

Timing Yield Estimation Using Statistical Static Timing Analysis

Min Pan¹, Chris Chong-Nuen Chu¹ and Hai Zhou²

¹ECE Department, Iowa State University, ²ECE Department, Northwestern University

Abstract—As process variations become a significant problem in deep sub-micron technology, a shift from deterministic static timing analysis to statistical static timing analysis for high-performance circuit designs could reduce the excessive conservatism that is built into current timing design method. In this paper, we address the timing yield problem for sequential circuits and propose a statistical approach to handle it. In our approach, we consider the spatial and path reconvergence correlations between path delays, set-up time and hold time constraints, as well as clock skew due to process variations. We propose a method to get the timing yield based on the delay distributions of register-to-register paths in the circuit. On average, the timing yield results obtained by our approach have average errors of less than 1.0% in comparison with Monte Carlo simulation. Experimental results show that shortest path variations and clock skew due to process variations have considerable impact on circuit timing, which could bias the timing yield results. In addition, the correlation between longest and shortest path delays is not significant.

1. Introduction

In recent technologies, intra-die variations are becoming a more dominant portion of the overall variability of device features. There are significant variations in circuit parameters (e.g., device size, ILD thickness, wire width and thickness) inside a die. Therefore, the static timing analysis needs to take these variations into consideration.

Conventional deterministic static timing analysis handles variability by analyzing a circuit at multiple process corners. Although this method is successful in dealing with variations between different dies on a wafer (inter-die variations), a worst-case analysis for variations inside a die (intra-die variation) will lead to very pessimistic analysis results because it assumes that all devices on a die have the worst-case characteristics. Another major drawback of this case-based static timing analysis is identifying incorrect critical paths. The critical paths are obtained by assuming all the devices have the same variation features. Therefore, the process variations can lead to the change of critical paths due to delay variations in different paths.

Compared with the deterministic approach, statistical static timing analysis is a probabilistic approach to analyze timing of circuits. In this approach, process variations are taken into consideration by modeling them with statistical distribution of process parameters. At the TAU 2004 Workshop, many researchers agreed that statistical timing eventually will come into play to reduce the excessive conservatism that is built into design timing now [15].

Many statistical timing analysis methods have been proposed in recent years [1-10]. In [4], a path based statistical timing yield computation was presented using an accurate delay model. In [5], a new circuit optimization method was proposed to reduce the number of near critical paths in a circuit, thereby improving the statistical delay of the circuit. However, these approaches have not considered the spatial and reconvergence correlations between parameter variations. In [6], A. Agarwal et al. proposed a method using statistical bounds and addressed the arrival time correlations due to path reconvergence. They furthered the method by considering the arrival time correlations due to spatial correlations in [7]. However, since it is a path-based method, a large number of paths need to be analyzed. In [8], a method based on Bayesian Networks was proposed for computing the exact probability distribution of the delay in a circuit. In [9] three novel algorithms were proposed for statistical timing analysis and parametric yield prediction of digital integrated circuits. In [10], Chang et al. proposed a PERT-like circuit graph traversal method to find the path delay distribution of a path from a source node to a sink node.

As far as we know, previous work just considered the problem of analyzing the timing of combinational circuits or paths. However, it is

not trivial to handle the timing yield problem of sequential circuits based on the results of previous work. There are many questions arising with the timing yield analysis of sequential circuits. How to deal with spatial and path reconvergence correlations between the path delays of all the registers? Is it enough to just consider the set-up time constraints (longest path delays)? If considering the hold time constraints (shortest path delays), is there any difference in the timing yield results? How could clock network affect the timing yield of a circuit under process variations? How are the longest path and shortest path correlated?

In this paper, we propose a systematic method to estimate the timing yield of a sequential circuit using statistical static timing analysis. We capture the random characteristic of process variations considering the spatial correlations as well as path reconvergence. The delay of each register-to-register path is modeled as multivariate normal distribution. Based on the delay distributions of gates and interconnects, we find the longest path and shortest path delay distributions for all sink registers by just traversing the graph which models the sequential circuit once. Then we apply Clark's approximation to find the distribution of timing margin for the whole circuit given a clock cycle time. The timing yield is the probability that both set-up time margin and hold time margin are greater than zero. Thus, for a given clock cycle time, we can find the timing yield for the whole circuit. The experimental results show that our approach achieves accurate timing yield with average error less than 1.0%. In addition, hold time constraints and clock skew due to process variations have considerable impact on the timing yield so that they should be included carefully in timing yield analysis.

