Reduced Delay Uncertainty in High Performance Clock Distribution Networks

Dimitrios Velenis[1], Marios C. Papaefthymiou[2] and Eby G. Friedman[1]

[1]Department of Electrical
and Computer Engineering
University of Rochester
Rochester, NY 14627-0231

[2]Department of Electrical Eng.
and Computer Science
University of Michigan
Ann Arbor, MI 48109-2122

Abstract

The design of clock distribution networks in synchronous digital systems presents enormous challenges. Controlling the clock signal delay in the presence of various noise sources, process parameter variations, and environmental effects represents a fundamental problem in the design of high speed synchronous circuits. A polynomial time algorithm that improves the tolerance of a clock distribution network to process and environmental variations is presented in this paper. The algorithm generates a clock tree topology that minimizes the uncertainty of the clock signal delay to the most critical data paths. Strategies for enhancing the physical layout of the clock tree to decrease delay uncertainty are also presented. Application of the methodology on benchmark circuits demonstrates clock tree topologies with decreased delay uncertainties of up to 90%. Techniques to enhance a clock tree layout have been applied on a set of benchmark circuits, yielding a reduction in delay uncertainty of up to 48%.

1. Introduction

The primary characteristic of the microelectronics revolution is the rapid decrease in device size, producing phenomenal increases in circuit density, functionality, and operational clock frequencies [1, 2]. The scaling of device geometries supports the system-on-a-chip integration of multiple subsystems [3], greatly increasing the number of on-chip clocked elements. This development has resulted in hundreds of thousands of elementary operations being executed in sequences specified by application-specific algorithms and controlled by a clock signal, operating within time periods well below a nanosecond [4]. These constraints require tight timing control of the arrival times of the clock signal at the many registers throughout an integrated circuit. Deviations of the clock signal from the target delay can cause incorrect data to be latched within a register, resulting in the system malfunctioning.

Uncertainty of the clock signal delay is introduced by a number of factors that affect the clock distribution network, examples of which include process and environmental parameter variations [5, 6, 7] and interconnect noise [8]. Effects such as the non-uniformity of the interconnect lines and interlevel dielectric variations [9] introduce uncertainty in the delay of the clock signal arriving at different registers. Environmentally induced parameter variations caused by changes in the ambient temperature [10] and external radiation [11] also produce delay uncertainty. On-chip noise due to inductance effects [12, 13] and coupling among interconnects [14] introduces additional clock signal delay uncertainty as the clock frequencies increase deep into the multi-gigahertz frequency range. The sensitivity of a clock distribution network to these effects has become an issue of fundamental importance to the synchronous design problem [15, 16, 17].

An algorithm that improves the tolerance of a clock distribution network to delay uncertainty is presented in this paper. The algorithm focuses on the topological design of a clock distribution network, extracting the clock tree topology based on the temporal criticality of the data paths. The concept of the algorithm is summarized in Section 2. The steps of the algorithm are described in Section 3. The application of the proposed algorithm to benchmark circuits in order to enhance the clock tree topology is discussed in Section 4. Enhancements to the physical layout of the clock tree are reviewed in Section 5. Finally, some conclusions are presented in Section 6.

2. Concept of the Algorithm

The most crucial effect of the uncertainty introduced in clock signal delays is the increased delay uncertainty be-

This research was supported in part by the Semiconductor Research Corporation under Contract No. 99-TJ-687, the DARPA/ITO under AFRL Contract F29601-00-K-0182, grants from the New York State Office of Science, Technology & Academic Research to the Center for Advanced Technology - Electronic Imaging Systems and to the Microelectronics Design Center, and by grants from Xerox Corporation, IBM Corporation, Intel Corporation, Lucent Technologies Corporation, Eastman Kodak Company and Photon Vision Systems, Inc.

tween the arrival times of different clock signals that drive sequentially-adjacent registers connected by a combinational path. The more strict the setup and hold time constraints of a combinational data path, the more sensitive the timing of a data path is to delay uncertainty. A small difference in the clock signal delay can violate these constraints and cause a circuit to malfunction. The clock signal is distributed along different paths within a clock tree. These paths share a *common part* of the clock tree from the source of the clock signal to a branch node, as shown in Fig. 1. At a branch node, the paths split and the clock signal propagates along different, *non-common parts* of the tree to arrive at the individual registers. As shown in Fig. 1, the effects of process and environmental parameter variations (PEPV) on the common part of the clock tree introduce identical delays to those clock signals driving sequentially-adjacent registers [4]. Alternatively, along the non-common part of the clock tree, PEPV may introduce different clock delays and thereby cause a violation of the strict timing constraints of the critical data paths.

