Optimal Sizing of High Speed Clock Networks Based on Distributed RC and Lossy Transmission Line Models

Qing Zhu   Wayne W.M. Dai   Joe G. Xi

Board of Studies in Computer Engineering
University of California, Santa Cruz
Santa Cruz, CA 95064

ABSTRACT

We have proposed an efficient measure to reduce the clock skew by assigning the clock network with variable branch widths. This measure has long been used for "H" clock tree. This paper formulates the optimal sizing of a general clock network as an optimization problem which minimizes the clock skew in a feasible set of widths. This feasible set of branch widths is decided by the process technology and routing resources. The skew minimization problem is turned into a least-squares estimation problem, and a modified Gauss-Marquardt's method is then used to determine the optimal widths of clock branches. This optimization method combines the best features of the methods based on Taylor series and methods based on gradients. An efficient algorithm is also proposed that assigns the good initial widths especially for a clock tree which let the later optimization process converge much more quickly. Our method is very flexible and can handle the general clock network including loops. The clock network can exhibit distributed RC and lossy transmission line behaviors. The method employs a scattering-parameters based delay macromodel to evaluate the timing of the clock network during the optimization process. The major objective of our sizing method is to minimize the skew, but as a by-product that the largest path delay is also reduced.

Keywords: sizing, clock network, skew minimization, least-squares estimation, distributed RC line, lossy transmission line

Contents

List of Figures

List of Tables

1   Introduction

The general trend in digital VLSI circuits is certainly towards higher complexity and faster clock frequency. Further, as more circuitry is integrated into a single VLSI system and wider length words are processed in parallel, larger registers are more frequently employed. These registers require precise clock signals to synchronize local activity with the global signal. For optimal system performance, clock signals must reach each register at exactly (or almost exactly) the same time. Clock skew is due to the variations in the time of arrival of a clock signal at clocked elements in a digital system. This skew is caused by several facts: (1) different path lengths from the source to terminals; (2) different capacitances of terminals; (3) non-symmetrical topology of clock tree which adds the loading unbalances; (4) variations of physical parameters across the VLSI system; (5) transmission line effects in high speed circuits such as the reflection and cross talk. Clock skew has been identified as one of major limiting factors for high speed synchronous VLSI systems in sub-micron ICs and multi-chip modules.

The "H" clock tree is widely used in the IC industry [2]. However, it is only applicable for symmetric arrays of logic elements. Some research work has been done involving the construction of a clock tree based on the general distribution of clock terminals. Algorithms presented in [13, 14, 23] try to construct a clock tree with equal path length from the source to terminals. It has been proved that the algorithm in [23] guarantees a planar equal path length clock tree with minimum path length for arbitrary distribution of clock terminals. An improved algorithm in [21] considers *Elmore delay* balance instead of geometric length balance. Another algorithm in [8, 7] improves the Elmore delay matching method by considering the minimization of the total wire length. But this algorithm yields minimal wire length only for a given clock tree and based on linear delay model.

The Elmore delay model or simpler linear delay model used in the previous methods is not enough for high speed interconnects which exhibit distributed RC behaviors in IC chips or transmission line effects in multi-chip modules [2]. Furthermore, all the previous methods assume the clock network to be tree topology. But in the IC industry, clock networks with loops such as clock meshes are often used [19].

The clock network can be assigned with variable widths on wires to reduce the skew, which is also often used in the industry. By adjusting widths of clock wires, the electrical parameters such as resistance, capacitance and inductance associated with a clock wire will be changed. According to circuit theory, the voltage waveforms at the terminals of a clock network will be changed. Skew may be reduced if we update the wire widths in such a way that the waveforms at the clock terminals have less difference in rise/fall time. Symmetrical variable widths are also used in "H" clock tree to achieve the proper impedance matching to reduce the clock skew caused by transmission line phenomena. In the case of non-symmetrical clock tree, i.e. the general distribution of clock terminals with variable loading capacitances, people in the industry often adjust the clock wire widths manually based on estimations or Spice simulation results. This usually takes long time and lacks accurate relationship of timing and wire widths.

