

Path Selection and Pattern Generation for Dynamic Timing Analysis Considering Power Supply Noise Effects

Jing-Jia Liou, Angela Krstić, Yi-Min Jiang* and Kwang-Ting Cheng

Electrical and Computer Engineering Department, University of California, Santa Barbara

*Synopsys Inc., Mountain View, CA 94043

Abstract

Noise effects such as power supply and crosstalk can significantly affect the performance of deep submicron designs. These delay effects are highly input pattern dependent. Existing path selection and timing analysis techniques cannot capture the effects of noise on cell/interconnect delays. Therefore, the selected critical paths may not be the longest paths and predicted circuit performance might not reflect the worst-case circuit delay. In this paper, we propose a path selection technique that can consider power supply noise effects on the propagation delays. Next, for the selected critical paths, we propose a pattern generation technique for dynamic timing analysis such that the patterns produce the worst-case power supply noise effects on the delays of these paths. Our experimental results demonstrate the difference in estimated circuit performance for the case when power supply noise effects are considered vs. when these effects are ignored. Thus, they validate the need for considering power supply noise effects on delays during path selection and dynamic timing analysis.

1 Introduction

Continuous shrinking of device feature size, increased number of interconnect layers and gate density, increased current density and higher voltage drop along the power nets give rise to the noise effects such as power supply noise and crosstalk noise. Analysis shows that excessive noise can significantly affect the performance of deep submicron designs [1]. For example, with the increasing complexity of deep submicron designs, more devices are switching simultaneously and this results in increased power supply noise. The power supply noise reduces the actual voltage level reaching a device, which, in turn, leads to increased cell and interconnect propagation delays. This results in increased signal arrival times at the primary outputs and the next state lines and causes performance degradation. If the cell is a clock buffer or is in the timing-critical path, this delay deviation could cause serious clock skew problems or a non-trivial increase in the critical path delay. Similarly, increase in design density leads to more significant crosstalk effects between the interconnects that can again affect the performance of the designs.

The goal of timing analysis is to predict the performance of the design. However, traditional timing analysis techniques cannot consider noise effects on the performance of the design. This is because these effects are highly input pattern dependent. *Dynamic timing analysis* is used to predict the performance of the design by simulating a set of input patterns. To be able to give an accurate and efficient estimate of the performance, dynamic timing analysis must simulate a small set of patterns producing the worst-case path delays in the circuit. Therefore, this process involves selecting the right set of paths and generating the right set of patterns for them. Traditionally, path selection is performed using timing analysis tools by selecting all paths that exceed some percentage of the clock period (cut-off time). However, traditional false-path-aware timing analysis tools (such as PrimeTime [2]) cannot take into account the noise effects and therefore, can fail to include some critical paths into the target set. In this paper, we address the problem of path selection and pattern generation for dynamic timing analysis considering power supply noise effects. Our path selection technique can also be used for selecting a set of paths for delay testing considering power supply noise effects [3].

Many parametric variations, noise sources and modeling errors (caused by, for example, inaccurate RC extraction and reduction) are statistical in nature. Therefore, the use of statistical methods for timing analysis to in-

corporate statistical timing deviations caused by these sources seems to be inevitable. Our path selection technique considering power supply noise is based on a statistical timing analysis framework [4]. In statistical timing analysis, the delays of cells/interconnects are modeled as correlated random variables with known probability density functions (pdfs). Given these cell/interconnect delays, the cell level netlist and the clock period, statistical timing analysis can derive the pdfs of the signal arrival times, required times and slacks at internal signals and primary outputs. Noise effects, such as power supply noise, can cause perturbations of the delay random variables of the affected cells and/or interconnects. We use the statistical model for power supply noise proposed in [1] to derive the perturbed delay random variables of each cell/interconnect in the presence of power supply noise. By running the statistical timing analysis using the perturbed cell/interconnect delays we derive the slack distributions at internal nodes and primary outputs of the circuit. These slack distributions are then used in our critical path selection strategy. After the set of critical paths considering power supply noise has been selected, we perform dynamic timing analysis. For this purpose, for each selected path, we generate vector pairs that sensitize the path while producing the worst-case power supply noise effects on its propagation delay. Our experimental results demonstrate the advantage of considering power supply noise effects during the path selection process as well as during pattern generation for dynamic timing analysis. Because different nodes/interconnects in the circuit may have dramatically different performance sensitivity with respect to the power supply noise, the set of timing-critical paths can significantly differ for the case when the effects of power supply noise on propagation delays are considered from the case when the effects are ignored. The experimental results clearly indicate the difference and thus validate the need for considering power supply noise effects on delay during path selection.