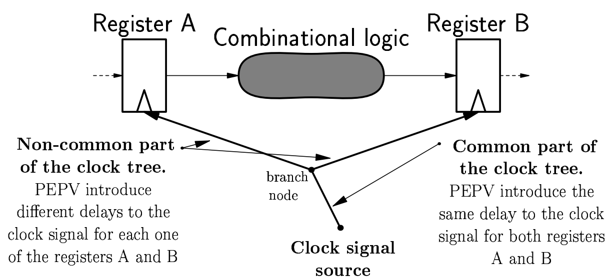


Figure 1. Introduction of different clock signal delays to the non-common portions of the clock tree.

The clock tree topology (CTT) that specifies the hierarchy of the branch nodes within the clock tree can greatly affect the delay uncertainty introduced along the clock paths [18]. In particular, as the common portion of two paths in a clock tree increases, the delay uncertainty between the leaves of these paths is likely to decrease. The common portion of the two paths can be increased by separating these paths from a branch node *deeper* within the clock tree (closer to the leaf registers). The algorithm presented in this paper relies on this principle to generate a clock tree topology with improved tolerance to PEPV. The objective of the algorithm is to minimize the delay uncertainty of the sensitive data paths within a circuit. In this way, the overall tolerance of a circuit to delay uncertainty can be improved.

A synchronous digital circuit is represented in the algorithm as an edge-weighted graph G, which is called an *uncertainty graph*. An example of an uncertainty graph representation is shown in Fig. 2. Each node u in the graph G denotes a register in the circuit. Each edge $u \to v$ in G denotes a combinational path between the registers corresponding to u and v in the original circuit. The weight $w(u, v)$ of each edge represents the tolerance of the corresponding data path

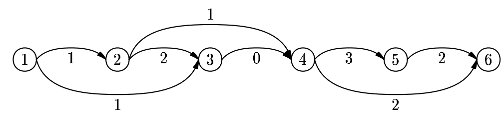


Figure 2. Graph representation of a circuit.

to PEPV which imposes a constraint on the delay uncertainty of the clock signals driving the registers u and v. In particular, for the circuit to function correctly, this uncertainty must not exceed $w(u, v)$. For example, the path corresponding to the edge 3→4, as shown in Fig. 2, is critical, since the path can tolerate zero uncertainty in the clock delays from the clock source to the bounding registers. Alternatively, path 4→5 can tolerate up to 3 tu (time units) of delay uncertainty.

Integer-valued time units are used in this example and in the rest of the paper to improve the clarity of the presentation. In the implementation of the algorithm, the tolerance of the data paths to delay uncertainty is represented using real numbers.

The algorithm relies on a topological delay uncertainty metric to generate clock trees that satisfy targeted uncertainty constraints. Specifically, given two paths, the delay uncertainty between the clock signals at the corresponding leaf nodes is assumed to be equal to the number of internal nodes within the tree (or branch nodes) in the *non-common* portions of the paths. The basic assumption underlying this metric is that as the number of non-common tree nodes between two paths increases, these paths share a smaller portion of the clock tree and the delay uncertainty between these paths therefore increases.

A clock tree topology that satisfies the delay uncertainty constraints of the graph shown in Fig. 2 is illustrated in Fig. 3. Consider, for example, the critical data path 3→4. This path can tolerate zero delay uncertainty between the clock signals arriving at the registers 3 and 4. The clock paths driving these registers split at the internal node 7 and arrive at registers 3 and 4. There are no non-common branch nodes between these two clock signal paths, therefore, these paths share the greatest portion of the clock tree. For data path 2→4, the clock paths split at the internal node 8. The number of non-common branch nodes of the two paths is one (node 7) which is equal to the delay uncertainty constraints for this data path. For data path 4→6, the paths from the clock source split at node 9, and the number of non-common branch nodes is two (nodes 7 and 8) which is also equal to the edge weight $w(4, 6)$.

