Nano-scale VLSI Clock Routing Module based on Useful-Skew Tree Algorithm

Chew Eik Wee[1], Ch'ng Heng Sun[2], Nasir Sheikh-Husin[3], Mohamed Khalil Hani[4]

VLSI-ECAD Research Laboratory (P04-Level 1),
Microelectronic and Computer Engineering Department (MiCE),
Faculty of Electrical Engineering (FKE),
Universiti Teknologi Malaysia.

[1]*Chew Eik Wee (eikweechew@hotmail.com)*
[2]*Ch'ng Heng Sun(chnghengsun@yahoo.com.sg)*
[3]*Nasir Shaikh-Husin(nasirsh@utm.my)*
[3]*Mohamed Khalil Hani (khalil@fke.utm.my)*

Abstract

Clock routing is critical in nano-scale VLSI circuit design. Clock routing needs to be precise to minimize circuit delay. Clock signals are strongly affected by technology scaling, the long global interconnect lines become highly resistive as line dimensions are decreased. The control of clock skew can also severely limit the maximum performance of the entire system and create catastrophic race conditions in which an incorrect data signal may latch within a register. Thus, we propose a clock routing synthesis module that applies non-zero skew (or called useful-skew) method to reduce the system-wide minimum clock period to improve the performance of synchronous digital circuit. We implemented Useful-Skew Tree (UST) algorithm which is based on the deferred-merge embedding (DME) paradigm, as the clock layout synthesis engine. The synthesis module is integrated with the UTM in-house design graph accelerator to enhance the computation performance. The novel contribution of this work is clock skew scheduling is performed simultaneously with clock tree routing. This way, the computation result of proposed synthesis module can generate a clock signal distribution routing path with minimum wire length and, ensures the reliability of data synchronization for nano-scale VLSI design.

Keywords:

Clock Routing, Synthesis Module, Useful Skew, Clock Skew Scheduling, Useful-Skew Tree.

1. Introduction

In a synchronous digital system, the clock signal is used to define a time reference for the movement of data within the system. Since this function is vital to the operation of a synchronous system, much attention has been given to the characteristics of these clock signals and the routing path used in their distribution. These clock signals are particularly affected by technology scaling, in that long global interconnect lines become much more highly resistive as line dimensions are decreased. Increment in resistance introduces greater propagation delay along the interconnect lines.

Due to differences in interconnect delays on the clock distribution network, clock signals do not arrive at all of the flip-flops at the same time. Thus, there is a *skew* between the clock arrival times at different latches. Let X_i and X_j denote clock propagation delay from the clock-source to flip-flop FF_i and FF_j respectively. The clock skew is defined as *skew* $= X_i - X_j$. In synchronous digital systems, the circuit performance is directly proportional to its clock frequency. The clock skew and the logic delay between two adjacent sequential elements directly determine the lower bound of the clock period, or hence the upper bound of the clock frequency. This is seen from the following well-know inequality [1][2]:

$$\text{clock period} \geq t_d + t_{skew} + t_{su} + t_{ds}$$

where t_d is the delay on the longest path through combinational logic, t_{skew} is the clock skew, t_{su} is the set-up time of the synchronizing elements and t_{ds} is the propagation delay within the synchronizing element.

High-performance clock design is an area of active research. Previous researches focus on elimination of the clock skew ($t_{skew} = 0$) and reduce the terms t_{su} and t_{ds} with advances in VLSI fabrication technology, to maintain the minimization of clock period. Recently, some of the researchers treat the clock skew (t_{skew}) as a manageable resources and applying the $t_{skew} < 0$ to the synchronizing circuit, to further reduce the value of clock period. This negative value of clock skew ($t_{skew} < 0$) is known as useful-skew.

Clock routing is critical in high performance VLSI circuits design. Clock routing needs to meet skew constraint and circuit delay. Interconnect lines become highly resistive as line dimensions are decreased. This increased line resistance is one of the primary limitations for clock distribution networks on maximizing the performance of synchronous integrated circuit.

Due to nano-scale technology below 0.2um process, the interconnection delays contribute significantly to the signal propagation delay, and achieving an exact zero skew become more and more difficult. Finally, the control of any differences in the delay of the clock signals (or skew) can

severely limit the maximum performance of the entire system as well as create catastrophic race condition in which an incorrect data signal may latch within a register.

