Effect of IR-Drop on Path Delay Testing Using Statistical Analysis

Chunsheng Liu[†] Yang Wu[†] Yu Huang[‡]

[†]Computer and Electronic Engineering
University of Nebraska-Lincoln
Omaha, NE 68182, USA
cliu@engr.unl.edu

[‡]Mentor Graphics
300, Nickerson Rd
Marlborough , MA 01752, USA
yu_huang@mentor.com

Abstract

IR-drop has become a major source of delay defects in deep sub-micron VLSI designs. In this work, we analyze the effect of IR-drop in path-delay test and how to obtain more accurate delay information of critical paths. For possible regions with IR-drop, we perform timing analysis on these nodes such that a certain amount of voltage drop can be associated with extra delays on victim nodes. Power analysis is conducted to determine the occurrence probability of a certain voltage drop. These probability values are used to weigh the extra delays caused by IR-drop of all victim nodes, which are then accumulated along each path. Experimental results show that such a process can effectively take the small delays caused by IR-drop into consideration and can have a significant impact on the identification and analysis of critical paths.

1 Introduction

In deep submicron designs, power dissipation has become an increasingly critical issue. Excessive power density can cause "hot spots" that require a large amount of current to be delivered to. Such non-uniform pattern of power consumption (or current distribution) can cause a non-uniform voltage drop, or IR-drop, on the power grid, along with many other signal integrity issues [10, 13]. As designs approach to 90nm and deeper, the problem caused by IR-drop has become one of the top culprits of defects [7].

Since IR-drop leads to lower instantaneous supply power voltage, an immediate impact is the increased delay on the affected gates. It's been reported that 10-15% drop on power voltage can account for 20-30% increase on gate delay [3]. This represents a critical issue in path delay testing. First, affected gates apply extra delay on paths that run through these gates (or so-called victims), which can directly change the path delays. As a result, previously non-critical paths may become critical. Second, if path delay ATPG does not consider IR-drop, the test pattern may not cover these newly produced critical paths, causing possible failure escape.

Although extensive work has been conducted on path delay test [11, 12], little has been done with the consideration of IR-drop effect. Prior work on IR drop analysis either does not address path delay test, or has been computationally intensive [4, 6, 9]. Timing analysis tool should be able to handle the IR-drop delay and transfer the information to ATPG. It is also preferable that the flow of IR-drop calculation can be either pre-layout, where estimated voltage drop can be used, or post-layout, where accurate physical design information is used. The former is not accurate but can be used to predict possible critical paths to guide physical design.

In this paper, we use statistical method to analyze the delays caused by IR-drop. We first conduct timing analysis on possible victim gates so that a certain amount of voltage drop on these gates can be associated with a specific extra delay. Power analysis tool is then used to estimate the occurrence probabilities of 1 or 0 on related nodes, which are then used to calculate an extra delay value weighed by occurrence probabilities. Extra IR-drop delays are then accumulated over all affected victim gates and along each critical path to constitute an "expected extra path delay" caused by IR-drop. This flow is suitable for both pre- and post-layout analysis. Experimental results show that the application of IR-drop delays can substantially affect the critical paths.

Note that in this work we do not attempt to target IR-drop faults by modifying ATPG, as previous work has done [2, 4, 6, 9]. Rather, the goal is to show that our method can be applied on top of the timing analysis results earlier in the design/test flow, such that not only ATPG, but other processes can also benefit from this method.

The rest of the paper is organized as follows. Section 2 lists some related prior work on IR-drop analysis. Section 3 presents the model and analysis techniques. In Section 4, we show experimental results on some ITC'99 benchmark circuits. Section 5 concludes this paper.

2 Related Work

Prior work has addressed the problems associated with IR-drop in several aspects. In [8], IR-drop effect is mitigated by layout enhancement through redundant metal fills. In [1, 14], monitoring/sensing circuits are embedded into

the design to identify and localize the areas where IR-drop occurs. The impact of IR-drop on delay faults is modeled and analyzed in [13].