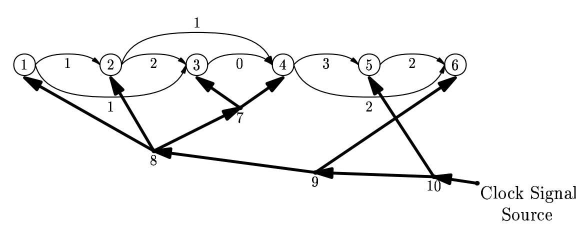
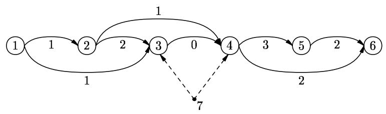
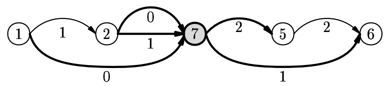


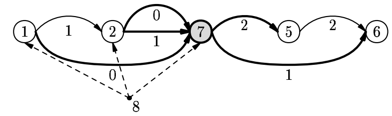
Figure 3. Clock tree topology for the circuit shown in Fig. 2.



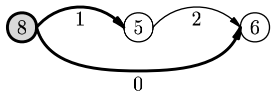
(a) **1st Iteration:** The clock signal is distributed to the registers of the most critical data path 3→4 from branch node 7.



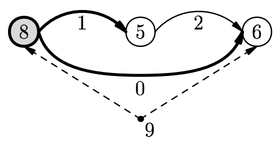
(b) Branch node 7 replaces the nodes that it drives within the graph. The weights of the redirected arches (**bold**) are reduced by one.



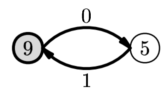
(c) **2nd Iteration:** The clock signal is distributed to the registers of the most critical data paths 1→7 and 2→7 from branch node 8.



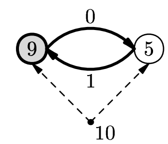
(d) Branch node 8 replaces the nodes that it drives within the graph. The weights of the redirected arches (**bold**) are reduced by one.



(e) **3rd Iteration:** The clock signal is distributed to the registers of the most critical data path 8→6 from branch node 9.



(f) Branch node 9 replaces the nodes that it drives within the graph. The weights of the redirected arches (**bold**) are reduced by one.



(g) **4th Iteration:** Only one node (10) is inserted in the graph. The algorithm terminates.

Figure 4. Iterations of the algorithm to reduce the input graph to a single node.

3. Description of the Algorithm

The algorithm presented in this section extracts the clock tree topology (CTT) by determining the hierarchy of the branch nodes of the tree such that the clocked elements of the most critical data paths share the greatest portion of the clock tree. The extracted topology can be further applied in the process of clock tree synthesis to generate a clock distribution network with improved tolerance to PEPV. Based on the information characterizing the topology of a clock tree, a physical layout can be developed which includes the interconnect, the placement of the branch nodes, and the location of the inserted clock buffers [16]. A technique that applies the concept of this methodology to produce a physical clock tree layout with reduced delay uncertainty is demonstrated in Section 5.

The algorithm starts by iteratively selecting from the uncertainty graph the registers of the data paths with the minimum tolerance to PEPV. These paths correspond to those edges with the minimum edge weight. In each iteration, a new branch node is introduced, and the clock signals are distributed from that node to the selected registers. The selected register nodes are replaced in the graph by a branch node. The edges entering or leaving the replaced nodes are redirected to the introduced branch node, and the edge weights are adjusted to reflect the new tolerance of these edges. The algorithm continues until only one node remains in the graph. The clock tree topology that satisfies all of the uncertainty constraints is obtained by unfolding the computation to establish the connections among the hierarchically introduced branch nodes.