The remainder of the paper is organized as follows. In the next section, we formulate the statistical timing yield analysis problem for sequential circuits. Section 3 presents parameter variations model and circuit model used in our algorithm. In Section 4, we propose an algorithm to handle the problem formulated in Section 2. Section 5 contains experimental results and detailed discussions followed by our conclusions in Section 6.

2. Problem Formulation

2.1 Timing Yield of Sequential Circuits

Considering the process variations, the process parameters are no longer fixed values but random variables with certain statistical distributions. Subsequently, the gate and interconnect delays, which are functions of process parameters, are also random variables. Therefore, given a clock cycle time, the timing yield of the circuit is not deterministic but a random variable with some statistical distribution and it is natural to estimate the timing yield using statistical approaches.

Timing yield of sequential circuits is based on set-up time constraints and hold time constraints. The circuit must satisfy that the signal should transfer from one register to the next fast enough so that it arrives at the second register at least one set-up time before the next clock edge. In addition, because of the hold time constraints, the signal cannot transfer too fast so that the second register can latch the value correctly. Therefore, the circuit functions correctly only when both the set-up time and hold time constraints are satisfied. In conventional deterministic STA, the longest path and shortest path are fixed in a given circuit. However, with process variations, the shortest path and longest path are no longer fixed. So the path-based approach has to consider many potentially feasible longest paths and shortest paths [4]. An approach is needed to consider the longest and shortest paths under process variations. In addition, we need to find the yield function based on the distributions of the longest and shortest path delay distributions.

Furthermore, clock skew also has impact on the timing yield. Considering the process variations in recent advanced technologies, it is very hard to control the clock skew. The intra-die variation has great impact on clock skews [12][13][14]. When we consider the timing of

the circuit, it is necessary to consider the clock skew between registers. The clock skew will affect the timing by offsetting the relative signal arrival time from one register to the other. Because clock skew is comparable to the shortest path delay in a large circuit, the clock skew problem is more serious for shortest paths.

2.2 Definitions

Before formulating the statistical timing yield analysis problem, we first give the definitions used throughout the work.

- R:** the set of all the registers in the circuit;
- R_i:** the set of all the registers in R that have paths from them to register i with only combinational logic between;
- T_{clk}:** the clock cycle time;
- t_{clk2q_j}:** the delay of the output of register j from clock edge;
- Path_{ji}:** the set of all possible paths from register j to register i;
- tPath_{ji}:** the set of path delays of Path_{ji};
- t_{clk_j}:** the clock arrival time at register i;
- t_{clk_{skew}_{ji}}:** the clock skew between register j and i (t_{clk_j} - t_{clk_i});
- t_{setup_i}:** the set-up time for register i;
- t_{hold_i}:** the hold time for register i.

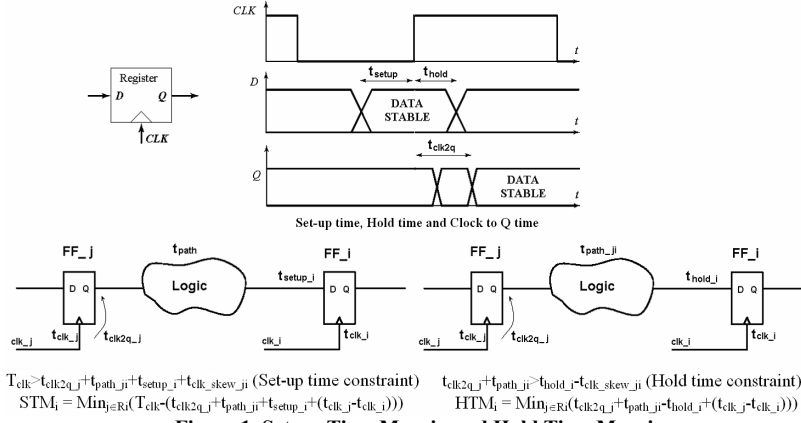


Figure 1. Set-up Time Margin and Hold Time Margin

Set-up time violation is that the logic is too slow for the correct logic value to arrive at the input to the register before one set-up time ahead of the clock edge. Assuming j is source register and i is sink register, the constraint to prevent set-up time violation is:

$$T_{clk} > t_{clk2q_j} + t_{Path_{ji}} + t_{setup_i} + t_{clk_skew_{ji}} \\ \Rightarrow T_{clk} - (t_{clk2q_j} + t_{Path_{ji}} + t_{setup_i} + (t_{clk_j} - t_{clk_i})) > 0 \quad (1)$$

Hold time violations occur when race-through is possible. The constraint to prevent hold time violation is:

$$t_{clk2q_j} + t_{Path_{ji}} > t_{hold_i} - t_{clk_skew_{ji}} \\ \Rightarrow t_{clk2q_j} + t_{Path_{ji}} - t_{hold_i} + (t_{clk_j} - t_{clk_i}) > 0 \quad (2)$$

The left sides of (1) and (2) are the time margins under the set-up time constraints and hold time constraints. We define them as Set-up Time Margin (STM) and Hold Time Margin (HTM). The STM and HTM for register i are defined as:

$$STM_i = \min_{j \in R_i} (T_{clk} - (t_{clk2q_j} + t_{Path_{ji}} + t_{setup_i} + (t_{clk_j} - t_{clk_i}))) \quad (3)$$

$$HTM_i = \min_{j \in R_i} (t_{clk2q_j} + t_{Path_{ji}} - t_{hold_i} + (t_{clk_j} - t_{clk_i})) \quad (4)$$

Except T_{clk} , all other time variables are circuit timing characteristics determined by the circuit itself. Figure 1 shows the derivations of Set-up Time Margin (STM) and Hold Time Margin (HTM) for register i .

We want STM and HTM of all registers greater than zero. In order to satisfy the set-up time constraints and hold time constraints, we need to find the minimum STM and HTM for the whole circuit. Therefore, we define STM and HTM of the whole circuit as STM_C and HTM_C :

$$STM_C = \min_{i \in R} (STM_i) \quad (5)$$

$$HTM_C = \min_{i \in R} (HTM_i) \quad (6)$$

Because STM_C and HTM_C are the minimum Set-up Time Margin and Hold Time Margin for all registers, if both of them are greater than zero, the timing constraints of the whole circuit are satisfied. In order to find the probability that both random variables STM_C and HTM_C are greater than zero, we define another random variable TM as $TM = \min(STM_C, HTM_C)$, the time margin for the whole circuit. Given a clock cycle time T_{clk} , the probability density of TM is $tm(T_{clk})$. If $TM(T_{clk}) > 0$, the circuit satisfies the timing constraints. So the timing yield for a given clock cycle time is just the probability that $TM(T_{clk}) > 0$:

$$Yield(T_{clk}) = P(TM(T_{clk}) > 0) \quad (7)$$

2.3 Statistical Timing Yield Analysis Problem

The statistical timing yield analysis problem for sequential circuits is formulated as following:

For a sequential circuit, given a clock cycle time T_{clk} , analyze the timing for the paths between all pairs of registers and get the statistical distribution of Set-up Time Margin (STM) and Hold Time Margin (HTM) for all registers. Based on all STM s and HTM s, find the timing yield $Yield(T_{clk})$.