Clock signal is a major power consumer also. It switches at every clock cycle. The dynamic power due to the capacitance switching forms a dominant part of system power dissipation.

In this paper, a complete architecture of clock routing CAD sub-system for nano-scale VLSI design is proposed. (Figure 1) Previous researchers concentrate on developing their clock signal distribution technique in separate areas. Their efforts involve clock scheduling, clock signal routing with general skew constraint clock, initial clock tree topology generation, etc. We are first to introduce a complete clock routing CAD sub-system, implementing several clock signal distribution techniques introduced by previous researchers. The proposed sub-system manages to compute an optimized allowable clock period, and simultaneously synthesis a clock signal distribution routing path. Due to limitation of the paper size, the scope of discussion in this paper focuses on useful-skew tree construction module only.

The remainder of this paper is organized as follows. In Section II, the literature review of research work is presented. In Section III, the algorithm of useful-skew tree (UST) based on deferred-merge embedding (DME) paradigm is presented and in section IV, the proposed implementation work is shown.

2. Literature Review

High-performance clock design is an area of active research. Due to the ever-increasing die sizes and the continued scaling of devices and interconnects, the control of clock skew in a clock distribution network is rapidly becoming critical design problem. Previous work in this area can be divided into two categories: focusing either on clock skew optimization without considering layout synthesis, or on clock routing with simplified skew constraints.

The concept of scheduling the system-wide clock skews for improved performance while minimizing the likelihood of race condition was first presented by [3]. Fishburn presents a methodology in which a set of linear inequalities are solved using standard linear programming techniques in order to determine each clock signal path delay from the clock source to every clocked register. [4] also present an algorithm to optimize the clock cycles based on linear programming. [5] improves Fishburn's methodology based on the model given in Sakallah for determining an optimal clock schedule by selectively generating the shortest path constraints, permitting the inequalities describing the timing characteristics of each local data path to be solved more efficiently. [6] and [7] show that the clock skew problem can be solved using efficient graph-theoretic techniques in polynomial time. The idea of using graph methods is to take advantage of the structure of the problem to arrive at an efficient solution. [8] extend the Deoker's work to determine an optimal clock skew scheduling that is tolerant to process variations.

These techniques determine the skews among clock pins optimized, for example, performance and robustness. However, these approaches generated a clock schedule without considering its impact on the layout synthesis of the clock net.
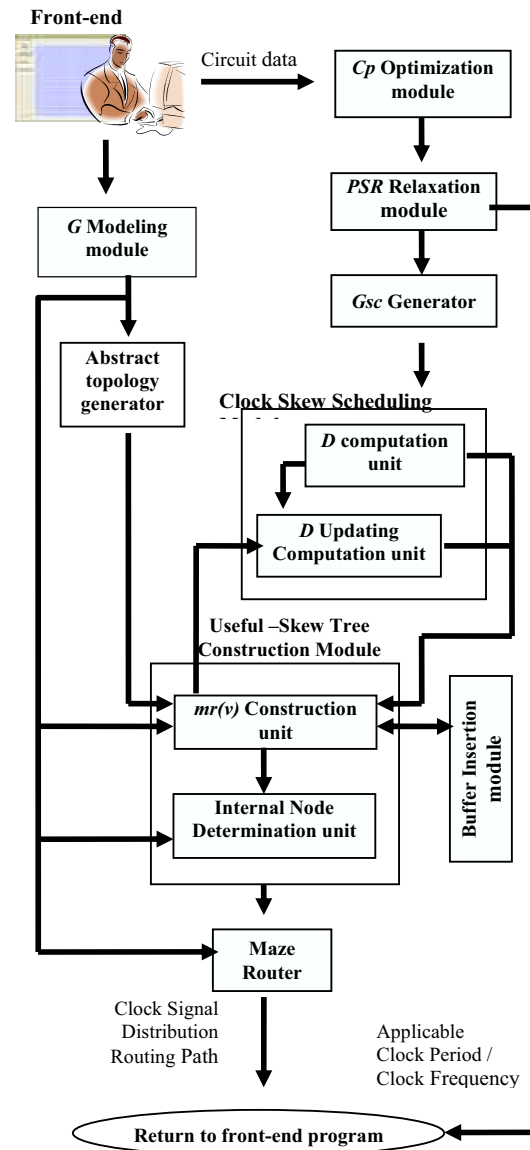
Reference	
Circuit data	The data of propagation delay of FF, combinational logic delay, and location of clock source / sinks.
Cp	Clock period / system frequency
PSR	Permissible Skew Range
Gsc	Skew Constraint Graph
D	All-pairs-shortest skew-constraint matrix
G	Graph Cartesian
Abstract topology	a binary tree such that all clock sinks are the leaf nodes of the binary tree.