The execution of the proposed algorithm on the uncertainty graph shown in Fig. 2 is illustrated in Fig. 4. The algorithm starts with the graph shown in Fig. 4(a). The minimum-weight edge in this graph is between nodes 3 and 4. The clock signal is distributed to these nodes from the new branch node 7. Node 7 is inserted in the uncertainty graph as shown in Fig. 4(b), replacing the selected nodes 3 and 4. The edges leaving or entering nodes 3 and 4 are redirected to node 7. The weights of these edges are reduced by 1 tu, the amount of uncertainty introduced by the branch node 7. The iterative application of this basic procedure continues until only one node remains in the graph (node 10) as shown in Figs. 4(c) through 4(g). At this point, the algorithm extracts the final clock tree topology, which is shown in Fig. 3. Note in Fig. 3 that nodes 3 and 4 (corresponding to the most critical data path with zero tolerance to PEPV) share the greatest portion of the clock tree from the clock signal source to branch node 7. In the case of a less critical data path such as the path between nodes 4 and 6, the clock paths to the registers have in common a smaller portion of the clock tree.

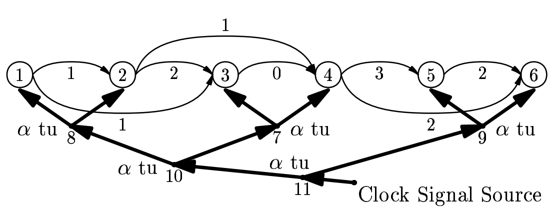
The correctness of the CTT generation algorithm can be proved by an inductive argument showing that after each iteration the generated clock tree satisfies all relevant uncertainty constraints. The algorithm has polynomial complexity, terminating in $O(n^2)$ steps, where n is the number of nodes in the uncertainty graph. The number of iterations is n, and within each iteration, the number of updates is proportional to n.

4. Enhanced clock tree topology

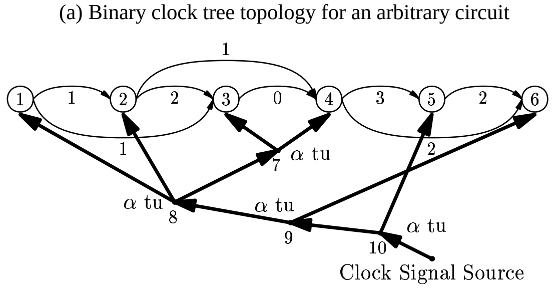
The process in which the proposed algorithm reduces the non-common portion of the clock tree that drives the critical data paths of a circuit is illustrated in Fig. 5. The topology generated by the proposed algorithm is compared with a binary tree topology under the assumption that the delay uncertainty between the clock signals immediately following a branch node is constant and equal to α tu. In this case, in a binary clock tree as shown in Fig. 5(a), the clock delay uncertainty for the data path 1→3 is 3α due to the branch nodes 7, 8, and 10. In the extracted clock tree topology as shown in Fig. 5(b), the corresponding delay uncertainty for the data path 1→3 is 2α due to the branch nodes 7 and 8. A 33% reduction in delay uncertainty is therefore introduced to those clock signals that drive the data path 1→3. In the same way, the delay uncertainty for the data paths 2→4 and 4→6 is reduced by 33% and 25%, respectively. Therefore, the timing margins for those data paths in the circuit shown in Fig. 5(b) can be less strict than for the circuit shown in Fig. 5(a). In this example, the clock period is not decreased since the delay uncertainty for the most critical data path 3→4 is the same for both clock tree topologies.

The proposed algorithm has been tested on a number of benchmark circuits (see Table 1). In these tests, the average reduction in delay uncertainty for a set of critical data paths has been determined. By reducing the delay uncertainty of these critical data paths, the overall timing constraints can be relaxed, thereby reducing the clock period and improving the overall circuit performance.

In evaluating the benchmark circuits it is assumed that the original clock tree topology is a balanced tree. The reduction in delay uncertainty is determined for four different branching factors (BF) of the balanced tree. The branching factor of a tree is the number of branches leaving from a branch node within a tree. The results of these experiments are listed in Table 1. It is shown that the delay uncertainty of the critical data paths can be reduced by up to 90%. Note in Table 1 that the smaller the branching factor of the original clock tree topology, the greater the reduction in delay uncertainty. As the branching factor decreases, the number of branch nodes within the clock tree increases and the tree becomes deeper. More non-common branch nodes can, therefore, be removed between the clock paths that drive the registers of the critical data paths, achieving a higher reduction in delay uncertainty. When the branching factor increases, the number of non-common branch nodes between the paths is reduced, and the achieved reduction in delay uncertainty is smaller.