In this paper, we present an optimization method to perform automatic sizing of clock wires for a given clock network. The topology of clock network can include loops. The clock network can behave distributed RC and lossy transmission line properties, which occur in high speed large scale sub-micron ICs or thin-film multi-chip modules. This method combines internally a delay macromodel to evaluate the timing of the clock network during

2.2   Problem Formulation

Given the topology of a clock network, the sizing of a clock network is to assign feasible widths on branches to optimize the performance, which includes minimizing the clock skew and path delay from the clock source to terminals. A *feasible width* of a branch is bounded by the maximum and minimum allowable widths. The *maximum* allowable width of the branch is decided based on the routing resource in the layout, and the *minimum* allowable width is usually due to the process technology. The set of possible feasible widths for a branch is discrete, with a increment width Δ decided by the technology.

We define the following notations for the formulation of the sizing of a general clock network.

n: number of branches in the clock network;

m: number of clock terminals;

w_i: width of branch b_i;

l_i: length of branch b_i;

w_i^t: maximum allowable width of branch b_i;

w_i^b: minimum allowable width of branch b_i;

Δ_i: increment width of branch b_i;

W: column vector of widths of all branches, that is $W = \{w_1, w_2, \ldots, w_n\}^T$, where the superscript T denotes matrix transposition.

d_i: delay from the source to terminal t_i;

c_i: capacitance at terminal t_i;

d_f: least delay from source to terminals;

d_s: largest delay from source to terminals;

The optimal sizing for a general clock network can be formulated as an n dimensional optimization problem as follows. Let $f(w_1, w_2, \ldots, w_n) = d_s - d_f$,

Objectives

$$Min\ f(w_1, w_2, \ldots, w_n) \tag{2.1}$$

Constraints

$$w_1^b \leq w_1 \leq w_1^t, \quad w_1 \in \{w_1^b, w_1^b + \Delta_1, w_1^b + 2\Delta_1, \ldots, w_1^t\}$$

$$w_2^b \leq w_2 \leq w_2^t, \quad w_2 \in \{w_2^b, w_2^b + \Delta_2, w_2^b + 2\Delta_2, \ldots, w_2^t\} \tag{2.2}$$

$$\vdots$$

$$w_n^b \leq w_n \leq w_n^t, \quad w_n \in \{w_n^b, w_n^b + \Delta_n, w_n^b + 2\Delta_n, \ldots, w_n^t\}$$

It is indicated by (2.2) that the branch widths should be bounded and discrete based on some increments. For examples, for a IC chip with $1\mu m$ technology, branch widths are allowable between $1\mu m \sim 10\mu m$ and the increment is $0.5\mu m$. For a substrate of a multi-chip module, branch widths are bounded between $10\mu m \sim 50\mu m$ and the increment is $1\mu m$. The objective of clock network sizing is to further minimize the clock skew shown in (2.1), while all branch widths satisfy the constraints shown in (2.2).

Suppose that each branch has the same number of k feasible widths to be selected, the *exhaustive enumeration method* takes $O(k^n)$ times to iterate all possible combinations of feasible widths of all branches to achieve the global optimization of the objective shown in (2.1). However, this method is not feasible since it has exponential time complexity. We have to seek other efficient optimization method to solve this problem.

3   Clock Network Modeling and Interconnect Delay Model

The optimal sizing of clock network , which is formulated in (2.1) and (2.2), depends on the calculation of the clock skew or delay along the network. The accuracy of the delay calculation, however, heavily depends on the modeling of the clock network and the adapted delay model. If the interconnects are sufficiently long or the circuits are sufficiently fast such that rise times of the waveforms are comparable to the time of flight across the line (determined by the speed of light), for example the wires on multichip substrates, the *inductance* also becomes significant, and the wires must be modeled as *transmission line*. The clock network usually crosses the whole VLSI system with long wires and large loads. With the increase of chip size and especially for the substrate of multi-chip modules, the clock network will be modeled as transmission line when the clock frequency increases [2].

