experimental results on ISCAS’89 benchmark circuits show the effectiveness of the proposed technique.

1. Introduction

The power dissipation of integrated circuits (ICs) in scan-based testing can be significantly higher than that during normal operation [1]. This will threaten the reliability of the circuits under test (CUT), because: (i) the elevated average power dissipation 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; (ii) 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 at-speed testing, thus invalidating the testing process and leading to yield loss [1-3].

It is likely that a CUT’s power rating is violated in shift and/or capture mode. These two types of test power violations, however, should be dealt with differently. In shift mode, test vectors are shifted into/out of scan chains bit by bit, which not only dominate the test time of the CUT, but also determine the CUT’s accumulated effect of test power dissipation. Therefore, the shift power reduction should be decreased as much as possible, so that we 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. In capture mode, since the duration is very short, it has limited effect on the CUT’s accumulated test power consumption. On the contrary, because test vectors are generated to detect as many faults as possible and hence often triggers more transitions in capture cycle, the main duty in capture power reduction is to keep it under a safe peak threshold, especially in at-speed testing. As long as this requirement is fulfilled, there is no need to further reduce it.

Various X-filling techniques have been proposed in literature to reduce shift- and/or capture-power in scan-based testing. However, they either target only one type of power consumption (shift-power reduction [15] or capture-power reduction [16-18]) or do not consider the difference of the two types of power consumptions [19]. In this paper, we investigate the impact of X-bits and propose a novel X-filling technique to reduce both shift- and capture-power during at-speed scan tests, namely iFill. In the proposed approach, we try to fill as few as possible X-bits to keep the capture-power under the peak power limit of the CUT, and use all the remaining X-bits to reduce shift-power as much as possible to cut down the CUT’s average power.

In addition, prior work on shift-power reduction (e.g., Adjacent fill [15]) considers the power consumption during shift-in process only, which, unfortunately, may lead to excessive power for the shift-out process. The proposed iFill technique is able to cut down power consumptions in both shift-in and shift-out processes, thus leading to significant shift-power reduction. To the best of our knowledge, this is the first work that is able to achieve the above goal without adding design-for-testability hardware. Experiments results conducted on ISCAS’89 benchmark circuits show that the proposed technique is superior to prior techniques in the literature, in terms of both shift and capture power reduction.

The remainder of this paper is organized as follows. Section 2 presents the background and Section 3 details the proposed iFill technique. Experimental results are presented in Section 4. Finally, Section 5 concludes this paper.

978-3-9810801-3-1/DATETIME © 2008 EDAA
Therefore, X-filling techniques have received lots of attention recently from both academia and industry [15-19].

Adjacent fill [15] is a simple yet effective technique targeting shift-power reduction. However, it can reduce the shift-in power only. Wen et al. [16] first addressed the low capture-power solution with X-filling. They considered the transitions at the output of SFFs during X-filling, which, however, does not necessarily have a good correlation with the total capture power of the whole circuit. Later, in [18], they took the above 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 [16, 18] is that their computational time is quite high. This is because: (i). they are incremental filling approaches, that is, they fill the X-bits in the test cube one by one; (ii). forward implications and backward justification are extensively used in their methodologies. In fact, the complexity of the set-simulation techniques proposed in [18] is quite high and it is difficult, if not impossible, to be applicable for two-pattern tests in industrial designs. In [17], 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. Remersaro et al. addressed this problem in [19], but filling half of the X-bits for capture-power reduction and the other half for shift-power reduction is not a very good strategy, without considering the average and peak power ratings of the CUT.

Based on the above observations, this paper presents an efficient impact-oriented X-filling method, namely iFill, which can keep the CUT’s capture-power within peak power constraint while reduce its shift-power as much as possible.

3. iFill: Impact-Oriented X-Filling

3.1 Impact of X-bits on shift- and capture-power

Test cube generally contains multiple X-bits, and as many X-bits in the test response are likely to become determined values after filling one single X-bit in the test stimulus [16, 18], their filling order significantly affects the CUT’s test power dissipation. We therefore try to model the impact of an X-bit on a CUT’s shift- and capture-power (namely S-impact and C-impact), and use them to guide the X-filling.