The way we find the timing yield is: based on all STM_i and HTM_i , we compute the STM_C and HTM_C for the whole circuit. Then we combine the distribution for STM_C and HTM_C to get the probability distribution of TM . Finally, for any given clock cycle time T_{clk} , find the timing yield $Yield(T_{clk})$:

$$Yield(T_{clk}) = P(TM(T_{clk}) > 0) = \int_0^{\infty} tm(T_{clk}) dt \quad (8)$$

3. Parameter Variations Model and Circuit Model

3.1 Parameter Variations Model

To consider the spatial correlations between path delays, we employ the model proposed in [7]. In the paper, a multi-level grid structure was proposed to handle the spatial correlations. For each grid on every level, we have an independent random variable with normal distribution associated with each process parameter. The path delay variation due to intra-die parameter variations is modeled as a linear combination of multiple standard normal independent random variables:

$$D_{path} = D_{path}^0 + \sum_{i \in \Psi} \sum_{j \in \Omega} K_{ij} \cdot \Delta P_{ij} \quad (9)$$

where D_{path} is the path delay, D_{path}^0 is the path delay under nominal values of process parameters, Ψ is the set of process parameters, Ω is the set of grids that the path traverses, K_{ij} is the pre-characterized sensitivities of process parameter i at grid j , and ΔP_{ij} is the change in process parameter i at grid j . The mean and variance of the path delay is:

$$\mu_{D_{path}} = D_{path}^0 \quad (10)$$

$$\sigma^2 = \sum_{i \in \Psi} \sum_{j \in \Omega} K_{ij}^2$$

3.2 Circuit Model

In our algorithm, the sequential circuit is modeled as a directed graph, each cell (gate) maps to a vertex, and each interconnect between two cells maps to a directed edge. When we consider the delay of an interconnect, we chop it down into several segments by the boundary of the grids. Therefore, each segment is inside a single grid so that it will have the variation features captured by that grid.

In order to traverse the graph and get all the path delays efficiently, we break each vertex representing a register into two vertices with no edge in between. The reason to do this is to transfer the original graph into a DAG (directed acyclic graph). We can apply the topological sort on the DAG thereafter. Then traverse all the vertices once using topological sorting order to get the longest delay (shortest delay) for all the registers. This is why our algorithm has the complexity which is linear in the number of cells and interconnects.

For the clock network, we choose the H-tree topology which ends at the center of every grid at the bottom level in the multi-level grid model. We should mention here that although H-tree is a very simple model, it can address the impact of clock skew on timing yield.

4. Statistical Timing Yield Analysis Algorithm

In this section, we will present our algorithm to handle the statistical timing yield analysis problem.

First, we construct the graph representation for a sequential circuit by mapping cells to vertices and interconnects to directed edges. Then, we transfer this graph to a DAG. As mentioned in Section 3, we break each vertex representing a register into two so that all the cycles in the original graph are broken.

After we have the DAG representation of the circuit, we apply topological sort on all the vertices based on their connections. If there is

a path from any vertex A to vertex B, vertex A should appear before B in the topological order. In fact, the topological sort assures that all edges go from the left to the right in the sorted vertices list. With this order, every time we compute the delay at a cell, we already have the delay of all the cells whose fanouts are fanins of the current cell.

Then we traverse the whole circuit using topological order to get the longest and shortest delays for every cell. We need to do two kinds of operations at a cell, one is *Sum*, and the other is *Max* or *Min*. First, for every input, we find its delay by summing up the delay at the output of its parent cell and the delay of the interconnect in between. Then we calculate the longest (shortest) delay which is *Max*(*Min*) delay of all input delays after we get the delays for all the inputs. Finally, we sum this delay with the delay of the cell itself to find the delay at the output of this cell. From the model in Section 3, we know all delays are linear combinations of independent random variables. Therefore, we need to do *Sum* and *Max*(*Min*) on a set of correlated random variables.

For *Sum* operation, since we express delays as linear combinations of independent random variables with standard normal distribution, we just need to sum up coefficients of all the independent random variables:

$$\begin{aligned} d_i &= d_0 + k_{i1}\Delta p_1 + \dots + k_{in}\Delta p_n \\ d_j &= d_0 + k_{j1}\Delta p_1 + \dots + k_{jn}\Delta p_n \\ d_i + d_j &= d_0 + (k_{i1} + k_{j1})\Delta p_1 + \dots + (k_{in} + k_{jn})\Delta p_n \end{aligned} \quad (11)$$

For *Max*(*Min*) operation, [10] employed Clark's approximation [11] to find *Max* of several correlated random variables. We also choose Clark's approximation to compute *Max* when traversing the graph. In fact, the *Max* operation takes care of the reconvergence correlations between paths. In [10], how to compute *Max* by Clark's approximation is detailed, so we do not repeat it here. We just want to point out that we can do *Min* the same way as we do *Max*: $Min(x,y) = -Max(-x,-y)$.