2 Background

Power Supply Noise Estimation. Power supply noise is a result of switching activity in the circuit and it is highly input pattern dependent. In general, power supply noise includes two components: inductive ΔI noise and power net IR voltage drop. The inductive ΔI noise is caused by the change of the instantaneous current on either the package lead inductance or on wire/substrate inductance and it is proportional to $L \frac{di}{dt}$. On the other hand, the IR voltage drop is caused by the instantaneous current through the resistive power and ground lines. Recently, several techniques have been proposed for estimating power supply noise in deep submicron designs [5, 6, 7, 8]. The first three approaches use closed-form equations or efficient models to estimate the power supply noise at each cell based on the given input conditions. The fourth technique provides an accurate and efficient method for estimating the maximum power supply noise in the entire circuit. This technique uses a Genetic Algorithm-based (GA-based) [9] approach for generating a small number of input patterns that cause high power supply noise. For efficiency reason we incorporate this GA-based technique into our timing analysis flow.

Statistical Timing Analysis Framework. The core engine for our statistical path selection considering power supply noise is a statistical timing analysis framework [4]. To achieve a balance between accuracy and efficiency, we adopt a cell-based approach in building our framework. It requires pre-characterization of cells, i.e., building libraries of cell delays and output transition times (as random variables). The input transition time and output loading of the cells are used as indices for building these libraries. Since the goal of statistical timing analysis is to describe the

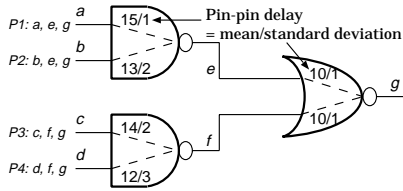


Figure 1: Statistical vs. nominal path selection method.

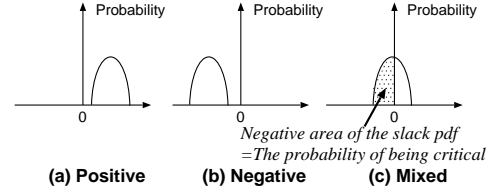


Figure 2: Types of slack distributions.

timing behavior of the circuit regardless of the applied input patterns, we adopt the worst-case analysis in building the cell libraries. The interconnect delay is also modeled as a random variable and is pre-characterized once the RCs are extracted. The random variables of the signal arrival times at cell/interconnect outputs are computed using the information on the arrival and transition times of the cell fanins as well as the information on the cell/interconnect delays [10]. Our framework uses Monte Carlo simulation based technique to approximate the pdfs of the signal arrival times at the internal signals and primary outputs. The convergence criteria are decided based on desired accuracy of results. Our framework also efficiently supports a large number of elements sharing the same correlation factor and it avoids the problem of the large size of covariance matrix used to model correlation between elements.

Statistical Model for Power Supply Noise and Its Delay Effects. To estimate the propagation delay increase of cells/interconnects (as a random variable) due to the power supply noise we use the technique proposed in [1]. It first builds a statistical model of the supply voltage. In this model, the supply voltage is represented as a random variable and its pdf is derived from the transistor-level simulation results using a small set of vectors [11] which cause high power supply noise. Next, by this statistical model, we can derive the perturbation on the pdfs of the propagation delays for the affected cells/interconnects. This is done by combining a set of delay-vs- V_{dd} curves (available in the cell library or derived during the cell library characterization phase) with the statistical model of the voltage level.