(a) Binary clock tree topology for an arbitrary circuit



(b) Algorithmically extracted clock tree topology for the same circuit

Figure 5. Comparison between a binary tree and the algorithmically extracted CTT.

Table 1. Reduction in delay uncertainty of the most critical data paths. BF describes the branching factor of the original binary tree.

Benchmark Files	Number of Registers	Avg. reduction (%) of delay uncertainty for a set of critical data paths			
		BF = 2	BF = 4	BF = 8	BF = 16
S27cp	20	33.3%	25%	0%	0%
S386	20	39.5%	18.7%	12.5%	0%
mm4a	23	72.9%	50%	37.5%	0%
S1196	46	83.3%	66.7%	50%	50%
S1238	46	60%	30%	0%	0%
mult16b	48	83%	66.7%	50%	50%
mult32a	66	84.5%	71.1%	58.9%	50%
S838_1	67	58.9%	35.7%	13.2%	5.9%
S953	68	66.4%	45.2%	23.8%	21.4%
S641	77	84.8%	71.8%	60.4%	50%
sbc	120	84.3%	70.2%	57.1%	50%
S9234	209	82%	66.3%	52.1%	44.6%
S5378	246	84.2%	70.3%	58.1%	48.8%
S38584_1	446	90%	80%	75%	66.7%
diffeq	454	87.7%	77.6%	65.6%	60.6%
dsip	644	88.7%	79.4%	66.7%	64.8%
bigkey	683	88.7%	79.4%	66.7%	64.8%

As shown in Table 1, in certain smaller circuits the reduction in delay uncertainty is zero when the branching factor is high. In these specific cases, the clock paths driving the registers of the critical paths already share the greatest common portion of the clock tree, therefore a decrease in delay uncertainty is not possible. Alternatively, the larger circuits have a deep clock tree even in those cases where the branching factor is high. A significant reduction in delay uncertainty in these circuits is shown for all of the branching factors.

5. Enhanced clock tree layout

The proposed methodology can also be exploited within the layout domain to further reduce delay uncertainty. The placement of the branch nodes of the clock tree is determined in order to reduce the length of the non-common portion of the clock tree for those clock paths that drive the registers of the critical data paths. The delay uncertainty of the clock signals driving these critical data paths is thereby reduced.

The input to the branch node placement process is a set of physical coordinates that specify the locations of the registers within a circuit. Each register is, therefore, represented as a terminal point. The routing of the clock lines is only allowed along the grid formed by the intersection of the horizontal and vertical lines through the terminal points. As shown in

Table 2. Reduction in wire length and delay uncertainty for the most critical data paths

Data Paths	Wire length of the non-common portion of the clock tree			Delay uncertainty between clock signal paths		
	MWL tree	RDU tree	Reduction	MWL tree	RDU tree	Reduction
3 → 4	7.3	4.9	32%	3.6	2.0	43%
11 → 4	7.9	5.5	30%	4.0	2.5	35%
2 → 3	13.5	11.1	17%	6.9	5.5	20%
1 → 3	12.6	10.2	19%	6.5	5.3	17%
10 → 3	7.4	5.7	22%	3.5	3.2	10%
6 → 4	12.4	12.6	-1%	4.1	3.8	5%
8 → 3	1.9	1.9	0%	0.8	0.8	0%
11 → 3	0.6	0.6	0%	0.3	0.3	0%
Average	7.9	6.5	17%	3.7	2.9	21%

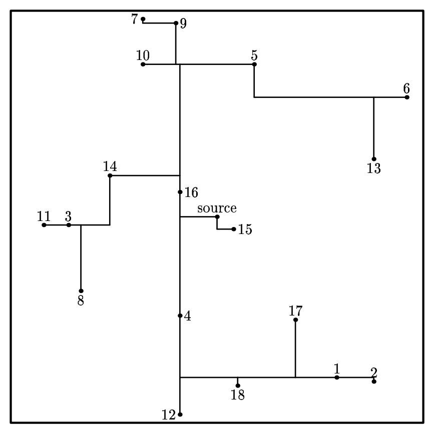
[19], a Minimal Rectilinear Steiner Tree exists in this grid which connects all of the terminal points. The root of the tree is the source of the clock signal which is assumed to be at the center of a square plane that contains all of the terminal points.