Figure 1. Proposed Architecture of Clock Routing CAD Sub-system

The second category of works include H-tree method, which used regular systolic arrays [9], [10] and [11], Method of Means and Medians (MMM) algorithm proposed by [2], which generates a topology by recursively partitioning the set of sinks into two equal-sized subsets. [12] have proposed Geometric Matching Algorithm (GMA), a bottom-up matching approach to clock tree construction and [13] proposed the Weighted Center Algorithm (WCA). However, these methods focus primarily on path length balancing, rather than actual delay balancing.

Recently, the research work is focus on topology synthesis of the zero-skew tree (ZST) [14], and ZST routing based on the deferred-merge embedding (DME) algorithm [15], [16], [17]. [18], [19], [20], [21] extended the DME algorithm to consider bounded-skew tree (BST) routing. Finally, [22] further enhance the BST algorithm and proposed an incremental scheduling technique to compute the feasible-skew range for constructing a useful-skew tree (UST) for general skew constraints.

3. Useful-Skew Tree (Ust) Algorithm Based on Deferred-Merge Embedding (Dme) Paradigm

The UST / DME routing technique consists of two main functions, which are Clock Skew Scheduling and Useful–Skew Tree Construction. (Figure 1) The Clock Skew Scheduling module consists of two computation unit, which is include the all-pairs-shortest skew-constraint matrix (D) computation unit and all-pairs-shortest skew-constraint matrix (D) updating computation unit. The all-pairs-shortest skew-constraint matrix (D) computation unit applies the Floyd-Warshall algorithm to compute the bounded limit of skew constraints for each sinks node (Figure 2). The all-pairs-shortest skew-constraint matrix (D) updating computation unit applies the incremental scheduling technique that proposed by [22] to narrow a nontrivial feasible skew constraint to a single skew value (Figure 3).

The Useful-Skew Tree Construction module also consists of two computation units. The first unit is merging region ($mr(v)$) construction unit, to perform the construction of a binary tree of merging region that represent the loci of possible embedding points of internal nodes in a bottom-up order(Figure 4). Secondly is internal node determination unit, which determines the exact location of internal nodes in a top-down order (Figure 5).

Input: original skew constraints matrix (W)
Output: All-pairs-shortest skew-constraints matrix (D)
1. $n \leftarrow$ rows[W]
2. $D^{(0)} \leftarrow W$
3.   **for** $k \leftarrow 1$ **to** n
4.   **do for** $i \leftarrow 1$ **to** n
5.   **do for** $j \leftarrow 1$ **to** n
6.   **do** $d^{(k)}_{ij} \leftarrow \min(d^{(k-1)}_{ij}, d^{(k-1)}_{ik} + d^{(k-1)}_{kj})$
7.   **return** $D^{(n)}$

Figure 2. Floyd-Warshall Algorithm

Input: skew commit $skew_{i,j} = x$,
all-pairs shortest distance matrix, $D = \{d_{i,i}\}$
Output: an updated matrix D
1.   **Set** $d_{i,j} = -x$
2.   **Set** $d_{j,i} = x$
3.   **for** each $d_{k,l}$, $1 \leq k \neq l \leq n$ in D
4.   **Set** $d_{k,l} = \min\{d_{k,l}, d_{k,i} - x + d_{j,l}, d_{k,j} + x + d_{i,l}\}$
5.   **return** matrix, D

Figure 3. Incremental Scheduling Algorithm

Input: clock pins / joining segment (Lu, Lw), local skew constraints
Output: merging region ($mr(v)$) or no solution
1.   Determine boundary of shortest-distance region (SDR) among Lu, Lw
2.   **for** $n = Lu$ to Lw along the boundary of SDR
3.   delay (t_u) = Elmore Delay (n, Lu)
4.   delay (t_w) = Elmore Delay (n, Lw)
5.   clock skew ($Tskew$) = t_u - t_w
6.   **if** $Tskew \in FSR_{i,j}$
7.   **mark** (**each** n at same Manhattan arc **and** $\in SDR$) $\in$ merging region ($mr(v)$)
8.   **return** $mr(v)$