In the end, after the traversal, we already find the longest and shortest delay distributions for all the registers. Then for any given clock cycle time, we employ Clark's approximation again to get the distribution of *STMC* and *HTMC*, where *STMC* and *HTMC* are set-up time margin and hold time margin of the whole circuit. The last step is to combine the two distributions to get the distribution of time margin $TM = Min(STMC, HTMC)$. We just multiply the probability of *STMC* and *HTMC* to get the probability distribution for *TM*. That is to say, we assume *STMC* and *HTMC* are independent. We will show in detail why we can make this assumption in Section 5. Since probability distribution of *TM* is known, we can get the yield by the probability of $TM > 0$: $Yield(T_{clk}) = P(TM > 0)$.

Table 1. Comparison results of our algorithm with Monte Carlo Simulation

Benchmark		Error compared with MC		Run Time (s)		
Name	#Cells	#Grids	Avg Error (%)	Max Error (%)	Our	MC (30000)
S38584	23815	256	0.6620	1.1920	63.39	5901.99
S38417	20705	256	0.5686	1.1535	60.36	4508.59
S35932	17793	256	0.3465	0.6848	50.27	4359.75
S15850	10369	256	0.6446	1.1784	20.10	1997.09
S13207	8260	256	0.4128	1.2017	27.09	2431.65
S9234	5825	64	0.0777	0.3467	3.51	877.58
S5378	2958	64	0.2816	0.7833	1.92	540.33
S1196	547	16	0.0685	0.3233	0.17	114.71

Table 2. The timing yield when there is no set-up time violations

Circuits	S38584	S38417	S35932	S15850	S13207
Yield	96.77%	97.30%	98.67%	98.94%	99.28%

Since the algorithm just traverses the DAG once, the run time of the algorithm is $O(m+n)$ which is linear to the number of cells m and interconnects n in the sequential circuit.

5. Experimental Results and Discussion

The proposed algorithm was implemented in C and tested on the edge-triggered ISCAS89 benchmark circuits. All experiments were run on a Linux machine with 3.00 GHz CPU and 2GB memory. We use the layout information from [10]. In order to show the effect of clock skew due to process variations, we build an H-tree clock network. We should mention here that although the clock skew due to process variations only causes a few percentage of error on timing yield in small circuits, such as ISCAS89 circuits, process variations can result very large clock

skew which affects timing yield severely in large high-performance circuits. The process parameters used in experiments are gate length and width, oxide thickness, wire width, wire thickness and ILD thickness. The number of grids satisfies that each grid contains less than 100 cells.

First, we compare the timing yield results obtained by our algorithm with Monte Carlo simulation. We chose to run 30,000 iterations for Monte Carlo simulation. The average and maximum errors of timing yield are shown in Table 1. We can see that our algorithm can get very accurate timing yield results. Among all the circuits, the largest average error is 0.6620% (S38584) and the largest maximum error is 1.2017% (S13207). We also provide the run time for both methods. Our algorithm is much faster than Monte Carlo simulation. The circuit with the longest run time, S38417, was analyzed in only 63.39 seconds using our algorithm, while Monte Carlo simulation needed 5901.99 seconds.