3 Path Selection

Due to a very large number of paths in practical designs, techniques for dynamic timing analysis and delay testing rely on selecting a small set of target paths for analysis/testing. The set of longest paths in the circuit can significantly differ for the case when noise effects on propagation delays are considered from the case when these effects are ignored. Existing path selection techniques cannot consider noise effects on the propagation delays and therefore, might not select the right set of paths. Also, the existing techniques for path selection are based on the results of nominal or worst-case timing analysis (i.e., the delay of each cell/interconnect is a single value - either its nominal delay or worst-case delay) instead of statistical timing analysis. This again can lead to selecting some not-so-critical paths while omitting other more critical ones for analysis/testing as illustrated by the following example.

Example 1 Consider the circuit in Figure 1. It has four paths connecting PIs (i.e., signals a, b, c, d) to PO (signal g): $P1, P2, P3, P4$. If the nominal pin-to-pin delays are used, the longest path is $P1$ with delay of 25 time units. On the other hand, if the 3σ pin-to-pin delays are used (e.g., the 3σ pin-to-pin delay from a to e is $15 + 3 \times 1 = 18$), the longest path becomes $P4$ with delay of 34 time units. If we can afford to select only one path in the circuit, then these two different delay models would result in a selection of completely different paths. As this example illustrates, conventional methods for path selection based on single-value analysis may fail to find the right critical paths. This observation motivates the need for a path selection methodology under the statistical timing analysis environment.

3.1 Statistical Method for Path Selection

Our goal is to find a minimum set of paths considering power supply noise such that sensitizing these paths leads to an accurate prediction of the circuit performance. This is equivalent to searching for *critical paths*. Critical paths are paths with delays exceeding a given limit which can be a certain percentage, e.g., 90%, of the intended clock period.

Critical Nodes. Critical paths are composed of *critical nodes*. Critical nodes are defined with respect to their slacks (*slack* is the difference of the

required time and the arrival time). If the slack of the node is negative, the node is a critical node. Here, the required times of all primary outputs and next state lines are set to the given limit of path delay. Under the statistical framework, the arrival times, the required times, and the slacks of internal nodes are all random variables. Therefore, a node could be critical for some circuit instances but non-critical for other circuit instances.

For any node in a circuit, there are three types of possible slack distributions as shown in Figure 2. If the entire pdf of a node's slack is in the positive region, as shown in Figure 2(a), we say the node has a positive slack. For all circuit instances, this node is non-critical and it will not be in any critical path. If a node has a negative slack (Figure 2(b)), the node is a critical node for all circuit instances. If the pdf of a node's slack covers both positive and negative regions as shown in Figure 2(c), we say the node has a mixed slack. The area of the pdf in the negative region is the probability of the node being a critical node. Note that if normal distributions are assumed, any slack will always have a non-zero probability of being in both positive and negative region. In that case, we can set a threshold for the classification. For example, if the threshold is set to 99.7%, then a node is classified as one with a positive slack if the area of the slack pdf in the positive region is over 99.7%.

Example 2 To illustrate different types of slack distributions and how they can be used for path selection, consider the circuit in Figure 1. The limit of path delay is set to 20.00 time units which is then the required time of primary output g . After statistical timing analysis, the mean and standard deviation of g 's slack random variable are -5.96 and 1.38 , respectively (assuming a normal distribution). As the 3σ bounds of this random variable are all in the negative region, by definition, g has a negative slack. On the other hand, f has a mixed slack ($-4.64/2.23$) as the pdf of f 's slack covers both positive and negative regions. If we perform Monte Carlo-based static timing analysis for 480 runs, the delay of $P1$ always exceeds the specified limit and thus is guaranteed to be a critical path (with a certain confidence level). On the other hand, the probability of $P2$ being critical is $439/480 = 91\%$ (in 439 out of 480 Monte Carlo runs it is a critical path). If the paths are ordered according to the runs (probabilities) in which the paths are critical, the order will be $P1(480), P3(469), P2(439),$ and $P4(373)$. If we would like to cover all paths whose probability of being critical is higher than 50%, then all paths should be tested. On the other hand, if we can afford to select only two paths, selecting $P1$ and $P3$ will maximize the probability of covering the most critical paths encountered in all circuit instances.