Alternatively, the non-common part between the clock paths that drive the registers of the critical data paths can be minimized rather than the overall wire length of the clock tree. Since the non-common portion of the clock tree is reduced, the delay uncertainty of the clock signal on these clock paths is also reduced. The delay uncertainty of a line segment is determined by a normal distribution function with a zero mean and a variance proportional to the length of the segment. The total delay uncertainty along a clock path is the sum of the delay uncertainties of all of the segments along that path. Furthermore, the delay uncertainty between two different clock paths is equal to the sum of the delay uncertainty values of all of the segments within the non-common portion of these paths. A similar model was proposed by Fisher and Kung in [20].

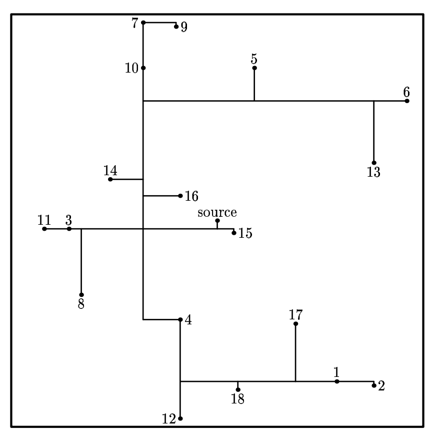
The proposed methodology has been tested on a number of benchmark circuits. The generated clock tree layouts are compared with minimal wire length trees to determine the reduction in delay uncertainty. A minimal wire length (MWL) tree for a benchmark circuit is shown in Fig. 6(a). A reduced delay uncertainty (RDU) tree, based on the model proposed in this paper for the same circuit is shown in Fig. 6(b).

The length of the non-common portion of the clock tree and the delay uncertainty of the critical paths for these two different layouts are compared and these results are listed in Table 2. The length of the non-common portion of the clock paths is reduced on average by 17% with an average reduction in delay uncertainty of 21%. The overall wire length of the RDU tree shown in Fig. 6(b) is increased by 4.5% as compared with the wire length of the MWL tree shown in Fig. 6(a). The increased wire length produces an increase in the non-common portion of the clock tree for certain, less critical, data paths. The reduction of delay uncertainty in these paths is, therefore, small or zero, as listed in Table 2 for data paths 6→4, 8→3, and 11→3.

The average reduction in the delay uncertainty and wire length of the non-common portion of the clock tree for a set of benchmark circuits is listed in Table 3. Note in Table 3 that the overall wire length of the RDU trees increases as compared with the wire length of the MWL trees. A trade-off, therefore, exists between a reduction in delay uncertainty



(a) Minimum wire length (MWL) tree



(b) Reduced delay uncertainty (RDU) tree

Figure 6. Comparison of the layout of a clock distribution network based on two different design objectives

Table 3. Reduction in delay uncertainty and wire length of the non-common portion of the clock tree for a set of benchmark circuits

Circuit	Increase in wire length for RDU trees	Average wire length of the non-common portion of the clock tree			Average delay uncertainty between clock signal paths		
		MWL tree	RDU tree	Reduction	MWL tree	RDU tree	Reduction
ex6	+13%	5.1	4.8	6%	2.5	1.2	54%
bbara	+12%	8.2	5.4	34%	2.5	1.2	48%
s420	+5%	5.9	5.1	13%	3.2	1.8	45%
opus	+1%	6.1	5.5	10%	2.7	1.8	33%
S420_1	+13%	9.1	9.1	0%	3.7	2.6	30%
ex4	+5%	8.0	6.4	19%	2.8	2.2	21%
s208	+7%	6.9	7.0	-1%	2.4	2.0	14%
dk16	+8%	5.0	4.7	5%	1.8	1.6	11%
dk17	+25%	6.6	5.8	11%	2.1	1.9	6%

and an increase in wire length when applying this clock tree topology synthesis methodology.

6. Conclusions

A methodology for generating a clock distribution network with high tolerance to process and environmental variations is presented. An algorithm that extracts a clock tree topology in order to minimize the delay uncertainty of the clock signals that drive the most critical data paths is presented. The hierarchy of the branch nodes of the clock tree is determined such that the clocked elements of the most critical data paths share the greatest portion of the clock tree. Simulation results from the application of the algorithm to benchmark circuits demonstrate significant improvement in the tolerance of a circuit to process and environmental variations. Further experiments in the layout domain demonstrate a significant reduction in the delay uncertainty of the critical data paths with a small penalty on the overall wire length of the clock tree.