To handle possible IR-drop failures in manufacture testing, a majority of work has resorted to test pattern generation, manipulation or validation [2, 4, 5, 6, 9, 10]. In [2], a method to measure the average power of at-speed test patterns was proposed. Meanwhile, a pattern generation method was also proposed to generate supply noise tolerant delay test patterns. In [4, 6], input patterns are generated such that the voltage drop is maximized, through which path delay is maximized, to capture possible delay failure caused by IR-drop. Genetic algorithm is used for patterns selection, hence these techniques are computational intensive and correspond to a worst case analysis. Another problem is that in testing, switching activities can be much higher than normal functioning. As a result, some IR-drop may occur in testing but not in normal functional mode, causing over-test and yield loss. This is addressed in [5] through a pattern validation process to remove the misleading test patterns. In [10], manufacture failures caused by IR-drop are analyzed.

More recently, the problem is targeted in ATPG [9]. It adapts an ATPG process for stuck-at faults to detecting voltage drop faults and uses an abstract model to simplify the process. The method sets a threshold voltage drop. If the accumulative voltage drop on a victim gate is higher than the threshold, a slow-to-rise or slow-to-fall failure is said to occur. A process is then used to generate patterns from existing patterns for stuck-at faults to excite the targeted gates and detect possible voltage drop faults. It reports a high coverage on such faults as well as the stuck-at faults.

3 IR-drop Calculation Based on Statistical Analysis

In this section, we present our method of calculating voltage drop based on a statistical analysis. We first illustrate the overall flow in 3.1. We then explain the simplified model based on which we establish the analysis in 3.2. Finally we show the procedure of statistical calculation of the extra delay caused by IR-drop in 3.3.

3.1 Overall Flow

The overall flow of the method is shown in Figure 1. We assume the design is given in gate-level netlist with other necessary information such as technology library, design rules, etc. In practice extensive simulation is required to obtain timing and current profile for standard cells, so that with physical information, the delay characteristics caused by excessive current drawn from power grid can be accurately calculated. In this paper, since we don't have physical design information and the timing/current characteristics, we use hypothetical data. However, we should note that if these data are available, they can be easily applied to our approach without any modification of the flow.

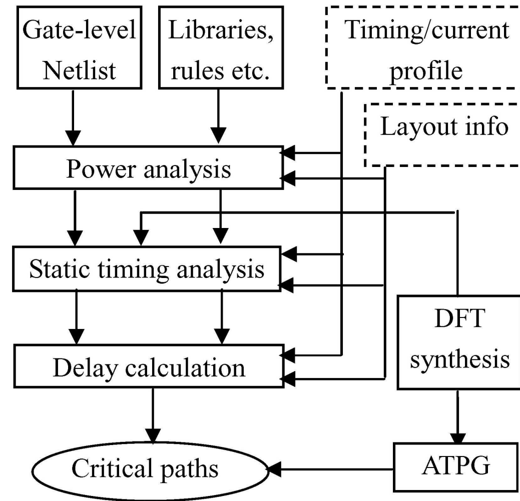

Figure 1. Overall flow.

We first apply power analysis to the design, which has two purposes. First, power analysis can estimate the switching activity of the design to identify potential "hot spots" that could have IR-drop problem. Second, power analyzer can estimate the 1/0 probability of each node, which can be used to calculate IR-drop delay in the subsequent process. With physical information and functional patterns, these two estimation can be made more accurate.

In static timing analysis, we identify critical paths in the design. These paths will be evaluated in the next step by calculating extra delays caused by IR-drop. Both can be more accurate with layout information and timing/current profile. After this analysis, the critical paths will be reevaluated and IR-drop can cause new paths to become critical. Finally, path list is targeted by path-delay ATPG.

Note that in this paper most of our work has focused on the delay calculation part. This will be depicted in the subsequent sections.

3.2 A Simplified Model

The aggressor/victim model we used in this paper is similar to the one used in [9, 13]. We assume that a gate is directly connected to its nearest M2-M3 V_{dd} and V_{ss} vias. More than one gates can be connected to a via. Each V_{dd} via (or V_{ss} via) I has a list F_I of gates in physical proximity whose switching activities contribute to the voltage drop or surge at I. As in [9], we assume such voltage drop and surge is symmetric and consider only V_{dd} drop. Simultaneous $0 \longrightarrow 1$ switching of some gates in F_I may cause a local voltage drop on I, which can then lead to an amount of extra delay on one or more gates directly connected to via I.