Path Selection Strategies. In our statistical path selection procedure, we first derive the cell/interconnect delay random variables by including the power supply noise effects on the propagation delays [1]. Next, using Monte Carlo sampling we derive the pdfs of signal arrival times, required times and slacks. The path selection procedure is then based on these pdfs. To select a set of most critical paths considering power supply noise effects, it is clear that we must select all paths on which every node has a negative slack. This is because the delays of these paths, if they can be sensitized, are guaranteed to exceed the specified limit. However, selecting paths containing only nodes with negative slacks is not sufficient. There are many paths which contain signals with mixed slacks. These paths have non-zero probabilities to be timing-critical. The probability of a path being timing-critical is equal to the smallest node probability for the nodes along the path. Path searching strategies are used to maximize the total probability of a selected path set.

We propose two path searching strategies. **Strategy 1** searches for paths whose probabilities of being timing-critical exceed a given threshold (e.g., 30%). This search can be done by searching for paths that contain nodes whose probabilities of being timing-critical exceed the given threshold. If

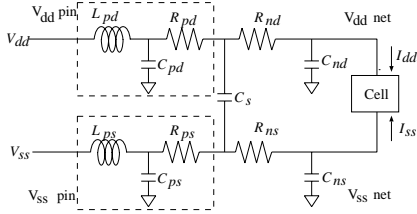


Figure 3: Circuit model for deriving power supply noise.

the given threshold is 100%, then the paths selected will be guaranteed to be timing critical as discussed at the beginning of this subsection. **Strategy 2** closely integrates path selection into Monte-Carlo-based statistical timing analysis and selects a path with the most negative slack in each Monte Carlo run (the slack here is a single value instead of a random variable). The union of the selected paths forms the set of paths for dynamic timing analysis. In general, with a low threshold, Strategy 1 will select more paths than Strategy 2, and very often cover most paths selected by Strategy 2. Therefore, Strategy 1 will provide better path selection results at the cost of more CPU time and memory for longer test sequences. Note that these strategies are used after the complete statistical timing analysis is done, so there is no explicit path enumeration at each run of Monte Carlo analysis.

4 Pattern Generation

Our pattern generation method for dynamic timing analysis consists of several steps. The first step is a pre-processing step and it is performed only once for each design. In this step, we extract the power net RC s and build a current/voltage waveform library for different cells. In the second step, for each selected path, we generate a partially specified input vector pair that sensitizes the path. In this step, we attempt to leave as many primary inputs unspecified as possible. In the third step, we assign the unspecified primary input values such that the power supply noise impact on the delays of the signals on the path is maximized.

Step 1. The circuit model used for deriving the current/voltage waveform libraries for each cell is shown in Figure 3. Symbols V_{dd} and V_{ss} are used to denote power and ground. In the following, we will discuss only the network used to model V_{dd} pin and segment (the model for V_{ss} is similar to V_{dd}). Each V_{dd} pin is modeled by an RLC network (L_{pd} , R_{pd} and C_{pd}). Symbol C_s is used to denote the substrate and on-chip decoupling capacitance. Symbols R_{nd} and C_{nd} are used to denote the effective resistance and capacitance of V_{dd} line from the V_{dd} pin to V_{dd} node of the cell, respectively. Power supply noise at the V_{dd} node of a cell can be computed by summing up the inductive ΔI noise and IR voltage drop along the power line segments from the V_{dd} pin to the V_{dd} node of the cell. Noise caused by the wire/substrate inductance is ignored, since it is substantially smaller than that caused by the package lead inductance. Though this package lead inductance is seen globally the same for all cells, it should be considered in this characterization phase because the final current waveforms are different for every cell, even with the same inductance value. In order to find the power supply noise, it is necessary to derive accurate current waveforms for all segments of V_{dd} and V_{ss} lines and pins, which depend on the charge and discharge current waveforms of all cells. For a given circuit with the netlist and its physical design, first the current waveforms for each cell with respect to a given input pattern are estimated. The current waveforms for all the cells are then used to compute the current waveforms through the V_{dd} and V_{ss} pins as well as all segments of V_{dd} and V_{ss} nets.

Step 2. Sensitizing a given path usually requires assignment of only a small number of primary input values. The unspecified values can be spec-

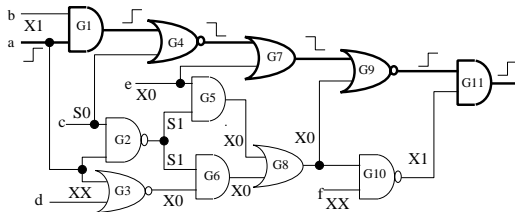


Figure 4: Minimally specified input vector pair.