Figure 4. Bottom-up Phase Merging Region Construction Algorithm

Input: Tree of merging region $Tmr(v)$
Output: Location of internal nodes
1.   **for** each internal node (v) in $Tmr(v)$
2.   **if** v is the root
3.   Choose any node ($l(v)$) $\in mr(v)$.
4.   **else**
5.   Let P be the parent node of v
6.   Let Q be the merging region of v's sibling node
7.   **if** $|e_v|$ not determined during merging region construction
8. $|e_v|$ = shortest distance of joining segment Q and node $l(p)$
9.   = $d(JS_Q(mr(v)), l(p))$
10.   **Choose** any $l(v) \in JS_Q(mr(v))$ closet to $l(p)$

Figure 5. Top-down Phase Internal Node Determination Algorithm

4. Implementation Of Ust/Dme Algorithm

The functional block diagram of proposed useful-skew tree construction module is provided in Figure 6. The proposed useful-skew tree construction module is targeted to be implemented in C++. The module will receive the data of abstract topology, location of clock source and clock sinks, and each local skew constraint as an input; to produce the useful-skew tree as an output that include the

location of each internal node and the linking relationship information to their child nodes.

Referring to the diagram, the proposed useful-skew tree construction module consists of three main design components, which is briefly described as follows:

- **Computation units**: These computation units include the *D* computation unit, *D* Updating Computation unit, *mr(v)* Construction unit, Joining Segment Determination unit, delay modeling unit and Internal Node Determination unit. The computation units will perform the algorithms described in the previous section. The bottom-up phase merging region construction algorithm (Figure 4) is simultaneously performed by *mr(v)* construction unit, joining segment determination unit and delay modeling unit.

- **Control units**: The control unit handles the data transfer (in/out) among the computation units, and data transfer to/from routing database. The control unit also controls the whole processing flow during the useful-skew tree construction.

- **Routing database:** The database is used to store the information such as location of each clock sinks, location of internal nodes, joining segments, all merging regions, linking relationship information and some necessary identifiers that will be used for trace back process.
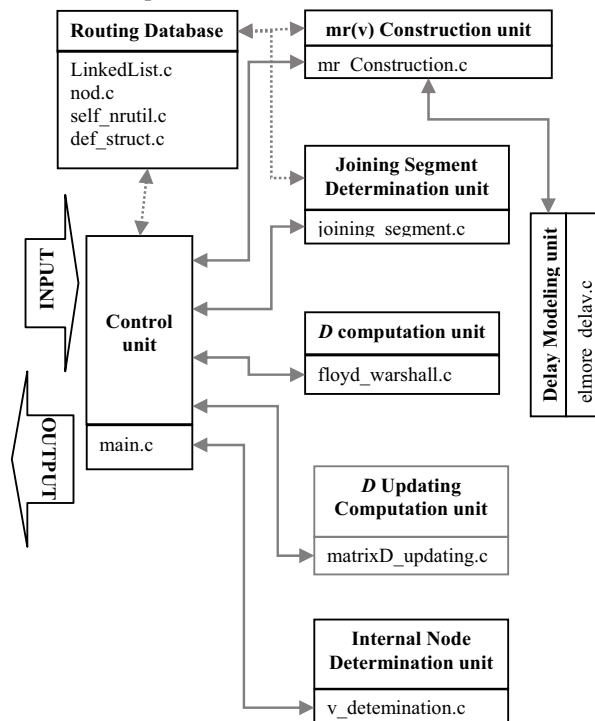

Figure 6. Functional Block Diagram of proposed useful-skew tree construction module

5. Conclusion

This paper has proposed the design of useful-skew tree construction module to synthesize the useful-skew tree routing path that implemented UST/DME algorithm. The advantage of this proposed architecture is that the clock layout is constructed and simultaneously determines the skew schedule, such that the resulting skew schedule is not only feasible, but also best for routing in term of wire-length.

References

[1] H. Bakoglu, *Circuits, Interconnections and Packaging for VLSI*. Reading, MA: Addison-Wesley, 1990.