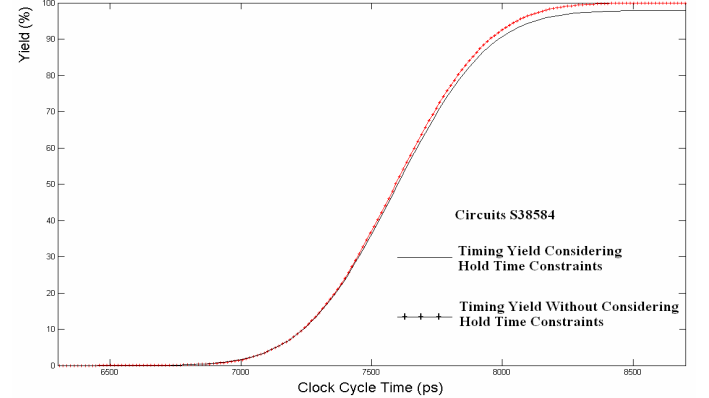


Figure 2. Considering hold time constraints vs. not considering (S38584)

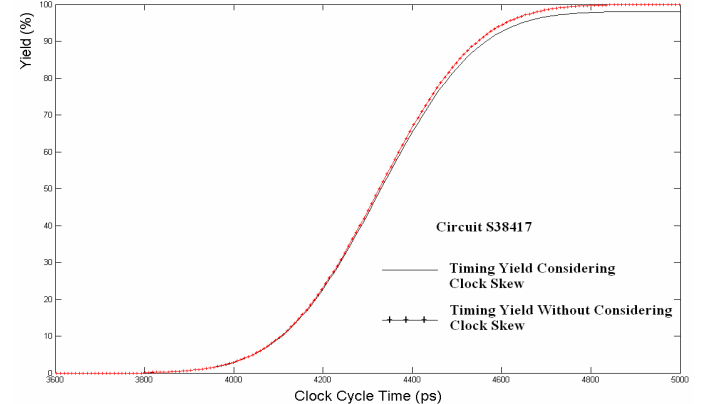


Figure 3. Considering clock skew vs. not considering clock skew(S38417)

Second, we show the necessity of considering hold time constraints. The timing yield results with and without considering hold time constraints are compared. From Figure 2, we see that the timing yield curve of circuit S38584 obtained by only considering set-up time constraints (longest path delays) is always above the curve obtained by considering both set-up and hold time constraints. Specially, results show that no matter what clock cycle time we set, the yield cannot reach 100% because of the possibility of the hold time violation. The gap between the two curves is caused by hold time violations. Table 2 shows the timing yield (obtained by Monte Carlo Simulation) for test circuits when setting clock cycle time very large so that no set-up time violation will occur. The reason for this is that the hold time violation is caused by the shortest path delay. If the hold time margin is less than zero, increasing clock cycle time will not help to fix the timing problem.

Third, we show the difference between including clock skew due to process variations and assuming no clock skew. Figure 3 shows the difference between timing yield curves with and without clock skew caused by process variations for S38417. The results indicate that if we neglect the clock skew due to variations, the timing yield we get will be too optimistic. The reason is that clock network variations will increase the standard deviations and shift the mean values of *STM* and *HTM*. Because *HTM* has standard deviation much smaller than that of *STM*, the clock skew effect has very significant impact on the hold time constraints while only little impact on the set-up time constraints. Although the difference caused by clock skew due to variations is only

a few percent here, in large circuits with very complicated clock network, it would be very dangerous to neglect it.

Fourth, in Section 4, we mentioned that we use the independent assumption rather than Clark's approximation [11] to compute $Min(STM_C, HTM_C)$. Now we show why we do not use Clark's approximation to compute $Min(STM_C, HTM_C)$. Table 3 gives the values of standard deviations of STM_C and HTM_C for the five largest test circuits. We can see that the standard deviation of STM_C is much larger than that of HTM_C . If we use Clark's approximation to compute the timing yield, we assume STM_C and HTM_C are normal distributed, calculate the mean and standard deviation of time margin for the whole circuit, $TM=Min(STM_C, HTM_C)$, then assume its distribution is normal to get the probability of $TM>0$. But this assumption is only valid when STM_C and HTM_C have mean values close to each other with similar standard deviations or they have mean values far from each other. However, if they have mean values very close to each other and very different standard deviations, the distribution of TM is far from normal and skews to the left. So the probability of $TM>0$ obtained by Gaussian function has significant error. In order to explain this effect clearly, we investigate $Min(x1,x2)$ of two normal random variables $x1$ and $x2$. First, we fix $x1$ with standard normal distribution ($\mu_1=0, \sigma_1=1$) and $x2$ with $\mu_2=0$. Then we change σ_2 and get the distribution of $Min(x1,x2)$ using Monte Carlo simulation. In Figure 4, the solid line is the normal distribution with the same μ and σ as $Min(x1,x2)$ and the dotted line is the distribution of $Min(x1,x2)$. As we can see that the distribution of $Min(x1,x2)$ skews to the left. When σ_2 is comparable to σ_1 , the normal distribution can be a good approximation. But when σ_2 is much larger than σ_1 , the normal approximation is far from accurate. (In fact, this is the case for $Min(STM_C, HTM_C)$ because $\sigma(STM_C)$ is much larger than $\sigma(HTM_C)$) If we use the normal approximation to get the yield, we will underestimate the yield because of the skewness of $Min(STM_C, HTM_C)$ when STM_C and HTM_C have mean values close to each other. Experiments show that the underestimation can be as large as 15%, so we cannot use Clark's approximation to compute $Min(STM_C, HTM_C)$.