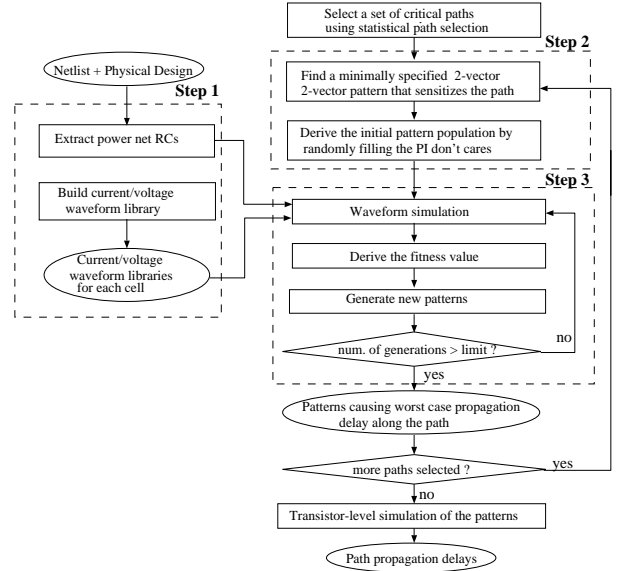


Figure 5: Dynamic timing analysis flowchart.

ified such that they activate the worst-case noise effects and produce the worst-case propagation delay on the path. Consider, for example, the circuit in Figure 4. The selected path is shown in bold line. There are 6 primary inputs in this circuit (marked with symbols a through f). In order to sensitize the path only the value of primary input a has to be fully specified. Different assignments of unspecified values can result in different path delays. This is because gate delays depend on the number of inputs that are switching and because different input vector pairs can also result in different power supply or crosstalk noise. For example, transistor level simulation [12] of a vector pair $V_1(abcdef) = \langle 011000, 111100 \rangle$ produces a path delay of 2.71 ns, while $V_2(abcdef) = \langle 000101, 111100 \rangle$ produces a path delay of 2.80 ns.

Step 3. For each selected path, our goal is to generate an input vector pair such that the impact of the power supply noise on the path delay is maximized. The assumption that a two-vector pair could excite the worst-case power supply noise is valid, since we assume that any vector can be initially applied to the sequential circuit. We extend the GA-based approach [8] to use the primary input don't care set for this purpose. The initial population for the GA-based iterative phase is derived from the partially assigned vector pair obtained in the previous step by randomly assigning the don't care values. Next, we use the cell current/voltage waveform libraries and the information about the extracted power net RC s derived in Step 1 to perform the waveform simulation of the current patterns. The fitness value of the pattern is calculated as a summation of the maximum power supply noise for the nodes on the path. In each iteration of the GA, we use the selection, crossover and mutation schemes suggested in [11] to generate a new pattern population. The GA-based iterative process stops when the number of generations exceeds a pre-specified limit. For the same path in Figure 4, this approach produces vector pair $V_3(abcdef) = \langle 010011, 111100 \rangle$ which results in a path delay of 2.99 ns. After the vector patterns for all selected paths have been generated, we use a transistor-level simulator [12] to predict the circuit performance. The flowchart of our dynamic timing analysis algorithm is shown in Figure 5.

5 Experimental Results

In this section, we describe our experimental results for path selection and pattern generation for dynamic timing analysis for ISCAS85 and ISCAS89 benchmark circuits. All results are obtained for 0.25 μm , 2.5 V technology.