The gates that contribute to the voltage drop at I are defined as *aggressors*, and the gates that are affected by the drop to have an extra delay when they switch are defined

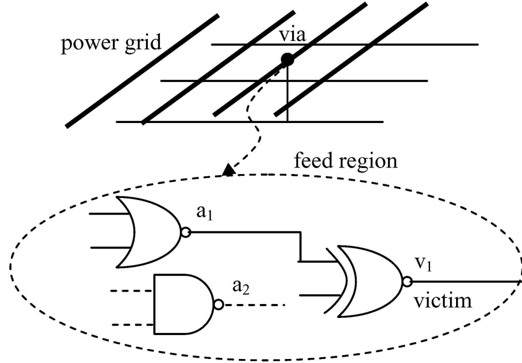

Figure 2. Power grid and aggressor/victim gates.

as *victims*. Note that a victim may be an aggressor itself, but not necessarily. F_I is defined as *feed region*. The feed regions that may incur high voltage drop can be identified in power analysis with physical information. It is reported in [13] that the number of such "targeted" regions is usually small and the number of gates in F_I is on average only 6. Hence the simplified model is practical in terms of computational complexity. In this work, we randomly select such regions for analysis, as done in [9], due to the lack of physical design information. In reality, they may be identified using various power analysis tools.

Figure 2 shows power grid, feed region, two aggressor gates a_1, a_2 and victim gate v_1. Note that victim itself may be an aggressor. Also note that gates in a feed region are in physical proximity but not necessarily logic proximity. The extra delay caused by IR-drop on the victim gate can cause any path that run through it to have increased path delay.

3.3 Expected Path Delay Method

Since voltage drop causes extra delay on victim gates, the associated path delay will increase. However, the exact delay is hard to determine due to several reasons. E.g., physical information and the timing/current profile can not be completely accurate. Moreover, various input patterns can cause different switching activities at aggressors, which may lead to different voltage drops, and hence different extra path delays. Functional patterns can better capture the behavior of such extra delays, but they are extremely expensive for manufacture testing purpose and therefore impractical for path-delay test, especially when the number of critical paths is large.

This problem is not well addressed in prior work, which has mostly focused on the worst case delay. In this scenario, the goal is to find a set of input patterns such that the corresponding transitions launched at aggressors can maximize the voltage drop, and hence the worst delay. However, in functional mode the transition is generally much lower and IR-drop is not as high as in the testing mode. Based on the worst case delay, chips will be over-tested, which causes yield loss. Functional patterns can give more accurate result, but are extremely expensive.

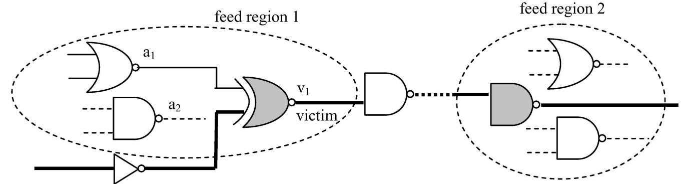

Figure 3. Two feed regions on a critical paths.

Here we propose the use of a statistical method for calculating the extra delay caused by voltage drop. We assume that using physical information, we have already obtained a set of feed regions, each containing a set of aggressors/victims. We also assume that static timing analysis has yielded a list of targeted critical paths. An example is given in Figure 3, which shows two feed regions on a critical path (in solid bold face) and two victims (shaded gates).

First, we use power analysis tool to obtain the probability of each aggressor node being logic 1/0. Without functional patterns and/or design information, tools will usually assume random values on inputs and then traverse the topology to calculate the probability for each node. If such information is available, the estimated probability can more accurately reflect the functioning behavior.

For a feed region F_i, let us assume aggressor set is $A_i = \{a_{i_1}, a_{i_2}, ..., a_{i_j}\}$, and the corresponding probabilities of being 0/1 are $\{p^0_{a_{i_1}}, p^0_{a_{i_2}}, ..., p^0_{a_{i_j}}\}$ and $\{p^1_{a_{i_1}}, p^1_{a_{i_2}}, ..., p^1_{a_{i_j}}\}$ respectively.