Generally, an SFF with larger fan-out logic network
involves more circuit transitions. Based on this observation, we model the impact of an X-bit on circuit transitions with its fan-out information only. Compared to the sophisticated method to calculate an X-bit’s X-score in [18], our method does not need to conduct the time-consuming set-simulation and we can target two-pattern at-speed scan tests.

For an at-speed scan tests with timing diagram as Fig. 1, we expand the CUT’s combinational portion into two time frames as Fig. 2. $S1$ and $S2$ denote the states of the scan cells before and after the launch operation, while $S3$ shows the final state after capture. $(P_1, P_0)$ denotes the probabilities for the circuit nodes to be ‘1’ or ‘0’. For each X bit in $S1$, $(P_1, P_0)$ is initialized as (0.5, 0.5). The probabilities of the other circuit nodes are calculated based on the logic structure [17].

To model the impact of a stimulus X-bits $(X_i, associated with SFF_i)$ on the CUT’s capture-power dissipation, we calculate its $C-impact_i$, as the total number of fan-out FFs and logic gates that have undetermined logic values in the capture cycle, e.g., in Fig. 2, among $X_i$’s fan-out, $S_{25}, G_7, G_4, G_5, G_6$ and $G_8$ probably have transitions, and hence its capture impact will be $C-impact_i=6$. We do not consider the transitions before the launch cycle because the CUT is typically not applied at-speed in this cycle.

During the scan shift phase, the test stimuli are shifted out in scan cells with previous test responses shifted out concurrently. To model the impact of an X-bit in the test stimuli on shift-power dissipation, we need to define a completely different cost factor because shift-power mainly concerns transitions between adjacent scan cells instead of switching activities in the entire CUT. Therefore, we first identify the scan cells in $S3$ that are possibly affected by filling an X-bit in $S1$ (denoted as $S_{3\text{affected}}$), by tracing its fan-out logic network. For example in Fig. 2, when filling $X_3$, $S_{3\text{affected}} = \{S_{25}, S_{33}, S_{15}, S_{16}\}$ are affected during shift-out.

Shift-power for a test vector depends on both the number of transitions in it and their relative positions. Consider an X-bit $i$ residing at position $x_i$ on a scan chain $sC$ with length $l_{sC,i}$, the impact of filling $X_i$ in $S1$ on shift-power is:

$$S-impact_i = p_i + \sum_{j \in S_{3\text{affected}}} (l_{sC,j} - p_j)$$

where the first and the second part of the equation denotes the impact of $X_i$ on shift-in and shift-out power, respectively.

3.2 iFill design flow

The calculation of $C-impact$ and $S-impact$ are used to guide our X-filling process in Fig. 3. As emphasized earlier, we only need to keep the capture power within the peak power limit while reduce the shift power as much as possible. Therefore, in the proposed flow, we first conduct X-filling for shift-power reduction (denoted as $S$-filling) and check whether the capture power is within the CUT’s peak power limit. If not, we need to fill X-bits for capture power reduction (denoted as $C$-filling). Once we have filled one X-bit to reduce capture power, the $S$-filling procedure is applied again to fill the remaining X-bits and the capture power violation will be checked again. If there is still violation, $C$-filling procedure is called again. These steps iterate themselves till there is no peak power violation or all X-bits have been utilized to reduce capture power.
During the $S$-filling ($C$-filling) process, we always try to fill the X-bits with highest $S$-impact ($C$-impact) values first. This incremental X-filling approach results in improved test power reduction when compared to the single-step filling approach such as Preferred fill [17] at the cost of higher computational time. However, as the proposed $i$Fill method does not need to run the complex set-simulation procedure as in [18], the computational complexity is well-controlled.

### 3.3 S-filling for Shift-power Reduction

Prior X-filling methods for shift-power reduction (e.g., [15]) consider the shift-in power only, but filling X bits may have impact on both shift-in and shift-out power. This section shows how we consider both shift-in and shift-out power reduction with X-filling.

As previously discussed, we fill the X-bit with the highest $S$-impact first. To model the shift transition probability in the test stimuli, we calculate the Shift-In Transition Probability ($SITP$) caused by filling one X-bit as follows:

$$ SITP_{i} = (P_{s_{i}} \times P_{n_{i}} + P_{n_{i}} \times P_{s_{i}}) \times (i - 1) + (P_{s_{i}} \times P_{n_{i}} + P_{n_{i}} \times P_{s_{i}}) \times i $$

where $P_{s_{i}}$ ($P_{n_{i}}$) represents the probability of $X_i$ to be 1 (0).