[2] M. A. B. Jackson, A. Srinivasan, and E. S. Kuh, *Clock routing for high performance IC's*, In Proc. ACM / IEEE Desing Automation Conference, pp. 573-579, 1990.

[3] J. P. Fishburn, *Clock skew optimization*. IEEE Trans. on Comput. 39, 7 (July), 945–951. 1990.

[4] K. A. Sakallah, T. N. Mudge, and O. A. Olukotun, *checkTc and minTc: Timing verification and optimal clocking of synchronous digital circuits.* In Proc. International Conference on Computer Aided Design, 552–555. 1990.

[5] T. Szymanski, *Computing optimal clock schedules*. In Proc. Design Automation Conference, 399–404. 1992.

[6] R. B. Deokar, and S. S. Sapatnekar, *A graph-theoretic approach to clock skew optimization*. In Proc. IEEE International Symposium on Circuits and Systems, 407–410. 1994

[7] S. S. Sapatnekar, and R. B. Deokar, Utilizing the retiming-skew equivalence in practical algorithm for retiming large circuit. IEEE Trans. Computer Aided Design Integrated Circuit System 15, 1237-1248. 1996.

[8] J. L. Neves, and E. G. Friedman, *Optimal clock skew scheduling tolerant to process variations*. In Proc. Design Automation Conference, 623–628. 1996.

[9] H. Bakoglu, J.T. Walker, and J.D. Meindl, *A symmetric clock-distribution tree and optimal high-speed interconnection for reduced clock skew in ULSI and WSI circuit*. In Proc. IEEE International Conference on Computer Design, pp. 118-122. 1986.

[10] D. Dhar, M.A Franklin, and D.F. Wann, *Reduction of clock delays in VLSI structure*. In Proc. IEEE International Conference on Computer Design, pp. 778-783. 1984.

[11] A.L. Fisher, and H.T. Kung, *Synchronizing large systolic arrays*. Proc. of SPIE, pp. 44-52. 1982.

[12] A. B. Kahng, J. Cong, and G. Robins, *Matching based Models for high performance clock routing*. IEEE Transaction on CAD of Integrated Circuit and System, 12:1157-1169. 1993.

[13] N.A. Sherwani, and B. Wu, *Clock layout of high performance circuits based on weighted center algorithm*. In Proc. Fourth IEEE International ASIC Conference and Exhibit, pages p15-5.1-5.4. 1991.

[14] R.S. Tsay, *An exact zero-skew clock routing algorithm*. IEEE Transaction Computer Aided Design Integrated Circuit System CAD-12, 242–249. 1993.

[15] M. Edahiro, *Minimum path-length equi-distant routing*. In Proc. IEEE Asia-Pacific Conference on Circuits and Systems (December), 41–46. 1992.

[16] T.H. Chao, Y.C. H. Hsu, J.M. Ho, K. D. Boese, and A. B. Kahng, *Zero skew clock routing with minimum wirelength*. IEEE Transaction Circuit System 39, 799–814. 1992.

[17] A. B. Kahng, and C. W. A. Tsao, *Planer-DME: A single –layer zero-skew clock tree router*. IEEE Transaction Computer Aided Design Integrated Circuit System 15, 8-19. 1996.

[18] J. Cong, and C.K. Koh, *Minimum-cost bounded-skew clock routing*. In Proc. IEEE International Symposium on Circuit and System, 1.215-1.218. 1995.

[19] J.H. Huang, A. B. Kahng, and C. W. A. Tsao, *On the bounded-skew routing tree problem*. In Proc. Design Automation Conference (June), 508-513. 1995.

[20] A. B. Kahng, and C.W. A. Tsao, *Practical bounded-skew clock routing*. J. VLSI Sig. Process. (Special Issue on High Performance Clock Distribution Networks) 16, 199–215. 1997.

[21] J. Cong, A. B. Kahng, C.K. Koh, and C.W. A. Tsao, *Bounded-skew clock and Steiner routing*. ACM Transaction Design Automation Electronic System 3, 3, 341–388. 1998.

[22] C. W. A. Tsao, and C. K. Koh, UST/DME: A clock tree router for General Skew Constraints. ACM Transactions on Design Automation of Electronic System, Vol. 7, No. 3. 2002.