With these probabilities, we can calculate the probability that a specific switching activity occur on the aggressors. Consider the feed region 1 in Figure 3 as an example. Since we only consider V_{dd} drop, if both aggressors have $0 \longrightarrow 1$ switch, the voltage drop is maximized and so is the extra delay. E.g., if $p^0_{a_{1_1}} = 0.4$, $p^1_{a_{1_1}} = 0.6$ and $p^0_{a_{1_2}} = 0.2$, $p^1_{a_{1_2}} = 0.8$, then the probability P to maximize the delay caused by V_{dd} drop in feed region 1 is $P_1 = (0.4{*}0.6){*}(0.2{*}0.8) = 0.0384$. (It can be easily seen that if we consider V_{ss} surge, the same probability can be obtained.) Similarly, if a_1 switches but a_2 doesn't, the probability is $P_2 = (0.4*0.6)*(0.2*0.2+0.8*0.8) = 0.1632$. Other probabilities can be correspondingly calculated. Note that we have assumed switching activities on two nodes are independent, which is a simplified assumption when functional behavior is not available. In practice, they could be logically dependent, which needs more complex calculation.

It can be seen that each probability corresponds to a specific V_{dd} drop. Since in this work we don't have physical design information and timing/current profile, we use hypothetical data. In practice, this can be obtained from the timing/current profile. The V_{dd} drop can then be simulated in static timing tool to determine an extra delay D. Some sample data are shown in Table 1 for illustration. Each row in the table represents a combination of values on the aggressors, i.e. a pattern. We use $\uparrow$ and - to denote $0 \longrightarrow$

Table 1. Sample data of switching probability and extra delay.

Activity on a_1, a_2	V_{dd} drop (V)	Prob P_k	Extra delay D_k (ns)	Weighted delay WD_k (ns)
↑↑	0.6	0.038	1.2	0.0456
↑-	0.3	0.163	0.8	0.1304
-↑	0.2	0.083	0.6	0.0498
- -	0	0.354	0	0

1 switching and no switching, respectively. Note that now each switching probability P_k is associated directly to a specific extra delay D_k. Also note that the sum of the probabilities is not 1, because we considered only V_{dd} drop without V_{ss} surge.

The last column of the Table represents the key to the method. It is given by $WD_k = P_k * D_k$, which is the delay weighted by occurrence probability. If we accumulate all the weighted delay values, it becomes the form of an expected value $ED = \sum_{k=1}^{m} P_k * D_k$, which we defined as an *expected extra delay*. Here m is the total number of different combinations of input switching, i.e. the number of rows in Table 1. The ED parameter evaluates the extra delay caused by IR-drop in a statistical manner. If there are several feed regions on a path, as shown in Figure 3, they will be accumulated to form the path's expected extra delay caused by IR-drop, $PED = \sum_{r=1}^{n} ED_r = \sum_{r=1}^{n} \sum_{k=1}^{m} P_k * D_k$.

The method has advantages over both using functional patterns and using structure ATPG to target the IR-drop defect. On one hand, as we have discussed early, using functional patterns can be either computationally impractical or extremely expensive. It may not guarantee to find the pattern(s) that leads to the worst extra delay. On the other hand, using structural path-delay ATPG can not guarantee that either, causing possible failure escape, or under-test. Meanwhile, the pattern(s) found by ATPG could cause over-test and hence yield loss. The proposed statistical method does not require intensive computation for functional patterns. It can capture all possible types of switching, mitigating the under-test problem. Meanwhile, if a pattern excites a false path, it is likely that the probability of the corresponding switching pattern is very low, or even 0, hence the delay is not exaggerated. Therefore, the PED is a useful measure. It can be used to reevaluate the critical paths reported by static timing analysis under the consideration of IR-drop.