The calculation of the Shift-Out Transition Possibility ($SOTP$) caused by filling $X_i$ is quite similar, as:

$$ SOTP_{i} = \sum_{j = \text{max}_j(X_i)} \left[(P_{s_{j}} \times P_{n_{j}} + P_{n_{j}} \times P_{s_{j}}) \times (l_{s_{j}} - j + 1) + (P_{s_{j}} \times P_{n_{j}} + P_{n_{j}} \times P_{s_{j}}) \times (l_{n_{j}} - j)] \right) $$

where $j$ ranges all the X-bits affected by $X_i$, notice that these X-bits can be in different scan chains. Now the Shift Transition Probability ($STP$) is decided as:

$$ STP_{i}(1) = SITP_{i}(1) + SOTP_{i}(1) $$

$$ STP_{i}(0) = SITP_{i}(0) + SOTP_{i}(0) $$

As shown in Fig. 5, if $STP_{i}(1) < STP_{i}(0)$, filling $X_i$ with ‘1’ is likely to generate fewer shift transitions on scan chains.

Consider filling $X_6$ (the X bit with the highest $S$-impact value) in Fig. 4, $X_6$ 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 scan chains are both 50, its $SITP$ and $SOTP$ will be:

$$ SITP_{6}(0) = (0.5 \times 1 \times 0.5 \times 1) \times 5 + (1 \times 0 \times 1) \times 6 = 2.5 $$

$$ SOTP_{6}(0) = (0.5 \times 1 \times 0.5 \times 1) \times 5 + (0 \times 0 \times 1) \times 1 \times 6 = 8.5 $$

$$ SITP_{6}(1) = (1 \times 1 \times 1 \times 0) \times (50 - 5) + (0 \times 0 \times 1) \times (50 - 6) $$

$$ + (1 \times 1 \times 1 \times 0.25 \times 0.25) \times (50 - 21) + (0 \times 0 \times 1 \times 0.5 \times 0.5) \times (50 - 22) = 167.75 $$

$$ SOTP_{6}(1) = (1 \times 1 \times 1 \times 0) \times (50 - 5) + (0 \times 1 \times 0.25 \times 0.25) \times (50 - 6) $$

$$ + (1 \times 1 \times 1 \times 0.25 \times 0.25) \times (50 - 21) + (0 \times 1 \times 0 \times 0.5 \times 0.5) \times (50 - 22) = 64.5 $$

So $STP$ of filling $X_6$ should be:

$$ STP_{6}(1) = SITP_{6}(1) + SOTP_{6}(1) = 2.5 + 167.75 = 170.25 $$

$$ STP_{6}(0) = SITP_{6}(0) + SOTP_{6}(0) = 8.5 + 64.5 = 73 $$

Therefore, we should fill $X_6$ with ‘0’ to achieve lower shift-power dissipation. After that, the next X-bit with the highest $S$-impact value will be filled, $X_5$ in this example. The iteration will continue till there is no X-bit in the test vector.

![Figure 4. Fill X-bits for shift-power reduction](image-url)
transition probability and we measure transition probability of logic nodes in
reduction, we target the X-bits with higher

test stimuli is calculated as:

\[
\text{S-filling} = \sum_{i} (P_i \times P_0 + P_i \times P_1)
\]

where for all logic nodes affected by \( X_i \), \( P_i \) is its probability to be ‘1’ (‘0’) before the launch operation, and \( P_0 \) is its probability to be ‘1’ (‘0’) after the launch. For the experimental circuit in Fig. 2, \( X_j \) has the largest C-impact and hence should be filled first. Because \( S_2, G_3, G_4, G_5 \) and \( G_6 \) are logic nodes having undetermined values in fan-out portion of \( X_j \) in the capture cycle, based on Eq. (5), we can calculate its CTP as follows:

\[
\text{CTP}_i(i) = \left( \sum_{j \in S_2, G_3, G_4, G_5, G_6} (P_{i,j} \times P_{0,j} + P_{i,j} \times P_{1,j}) + (P_{i,j} \times P_{0,j} + P_{i,j} \times P_{0,j}) + (P_{i,j} \times P_{0,j} + P_{i,j} \times P_{0,j}) \right)_{i,j = 0,1,2,3} = 0.0
\]

\[
\text{CTP}_i(0) = \left( \sum_{j \in S_2, G_3, G_4, G_5, G_6} (P_{i,j} \times P_{1,j} + P_{i,j} \times P_{1,j}) + (P_{i,j} \times P_{0,j} + P_{i,j} \times P_{0,j}) + (P_{i,j} \times P_{1,j} + P_{i,j} \times P_{0,j}) \right)_{i,j = 0,1,2,3} = 1.5
\]

Therefore we should filled \( X_j \) with logic ‘0’.