To demonstrate the need for considering power supply noise effects during path selection process, we select three sets of critical paths. The first set of paths is selected using Strategy 1 with threshold set to 10%, which controls the minimum probability of nodes to be considered in path selection. We denote this set of paths as **S1**. The second set of paths (denoted as **S2**) is selected using Strategy 2. The paths in these two sets are selected considering cell/interconnect delays that take into account delay pertur-

Ckt	# cells	circuit performance for simulated patterns (ns)												# selected paths			CPU time (s)	
		Max delay						Ave Delay										
		Pattern set M1			Pattern set M2			Pattern set M1			Pattern set M2			S1	S2	N	path	GA
		S1	S2	N	S1	S2	N	S1	S2	N	S1	S2	N					
C1355	517	2.79	2.71	2.51	2.41	2.35	2.11	2.42	2.55	2.32	2.00	2.10	1.86	248	52	248	1.00	12
C1908	615	3.87	3.77	3.34	3.59	3.54	3.11	3.45	3.51	3.26	3.11	3.20	2.92	94	12	122	2.47	21
C2670	811	4.34	4.28	4.05	4.02	3.99	3.54	3.91	4.08	3.77	3.65	3.78	3.22	96	16	112	2.56	25
C3540	1135	6.55	6.55	5.95	6.03	6.03	5.31	6.21	6.21	5.64	5.88	5.88	4.71	16	16	16	7.05	54
s1196	487	2.21	2.21	1.90	1.88	1.88	1.69	2.00	2.01	1.71	1.70	1.71	1.50	28	27	29	1.92	9
s1238	540	2.86	2.79	2.46	2.83	2.66	2.31	2.63	2.71	2.30	2.43	2.51	2.07	7	3	10	1.88	12
s1488	608	4.85	4.80	4.50	4.51	4.33	4.21	3.83	3.92	3.80	3.71	3.82	3.71	114	12	142	2.64	18
s5378	1256	3.89	3.85	3.74	3.47	3.22	3.12	3.60	3.71	3.21	3.01	3.13	2.98	30	22	22	3.52	32
s9234	1192	6.17	6.14	5.90	5.81	5.78	4.91	5.78	5.88	5.21	5.31	5.55	4.70	90	8	114	4.77	28
s13207	2975	2.95	2.93	2.85	2.71	2.70	2.65	2.69	2.65	2.58	2.40	2.37	2.33	6	6	4	15.7	85
s38584	12705	18.1	17.4	16.2	16.5	16.1	14.6	15.2	16.9	13.8	14.2	15.8	12.1	7	2	7	52.2	102
Average (norm)	-	1.23	1.22	1.12	1.13	1.11	1.00	1.23	1.26	1.13	1.11	1.14	1.00	-	-	-	-	-

Table 1: Circuit delay for different sets of paths and different test vectors.

bations caused by power supply noise effects. The third set of paths is selected using nominal delays for cells/interconnects (power supply noise is not being considered). The required time for the outputs is set such that the number of selected paths in the third set approximately matches the number of paths selected by Strategy 1 (because of the presence of false paths we cannot easily select the same number of paths in these sets). We denote the third set of paths as N. This set represents the paths selected by the traditional method. For each of these three sets of paths, we generate two different sets of vectors. The first set of vectors (denoted as M1) is generated using the technique described in Section 4. Therefore, these vectors are generated such that they sensitize the paths and the don't care set at the primary inputs is used to maximize the power supply noise along the nodes in the paths. The second set of patterns is generated such that the vectors sensitize the paths and the don't care primary inputs are assigned randomly. We denote this set of vectors as M2. The two sets of patterns are next simulated using a transistor-level simulator (PowerMill) [12] to estimate the performance of the design.

The results of the experiment are shown in Table 1. Columns 3 through 8 show the delay of the longest path in the circuit. Columns 9 through 14 show the average delay for the tested paths. Columns 3, 4, and 5 show the propagation delays of the longest paths in sets S1, S2 and N with respect to the patterns generated under M1 strategy, respectively. Columns 6, 7, and 8 show the delays of the same sets of paths after applying the patterns generated using strategy M2 which represents the existing pattern generation methods. The results for the average delays based on the three path selection strategies with respect to M1 and M2 are shown in Columns (9–11), and (12–14), respectively. The last row of the table shows the average of the delays normalized with respect to the set of paths selected for nominal delays under strategy M2. The average is obtained by first finding the normalized values for each circuit and then the average for each column over the set of tested circuits. Columns (14–16) show the number of paths in each of the three sets. For each path we generate one 2-vector pattern. The experiments have also demonstrated the efficiency of our path selection and pattern generation process. We listed the CPU time used to perform path selection (STA path) and GA test generation process for each path (GA) at the last two columns. These two processes show the impact of using our statistical and GA methods in our dynamic simulation flow.