Finally, we show a partial result by applying the proposed flow and the statistical method to one of the ITC'99 benchmark circuits, namely b14. We pick one of the many paths under simulation and show the results in Table 2. There are two feed regions on the path, each has four aggressors and one victim, which is one of the aggressors. Column 1 gives the number of aggressors that switch simultaneously under certain inputs. Column 2 gives the corresponding probability. Column 3 gives the extra delay incurred on the victim, and Column 4 shows the corresponding weighted delay WD. Column 5 gives the sum of the weighted delays, which is the expected extra delay ED.

Table 2. Results of two feed regions on a path of b14.

# of switch	Probability	Extra delay delay (ns)	Weighted delay delay (ns)	Expected delay (ns)
feed region 1				
4	0.0004	1.95	0.0007	
3	0.008	1.46	0.011	
2	0.06	0.98	0.059	0.174
1	0.21	0.49	0.10	
0	0.27	0	0	
feed region 2				
4	0.0004	3.19	0.0012	
3	0.008	2.39	0.019	
2	0.06	1.6	0.097	0.282
1	0.21	0.79	0.165	
0	0.27	0	0	

Table 3. Results of a path delay in b14.

Original path delay (ns)	New path delay (ns)	Increase by (%)	Worst path delay (ns)	Increase by (%)
7.4	7.86	6.2	12.54	69.5

In Table 3, we show the results of IR-drop on the path delay. It can be seen that the expected path delay gives rise to a 6.2% increase to the original path delay. If we only consider the worst case (maximum voltage drop) without being weighted by the probability, the path delay can increase by 69.5%. However, from the probabilities shown in Table 2, we can see this worst case is unlikely to be reached. With the help of functional information, we can further determine if the worst case is indeed possible.

4 Experimental Results

In this section, we present some experimental results on several ITC'99 benchmark circuits. We use commercial tools to perform power analysis, static timing analysis and path-delay ATPG. We use hypothetical data due to the lack of physical design information. Results are presented in Tables 4 and 5, respectively. All delays are shown in ns.

For each circuit, we randomly select some feed regions that could cause IR-drop on the corresponding victim gates. We also select a number of paths that have least slack times (false paths are removed). For each path that runs through any feed region, we perform the IR-drop analysis proposed above, and examine the corresponding increase of path delay and decrease of slack time.

In Tables 4, we show the path delay and slack time

Table 4. Results of path delays for ITC'99 circuits.

Before IR-drop analysis		After IR-drop analysis			
Path delay	Slack time	Path delay	Worst path delay	Slack time	Worst slack slack
Circuit b12, 100 paths					
0.727	5.272	0.913 (25.58%)	8.418 (1057.91%)	5.087 (-3.51%)	3.372 (-36.04%)
Circuit b14, 31 paths					
2.855	26.54	3.339 (16.95%)	10.207 (257.51%)	26.056 (-1.82%)	21.862 (-17.63%)
Circuit b15, 200 paths					
2.525	16.805	2.668 (5.66%)	14.248 (464.28%)	16.662 (-0.85%)	15.604 (-7.15%)
Circuit b17, 200 paths					
13.068	39.061	13.26 (1.47%)	54.655 (318.24%)	38.869 (-0.49%)	4.734 (-87.88%)

changes before our IR-drop analysis and after the analysis. In Columns 1 and 2, we show the average (over all paths considered) path delay and slack time before the IR-drop analysis, respectively. Corresponding results after analysis are shown in Columns 3 and 5, respectively. We also show the percentage of increase on path delay and decrease on slack time. In columns 4 and 6, we show the worst case path delay and slack time (on average), when a maximum number of aggressors are switching simultaneously in each feed region. It can be seen that the consideration of IR-drop using the proposed method can lead to a rather discernible increase on path delay. The decrease on slack time is not as discernible, though. However, we have observed that some paths' slack time become negative after the process, indicating that timing closure will most probably fail on these paths, which may need redesign. Worst case changes in both cases are significant. We expect a much higher impact on delays when physical design information is available, as reported by earlier work.