4. Experimental Results

To evaluate the performance of the proposed iFill technique, we conduct experiments on several ISCAS’89 circuits. MINTEST [20] is utilized as the test cube. Table I compares the shift-power reduction between using Adjacent fill [15] and the proposed S-filling procedure, in terms of WTM [4]. We can see S-filling leads to significant shift-power reduction compared to Adjacent fill. This is expected because Adjacent fill fills X-bits for shift-in power reduction only, which may result in excessive shift-out power dissipation. However, the computational time of S-filling is longer than Adjacent fill because we need to calculate the transition probability in test responses, but it is still acceptable: for the largest ISCAS’89 benchmark circuit, s38584, S-filling takes less than 10 minutes to finish with a 2GHz PC with 1G RAM.

Table II shows capture-power reduction of C-filling when compared to Preferred fill [17] (we do not compare with [18] because it only has results for stuck-at tests). “Cap. in SFFs” and “Cap. in all nodes” represent the number of capture transitions on SFFs/all logic nodes, and the number of test vectors that have capture transition violations, respectively. From this table, we can observe that C-filling generally achieves better capture-power reduction than Preferred fill, which proves the effectiveness of the proposed C-impact in determining the X-filling order. The computational time of C-filling is similar as S-filling.

Finally, Table III compares iFill with Adjacent fill and Preferred fill for the same test cubes, in terms of both shift-power reduction and capture-power violations. The peak constraint for the CUT’s transitions is set as 10% of the total logic nodes, that is, fewer than 10% logic nodes in the CUT are allowed to make transitions during capture. The number of scan cells and test patterns for each circuit are shown under “# of SFFs” and “# of Patterns”. The X-bits percentages in each test cube are under “% of X-bits”. There are already capture violations in specified bits in several test vectors before X-filling, the number of such vectors are listed under “# of Original Vios”. “Ave. Shift”, “Ave. Cap.”, and “# of Vios” represent the average shift-power in terms of WTM, the average capture transition count on SFFs, and the number of capture transition violations in the CUT, respectively. Our goal is to reduce the average shift-power dissipation and the number of capture-power violations.

From Table III, we can see that, iFill can generally achieve the minimum average shift-power dissipation and the minimum number of capture violations among these three methods. It is important to note that in circuit s5378 and s15850, while Preferred fill can achieve less average capture transitions, it actually causes more peak power violations with significantly high average shift-power. Clearly it is therefore beneficial to use the proposed iFill technique for both shift- and capture-power reduction.
Table II. Comparison of capture-power reduction

<table>
<thead>
<tr>
<th>Circuits</th>
<th>Random fill</th>
<th>Preferred fill</th>
<th>C-filling</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td># of SFFs</td>
<td>Cap. in all nodes</td>
<td># of Vios</td>
</tr>
<tr>
<td>s1196</td>
<td>8</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>s1238</td>
<td>9</td>
<td>30</td>
<td>2</td>
</tr>
<tr>
<td>s5378</td>
<td>89</td>
<td>905</td>
<td>111</td>
</tr>
<tr>
<td>s9234</td>
<td>77</td>
<td>1455</td>
<td>152</td>
</tr>
<tr>
<td>s13207</td>
<td>233</td>
<td>2074</td>
<td>228</td>
</tr>
<tr>
<td>s15850</td>
<td>162</td>
<td>1847</td>
<td>3</td>
</tr>
<tr>
<td>s38417</td>
<td>413</td>
<td>5665</td>
<td>1</td>
</tr>
<tr>
<td>s38584</td>
<td>388</td>
<td>3484</td>
<td>0</td>
</tr>
</tbody>
</table>

5. Conclusion

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

References