Table 3. Standard deviations of STM_C and HTM_C

Benchmark	σ_{STM_C}	σ_{HTM_C}	ρ	$(\sigma_{STM_C}^2 + \sigma_{HTM_C}^2) / 2\rho\sigma_{STM_C}\sigma_{HTM_C}$
S38584	279.68	40.78	0.142	24.66
S38417	172.23	19.84	0.128	34.36
S35932	232.73	24.98	0.107	44.04
S15850	226.08	33.97	0.264	12.89
S13207	215.31	19.39	0.042	133.26

Fifth, we investigate the dependence of STM_C and HTM_C . Table 3 also presents the values of correlation coefficient ρ for test circuits. We also show $(\sigma_{STM_C}^2 + \sigma_{HTM_C}^2) / 2\rho\sigma_{STM_C}\sigma_{HTM_C} >> 1$ for all five cases. In Clark's paper [11], mean v_1 and variance v_2 of the Max of two random variables are:

$$\begin{aligned}
 a^2 &= \sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2 \\
 \alpha &= (\mu_1 - \mu_2) / a \\
 v_1 &= \mu_1\Phi(\alpha) + \mu_2\Phi(-\alpha) + a\phi(\alpha) \\
 v_2 &= (\mu_1^2 + \mu_2^2)\Phi(\alpha) + (\mu_2^2 + \sigma_2^2)\Phi(-\alpha) + (\mu_1 + \mu_2)a\phi(\alpha) \quad (12)
 \end{aligned}$$

The small error caused by neglecting ρ will not cause big error in calculating the distribution. In large circuits, the longest path and shortest path will not have high correlations in most cases. Even when there is high correlation between them, we can easily employ Monte Carlo simulation to get the distribution of $Min(HTM_C, STM_C)$ since it is

very fast to do Monte Carlo simulation on two random variables to get the distribution of their Min . The experimental results show that we can get very accurate timing yield by assuming HTM_C and STM_C are independent for all the benchmark circuits.

6. Conclusion and Future Work

In this work, we formulate the statistical timing yield analysis problem for sequential circuits. To solve this problem, we propose an algorithm which considers spatial and path reconvergence correlations of parameter variations, statistical longest and shortest path of the whole circuit and the clock skew caused by process variations to get the timing yield of the circuit. Experiments show that the timing yield obtained by the algorithm is very accurate compared with Monte Carlo simulation but takes much less running time. In addition, comparative results also show that the shortest paths and clock skew have considerable impact on the timing yield result. For the Clark's approximation, although the normal assumption will not cause significant errors most of the time, it is not valid when computing $Min(STM_C, HTM_C)$ to get the timing yield.

In our algorithm, we just construct the H-tree to simplify the problem of clock skew and address the importance of clock skew to timing yield. However, in high performance circuits, the clock network is much more complicated (many buffers inserted and with different topologies). Therefore, more work needs to be done to carefully incorporate the clock skew due to process variations into timing yield computation. Another issue here is that Clark's approximation is not always safe to use in the timing yield analysis. We believe some other techniques such as skew-normal distribution could achieve better accuracy and be more robust to handle the $Max(Min)$ problem of correlated multivariate random variables.