In Tables 5, we show the number of critical paths before and after the IR-drop analysis for a given threshold. In practice, a threshold is often used to determine if a path can be viewed as "critical". The threshold can be either set for path delay, or slack time. I.e., the paths whose delay exceed the path threshold will be considered. Similarly the paths whose slack time is less than the slack threshold will be considered.. We present these results, with threshold values in the table. It can be seen that after adding the delay caused by IR-drop, there is a significant increase on the number of critical paths. This indicates that some paths that are not originally identified as critical paths are now captured, hence possible under-test is avoided. This result also has an impact on path-delay ATPG, which is not shown in the table. We have observed that in most cases, the test

Table 5. Number of critical paths under threshold for ITC'99 circuits

Path delay threshold		Slack time threshold	
Before	After	Before	After
b12, Slack threshold: 5.9, Delay threshold: 0.5			
85	100	53	100
b14, Slack threshold:27, Delay threshold: 3.0			
13	18	19	23
b15, Slack threshold:17, Delay threshold: 2.5			
169	179	172	184
b17, Slack threshold:54, Delay threshold: 12.5			
100	129	174	180

patterns targeting original critical paths will have a much lower coverage when new paths are added due to the IR-drop analysis.

5 Conclusions

In this paper, we have analyzed the effect of IR-drop in selecting the critical paths for delay testing. We proposed a method to analyze the delay caused by power grid voltage drop. The method is based on statistical analysis and the calculation of the expected value of delay. Experimental results on ITC'99 benchmark circuits have shown that the proposed method can effectively capture the IR-drop effects in the path delay and can have a significant impact on the identification and analysis of critical paths.

References

[1] Z. Abuhamdeh et al., Characterize Predicted vs Actual IR Drop in a Chip Using Scan Clocks. *Proc. Int. Test Conf.*, 21.1, 2006.

[2] N. Ahmed, M. Tehranipoor and V. Jayaram, IR-drop Tolerant Transition Delay Fault Testing. *Proc. IEEE North Atlanta Test Workshop*, pp.112-121, 2007.

[3] Y-M Jiang and K-T Cheng, Analysis of Performance Impact Caused by Power Supply Noise in Deep Submicron Devices. *Proc. Design Automation Conf.*, pp. 760-765, 1999.

[4] Y-M Jiang, T. K. Young and K-T Cheng, VIP - An Input Pattern Generator for Identifying Critical Voltage Drop for Deep Sub-Micron Designs. *Proc. Int. Symp. Low Power Electronics and Design*, pp. 156-161, 1999.

[5] A. Kokrady and C.P. Ravikumar, Static Verification of Test Vectors for IR Drop Failure. *Proc. Int. Conf. CAD*, pp. 760-764, 2003.

[6] A. Krstic, Y-M Jiang and K-T Cheng, Pattern Generation for Delay Testing and Dynamic Timing Analysis Considering Power-Supply Noise Effects. *IEEE Trans. CAD*, vol 20, pp. 416-425, 2001.

[7] S. Kundu, Analysis of Power Grid of Microprocessors. *Proc. Int. Test Conf. Tutorial: TEST CIRCUIT MARGINALITIES*, 2005.

[8] K-S Leung, SPIDER: Simultaneous Post-Layout IR-Drop and Metal Density Enhancement with Redundant Fill. *Proc. Int. Conf. CAD*, pp. 33-38, 2005.
[9] D. Mitra, et al., Test Pattern Generation for Power Supply Droop Faults. *Proc. Int. Conf. on VLSI Design*, pp. 343-348, 2006.
[10] J. Saxena et al., A Case Study of IR-Drop in Structured At-Speed Testing. *Proc. Int. Test Conf.*, pp. 1098-1104, 2003.
[11] M. Sharma and J. Patel, Testing of Critical Paths for Delay Faults. *Proc. Int. Test Conf.*, pp. 634-641, 2001.
[12] M. Sharma and J. Patel, Finding a Small Set of Longest Testable Paths That Cover Every Gate. *Proc. Int. Test Conf.*, pp. 974-982, 2002.
[13] C. Tirumurti, et al., A Modeling Approach for Addressing Power Supply Switching Noise Related Failures of Integrated Circuits. *Proc. Design, Automation and Test in Europe*, pp. 1078-1083, 2004.
[14] B. Xue and D. M. H. Walker, I_{DDQ} Test Using Built-In Current Sensing of Supply Line Voltage Drop. *Proc. Int. Test Conf.*, 37.1, 2005.