As the results illustrate, there is a significant difference in the estimated circuit performance when paths are selected considering power supply noise (columns under S1 and S2) vs. when the power supply noise effects are ignored (columns under N). For example, for circuit s38584 the performance is estimated as 18.1 ns (under pattern set M1, column S1) if power supply noise is taken into account during path selection and pattern generation. If power supply noise effects are considered only during pattern generation and not in the path selection process the performance is estimated as 16.2 ns (under pattern set M1, column N). If power supply noise is not considered neither during path selection nor during pattern generation, the performance is 14.6 ns (under pattern set M2, column N). This clearly indicates the need for considering noise effects. On the average, the estimated worst-case delay is 23% larger if power supply noise is taken into account during both path selection and pattern generation (as compared with the delay estimated without considering power supply noise). The delay is

12% larger if the noise is taken into account only during pattern generations. We would like to stress that the final part of this experiment is done by a full-circuit Powermill simulation. This complete simulation can consider various aspects of analog behavior of the circuit. For example, the correlation of power surge in different parts of the circuit is taken into account by the simulator automatically. In general, the measured worst-case delays should be considered accurate with the power supply model used.

6 Conclusions

Noise effects can significantly affect the performance of deep submicron circuits. Traditional critical path selection and timing analysis tools cannot consider these effects on the propagation delays of interconnects/cells. Therefore, the predicted longest delay of the circuit might be shorter than the true worst-case delay of the circuit. We have proposed a methodology for selecting critical paths and estimating the performance of the design that can take into account the impact of power supply noise effects on the performance of the circuit. We use statistical timing analysis as a core engine for our path selection strategy, together with statistical models for estimating the impact of power supply noise on the interconnect/cell delays. Also, we propose a technique for generating patterns for the selected paths such that they produce the worst-case noise effects on the delays. Our experimental results show a significant increase in the estimated circuit performance when power supply noise effects are considered.

References

- [1] Y.-M. Jiang and K.-T. Cheng. Analysis of Performance Impact Caused by Power Supply Noise in Deep Submicron Devices. *Proc. DAC*, pages 760–765, June 1999.
- [2] Synopsys. PrimeTime User Guide. May 1999.
- [3] A. Krstić, Y.-M. Jiang, and K.-T. Cheng. Delay Testing Considering Power Supply Noise Effects. *Proc. ITC*, pages 181–190, September 1999.
- [4] J.-J. Liou, A. Krstić, K.-T. Cheng, D. Mukherjee, and S. Kundu. Performance Sensitivity Analysis Using Statistical Methods and Its Applications to Delay Testing. *Proc. ASP DAC*, pages 587–592, January 2000.
- [5] R. Senthinathan and J. L. Prince. *Simultaneous Switching Noise of CMOS Devices and Systems*. Kluwer Academic Publishers, Boston, MA, 1997.
- [6] Y.-S. Chang, S. K. Gupta, and M. A. Breuer. Analysis of Ground Bounce in Deep Sub-Micron Circuits. *Proc. VTS*, pages 110–116, April 1997.
- [7] H. H. Chen and D. D. Ling. Power Supply Noise Analysis Methodology for Deep Submicron VLSI Chip Design. *Proc. DAC*, pages 638–643, June 1997.
- [8] Y.-M. Jiang, K.-T. Cheng, and A.-C. Deng. Estimation of Maximum Power Supply Noise for Deep Sub-Micron Designs. *Proc. of ISLPED*, pages 233–238, August 1998.
- [9] D. E. Goldberg and R. Burch. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, MA, 1989.
- [10] H. Edamatsu, K. Homma, M. Kakimoto, Y. Koike, and K. Tabuchi. Pre-Layout Delay Calculation Specification for CMOS ASIC Libraries. *Proc. ASP DAC*, pages 241–248, February 1998.
- [11] Y.-M. Jiang, K.-T. Cheng, and A. Krstić. Estimation of Maximum Power and Instantaneous Current Using A Genetic Algorithm. *Proc. CICC*, pages 135–138, May 1997.
- [12] Synopsys. PowerMill Reference Manual. May 1999.