Reference

- [1] R.-B. Lin; M.-C.Wu, "A new statistical approach to timing analysis of VLSI circuits", Proc. Int. Conf. on VLSI Design, 1998.
- [2] M. Berkelaar, "Statistical Delay Calculation, a Linear Time Method," Proceedings of TAU 97, Austin, TX, December 1997
- [3] M. Orshansky, K. Keutzer, "A general probabilistic framework for worst-case timing analysis", Proc. DAC 2002.
- [4] A. Gattiker, S.Nassif, R.Dinakar, C.Long "Timing Yield Estimation from Static Timing Analysis," Proc., ISQED 2001
- [5] X. Bai, et al., "Uncertainty-aware circuit optimization," Proc. DAC 2002.
- [6] A. Agarwal, et al., "Computation and Refinement of Statistical Bounds on Circuit Delay," DAC 2003, pp. 348-353.
- [7] A.Agarwal, et al., "Statistical timing analysis for intra-die process variations with spatial correlations," Proc. ICCAD2003, pp. 900 - 907
- [8] Bhardwaj, S.; Vrudhula, S.B.K.; Blaauw, D., "TAU- Timing Analysis Under Uncertainty" Proc. ICCAD2003, pp. 615 - 620
- [9] Jess, J.A.G. et al., "Statistical timing for parametric yield prediction of digital integrated circuits", Proc. DAC 2003, pp. 932 - 937
- [10] Chang, H.; Sapatnekar, S.S., "Statistical timing analysis considering spatial correlations using a single pert-like traversal", Proc. ICCAD2003, pp. 621 - 625
- [11] C. E. Clark, "The Greatest of a Finite Set of Random Variables," Operations Research, vol. 9, pp. 85-91, 1961.
- [12] S. Zanella, et al. "Analysis of the Impact of Intra-die Variance on Clock Skew", IEEE Transactions on Semiconductor Manufacturing, pp. 401-407, 2000.
- [13] V. Mehrotra and D. Boning, "Technology Scaling Impact of Variation on Clock Skew and Interconnect Delay", IITC2001, pp. 122-124.
- [14] Agarwal, A.; Blaauw, D.; Zolotov, V., "Statistical clock skew analysis considering intra-die process variations", Proc. ICCAD 2003, pp. 914 - 921.
- [15] <http://www.eetimes.com/showArticle.jhtml?articleID=18311086>

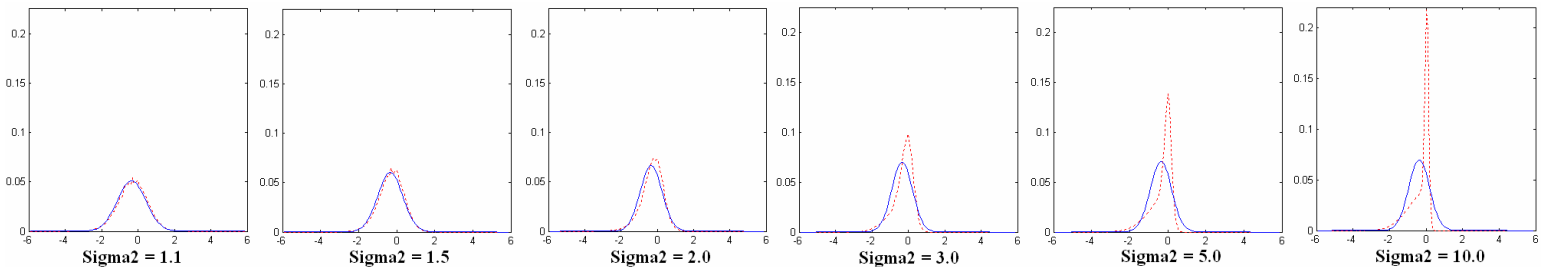


Figure 4. The distribution of $Min(x1,x2)$

Dotted line: the distribution of $Min(x1,x2)$ by Monte Carlo Simulation; Solid line: the normal distribution with the same μ and σ as $Min(x1,x2)$