To minimize the clock skew, the sizing should be performed on a good topology of clock network, since the topology decides the branch lengths and the network structure. Some of the previous work can be found in [2, 13, 14, 21, 7, 23].

3.1   Clock Network Topology

An algorithm was proposed in [23] to construct a *planar clock tree*. It has been proved [24] that the algorithm guarantees a planar equal path length clock tree rooted directly at the source, such that the path length from the source to terminals is minimized. A planar clock tree may be implemented on a single metal layer. Since it is easier to achieve uniform electrical parameters on a single layer than with multiple layers, it is easier to adjust a planar clock tree for zero skew and minimal path delay. Planar clock routing becomes feasible when three or more layers are available. Also equal path-length is the first order indication of balanced path delay from the source to every clock terminal. So, such a planar clock tree with equal path length provides a good clock topology for designers to optimize the branch widths for minimizing the clock skew while maintaining the planarity of the clock network. In Section 6 of experimental results, there are two planar equal-path length clock trees are tested which are generated by the clock routing algorithm [23]. However, the sizing method presented in this paper can be applied on a general clock network even with loops as shown in Figure 2.1(a).

3.2   Clock Network Modeling

Lumped RC circuit models[21, 7] become inaccurate when the clock frequency increases and the wire length is enlonged in multi-chip modules or extra-large IC chips, and the interconnect really exhibits distributed RC behaviors or even transmission line effects. The coupling noise on the clock network also becomes serious in high speed circuits. However, in current thin-film multi-chip module or large scale IC chip, the resistive effect still dominates the inductive phenomena such that the interconnection functions as a *lossy transmission*

line. So, we use two types of modeling for a branch of the clock network on chip and the substrate of thin-film multi-chip module.

- a distributed RC line, when inductive effect is ignored.

- a distributed RLC line, when inductive effect is considered.

Given a branch with width w_i, we get the resistance R and capacitance C of a unit length line [1] as

$$R = \gamma/w_i, \qquad C = \beta w_i \tag{3.1}$$

where γ is the *wire sheet resistance* and β is the *wire unit area capacitance*. For the lossy RLC transmission line model of a branch, the inductance L of a unit length line is obtained based on the semiconductor substrate [22], which can be expressed as

$$L = \frac{\mu}{2\pi} ln(\frac{8h}{w_i} + \frac{w_i}{4h}) \tag{3.2}$$

where μ is the magnetic permeability of the insulator, and h is the interval between two adjacent layers. If we enlarge the width of a branch, based on (3.1) and (3.2), the resistance and inductance are decreased but the capacitance is increased.

We size the whole clock network by modeling the branches as distributed or lossy RLC line. Figure 2.1(b) shows the modeling of a clock tree, where we assume this planar tree is implemented on the same layer without vias. Each branch is modeled as distributed RC line or lossy RLC transmission line. Note that terminals can have variable loading capacitances.

3.3 S-Parameter Based Interconnect Delay Model

The Elmore delay model [20, 21] is usually used for a *RC tree*. It is not suitable for the interconnect network which is based on lossy transmission line model and has a general topology including loops. We use a generalized delay macromodel [15, 16] which is based on *scattering-parameters (S-parameter)*. S-parameter is easier to measure and to work with at higher frequencies compared to using other type of parameters. It also provides a very convenient means for describing distributed RC line or lossy transmission line.

A RLC line can be viewed as a two port element based on the ground plane [15]. The S-parameter of a RLC line [10] can be expressed as

$$S = \frac{1}{2Z_0 Z_c cosh(\theta) + (Z_c{}^2 + Z_0{}^2)sinh(\theta)} \times \begin{pmatrix} (Z_c{}^2 - Z_0{}^2)sinh(\theta) & 2Z_c Z_0 \\ 2Z_c Z_0 & (Z_c{}^2 - Z_0{}^2)sinh(\theta) \end{pmatrix} \tag{3.3}$$

where $Z_c = \sqrt{(R + sL)/(sL)}$ is the characteristic impedance, $\theta = \sqrt{(R + sL)(sC)}l$ is the propagation constant. l is length of the line, and R, L and C are resistance, inductance and capacitance of the unit length line. Z_0 is an arbitrary reference impedance.