References

[1] G. E. Moore, "Progress in Digital Integrated Circuit," *Proceedings of the IEEE International Electron Devices Meeting*, pp. 11-14, December 1975.

[2] SIA, "The National Technology Roadmap for Semiconductors," Technical report, Semiconductor Industry Association, 1997.

[3] H. B. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI*. Addison-Wesley Publishing Company, 1990.

[4] I. S. Kourtev and E. G. Friedman, *Timing Optimization Through Clock Skew Scheduling*, Norwell, Massachusetts: Kluwer Academic Publishers, 2000.

[5] S. Natarajan, M. A. Breuer, and S. K. Gupta, "Process Variations and Their Impact on Circuit Operation," *Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 73-81, November 1998.

[6] S. R. Nassif, "Delay Variability: Sources, Impacts and Trends," *Proceedings of the IEEE International Solid-State Circuits Conference*, pp. 368-369, February 2000.

[7] P. Zarkesh-Ha, T. Mule, and J. D. Meindl, "Characterization and Modeling of Clock Skew with Process Variations," *Proceedings of the IEEE Custom Integrated Circuits Conference*, pp. 441-444, May 1999.

[8] K. T. Tang and E. G. Friedman, "Interconnect Coupling Noise in CMOS VLSI Circuits," *Proceedings of the ACM International Symposium on Physical Design*, pp. 48-53, April 1999.

[9] V. Mehrotra, S. L. Sam, D. Boning, A. Chandrakasan, R. Vallishayee, and S. Nassif, "A Methodology for Modeling the Effects of Systematic Within-Die Interconnect and Device Variation on Circuit Performance," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 172–175, June 2000.

[10] S. Sauter, D. Schmitt-Landsiedel, R. Thewes, and W. Weber, "Effect of Parameter Variations at Chip and Wafer Level on Clock Skews," *IEEE Transactions on Semiconductor Manufacturing*, Vol. 13, No. 4, pp. 395–400, November 2000.

[11] J. F. Chappel and S. G. Zaky, "EMI Effects and Timing Design for Increased Reliability in Digital Systems," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol. 44, No. 2, pp. 130–142, February 1997.

[12] Y. I. Ismail and E. G. Friedman, "Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 8, No. 2, pp. 195–206, April 2000.

[13] M. W. Beattie and L. T. Pileggi, "Inductance 101: Modeling and Extraction," *Proceedings of the ACM/IEEE Design Automation Conference*, pp. 323–328, June 2001.

[14] K. T. Tang and E. G. Friedman, "Delay and Noise Estimation of CMOS Logic Gates Driving Coupled Resistive-Capacitive Interconnections," *Integration, The VLSI Journal*, Vol. 29, No. 2, pp. 131-165, September 2000.

[15] E. G. Friedman, *High Performance Clock Distribution Networks*, Norwell, Massachusetts: Kluwer Academic Publishers, 1997.

[16] E. G. Friedman, *Clock Distribution Networks in VLSI Circuits and Systems*, Piscataway, New Jersey: IEEE Press, 1995.

[17] M. Nekili, Y. Savaria, and G. Bois, "Design of Clock Distribution Networks in Presence of Process Variations," *Proceedings of the IEEE Great Lakes Symposium on VLSI*, pp. 95-102, February 1998.

[18] J. L. Neves and E.G. Friedman, "Buffered Clock Tree Synthesis with Non-Zero Clock Skew Scheduling for Increased Tolerance to Process Parameter Variations," *Journal of VLSI Signal Processing*, Vol. 16, Numbers 2/3, pp. 149-161, June/July 1997.

[19] M. Hanan, "On Steiner's Problem With Rectilinear Distance," *SIAM Journal of Applied Mathematics*, Vol. 14, pp. 255-265, March 1966.

[20] A. L. Fisher and H. T. Kung, "Synchronizing Large VLSI Processor Arrays," *IEEE Transactions on Computers*, Vol. 34, No 8, pp. 734-740, August 1985.