A RC line can also be viewed as a two port element based on the ground plane. The S-parameter of a RC line can also be expressed as in (3.3), where $Z_c = \sqrt{R/(sC)}$ and $\theta = \sqrt{sRC}l$.

The S-parameter based interconnect delay model can handle the general network including capacitive cutsets, arbitrary loops and lossy transmission lines. It employs an efficient network reduction algorithm to reduce the original network into a network containing one multiport component together with the source and terminals. Then *Pade approximation* [16] or *Exponentially Decayed Polynomial Function approximation* [15] is used to derive the macromodel. This delay macromodel is very flexible that the trade-off of accuracy and running time can be controlled by adjusting the order of approximation. At different design stages, the simulation accuracy and speed are combined to determine the level of the macromodel needed.

4   Optimization Method

The optimal sizing problem for a general clock network formulated in (2.1) is a n-dimensional optimization problem. The constraints shown in (2.2) constitute a *feasible set* where the optimum solution is located. We turn the clock skew minimization problem into a *least-squares estimation* problem. We use an efficient optimization method [17] to achieve the optimal widths of clock branches. This method combines the best features of the methods based on Taylor series and methods based on gradients.

Let d_f be the least delay among all delays from the source to terminals. For each terminal t_i ($1 \leq i \leq m$) with delay d_i, define the error $g_i = d_i - d_f$. The major objective of clock network sizing is to minimize the skew, however as a by-product, the path delay will also be decreased. Minimizing g_i means approaching d_i to d_f such that both skew and path delay are reduced. Let a column vector $G = (g_1, g_2, \cdots, g_m)^T$, and[1] we have the summation of all squares of errors.

$$\Phi(w_1, w_2, \cdots, w_n) = G^T G = \sum_{i=1}^{m} g_i{}^2 = \sum_{i=1}^{m} (d_i - d_f)^2. \qquad (4.1)$$

We thus get the *root-mean-square (rms) error* as

$$\delta = \sqrt{\Phi/m} = \sqrt{\sum_{i=1}^{m} g_i{}^2/m} \qquad (4.2)$$

The clock skew $f(w_1, w_2, \cdots, w_n) = d_s - d_f$. We have the next theorem to show the consistency of minimizing the skew and minimizing the *rms* error when the optimization proceeds.

Theorem 1: *Given a clock network, the root-mean-square error defined in (4.2) and the clock skew are linearly bounded each other.*

Proof: Since $d_i \leq d_s$ for all $1 \leq i \leq m$, where m is the number of terminals and it is a constant for a given clock network, based on (4.2) and (4.1), we have $\delta = \sqrt{\sum_{i=1}^{m} (d_i - d_f)^2/m} \leq \sqrt{\sum_{i=1}^{m} (d_s - d_f)^2/m} \leq f(w_1, w_2, \cdots, w_n)$. On the other hand, we have $f(w_1, w_2, \cdots, w_n) \leq \sqrt{m \sum_{i=1}^{m} (d_i - d_f)^2/m} \leq \sqrt{m}\delta$.   □

---

[1]The superscript T denotes matrix transposition.

Most algorithms for the least-square estimation of nonlinear parameters have centered about either of two methods [17]. In one method, the objective model is expanded as a *Taylor series* and corrections to parameters which are calculated at each iteration based on the assumption of local linearity. The other method is based on the modifications of the *steepest-descent method*. Both of these two methods have their advantages depending on the applications. While the methods based on Taylor series suffer from the possible divergence of the successive iterates, the steepest-descent based methods may have a very slow convergence of the optimum solution after the first few rapid iterations. The Gauss-Marquardt's method [17] tries to perform an optimum interpolation between the Taylor series and the gradient based method. This method is based upon the maximum neighborhood in which the truncated Taylor series gives an adequate representation of the nonlinear objective model. Some properties proved in [17] show that this method combines the best features of the previous two methods but hopefully avoids their limitations. This method shares with the gradient methods the ability to converge from an initial guess quickly, and the ability of Taylor series method to reach the converged values rapidly after the vicinity of the converged values has been reached.

We apply the Gauss-Marquardt's method to solve the optimization problem stated in (2.1) and (2.2), since Theorem 1 shows the consistency of minimizing the skew and the *rms* error for a given clock network. Set the width vector $W = \{w_1, w_2, \cdots, w_n\}^T$. Starting from a initial solution of $W^{(0)}$, based on the Gauss-Marquardt's method, W is optimized according to the following iteration formula:

$$W^{(k+1)} = W^{(k)} - (J^T J + \lambda \Lambda)^{-1} J^T G|_{W^{(k)}} \qquad (4.3)$$

where the superscript k is the iteration count, and $G|_{W^{(k)}}$ is the column vector of errors for all terminals at the kth iteration. J is a $m \times n$ sensitivity matrix with the $(i,j)th$ element $J(i,j) = \partial g_i / \partial w_j$. Λ is a diagonal matrix in which the values of its diagonal elements are the same as diagonal elements of $J^T J$. λ is the *Lagrange multiplier* properly selected for speeding up the convergence of the iteration process.

The *Lagrange multiplier* λ is dynamically adjusted at each iteration. The key feature of this optimization method is to search the feasible λ such that the objective function is always decreased at each iteration until the convergency of global optimum is reached. It has some *up-hill* property similar to simulated annealing by selecting proper λ to avoid the local optimum. We use the strategy in [17] of selecting λ, and allow the branch widths always constrainted between the lower and upper bounds as shown in (2.2). At each iteration, we fit the resultant width of each branch into the nearest discrete slot which is apart an increment Δ from adjacent ones, such that the constraints shown in (2.2) are satisfied.

To obtain the sensitivity matrix J, one needs to get the sensitivity of error g_i at each terminal t_i with respect to all branch widths. These sensitivities are computed by numerical differentiation method [18]. Numerical differentiation is applicable to a complex clock network which may have loops and is based on distributed RC and transmission line models.

The iteration continues until the skew is less than a prescribed value, or required iterations exceed. The skew is usually accepted if it is under five percent of the clock period. Note that at each iteration the clock skew is monotonically decreased. The convergency to optimum values of Gauss-Marquardt's method is proved in [17].

5   Initial Widths Determination

5.1   Overview

The optimization procedure starts from an initial branch widths $W^{(0)}$. A good starting point of $W^{(0)}$ will make the iterations shown in (4.3) converge to the optimum solution quickly. A good initial $W^{(0)}$ should be in the feasible set and close to the optimum solution.

We propose an efficient algorithm to determine the initial branch widths specially for a clock tree topology modeled as distributed RC lines. As we pointed out, the exhaustive enumeration method will take exponential times of the number of branches. This algorithm iterates the clock tree in linear time of the number of branches, where an branch is sized each time with the best *feasible width* which results in the smallest clock skew. A feasible width is located into possible width slots for this branch shown in (2.2). The order of sizing branches is from the source to terminals which tends to decrease both the skew and the largest path delay from the clock source to terminals. The S-parameter based delay model is used to evaluate the timing or clock skews.

5.2   Algorithm

We start on a clock tree with all branches assigned normal uniform widths. Our algorithm assigns one branch at a time with the best *feasible width* that achieves the smallest clock skew. We have the first rule for the clock tree sizing.

Rule 1: *At each stage, always assign a branch with the feasible width that results in the smallest skew.*

A reasonable sizing order of branches in the clock tree is necessary to achieve a good result. As shown in Figure 5.1, the node v_0 in a clock tree is supposed to have three children v_1, v_2 and v_3 and one parent v_4 which has shorter path to the source. Branch $\overline{v_4 v_0}$ is called the *parent branch* of the three branches $\overline{v_0 v_1}$, $\overline{v_0 v_2}$ and $\overline{v_0 v_3}$, and these three branches are called *children branches* of $\overline{v_4 v_0}$. The clock tree is partitioned into four subtrees shown in Figure 5.1. A set of terminals is connected in each of Subtree 1, Subtree 2 and Subtree 3. Assume that at the current stage the slowest terminal with the largest path delay d_s is in Subtree 1, say terminal $t_1{}^1$. We determine which branch of $\overline{v_0 v_1}$, $\overline{v_0 v_2}$ and $\overline{v_0 v_3}$ is enlarged from the initial width. If we enlarge the width of branch $\overline{v_0 v_1}$, the resistance of $\overline{v_0 v_1}$ is decreased but the capacitance of $\overline{v_0 v_1}$ is increased. The decreased resistance of branch $\overline{v_0 v_1}$ tends to decrease the delays from v_0 to all descendant terminals in Subtree 1 including the slowest terminal. On the other hand, the increased capacitance of $\overline{v_0 v_1}$ tends to increase the delay from the source to v_0, such that the delays from source to descendant terminals in other subtrees Subtree 2 and Subtree 3 tend to be increased. The path delay from source to descendant terminals in Subtree 1 is the sum of delays from source to v_0 and from v_0 to these terminals. So, we assign a proper width on $\overline{v_0 v_1}$ which results in the smallest clock skew in such a way that increases the path delays of faster terminals in Subtree 2 and Subtree 3 and may decreases the path delays of slower terminals in Subtree 1. As the result, the clock skew is reduced, and at the same time the largest path delay may also be decreased. The above observation is well verified in experiments, and matches the expectation based on the commonly used delay models [20] for RC trees. This effect on path delays by adjusting the width of a branch, is applicable to a RC clock tree with arbitrary node degree.

If we consider enlarging the width of $\overline{v_0 v_2}$ (or $\overline{v_0 v_3}$) in the clock tree shown in Figure 5.1, when the slowest terminal t_1^1 is in Subtree 1, based on the previous analysis, the increased

explored to be assigned with the best feasible width according to Rule 1. The search wave frontier is then expanded to the children branches of this branch, such that all these children branches are inserted into the priority queue. Note that the slowest terminal is changed during this course. This process continues until the parent branch of the slowest terminal have been explored, since according to Rule 2, the further sizing of remaining branches has no help in reducing skew.

Our algorithm is highlighted in the following pseudo-code.

Clock tree initial sizing algorithm

Input: a clock tree T modeled as distributed RC lines with a source and a set of terminals;

Output: a clock tree with assigned branch widths;

Procedure **ClockTreeInitialSizing**(T) {

    Q = set of branches connected to the source;

    t_s = the slowest terminal of the clock tree;

    b = ancestor branch of t_s extracted from Q;

    s = skew of the clock tree;

    while ($b \neq NULL$) {

        Assign b the feasible width which results in the smallest s;

        for (each children branch b_i of b)

            Insert b_i into Q;

        Update t_s;

        b = ancestor branch of t_s extracted from Q;

        }

    }

5.3   Analysis

Suppose that there are n branches in the clock tree. The algorithm iterates branches of the clock tree n times in the worst case. But it usually takes fewer iterations because the algorithm stops if the parent branch of the slowest terminal has been explored. The priority Q can be implemented as a binary heap, and the operation on the queue takes $O(\log n)$. Most computational time is used on the evaluation of delays.

The next theorem shows the convergence of skew reduction during the course of initial clock sizing.

Theorem 2: *The clock skew is monotonically decreased as the initial clock tree sizing proceeds.*

Proof: First the clock tree has a skew when all branches are set the normal uniform width. Every time when a branch is sized, according to Rule 1, the new width assigned on this branch will result in the smallest clock skew at current stage. This least skew should not be larger than the skew when the original width is assigned on this branch. It follows that the skew is monotonically decreased as clock tree sizing proceeds.   □

(a)   (b)

Figure 6.1: (a) a planar equal path length clock tree with 18 terminals, s is the clock source. (b) a planar equal path length clock tree with 54 terminals.

Examples	Maximum Path Length		Longest Branch Length		Shortest Branch Length	
	Chip (mm)	MCM (mm)	Chip (mm)	MCM (mm)	Chip (mm)	MCM (mm)
Example 1	13.5	135	5.0	50.0	0.5	5.0
Example 2	6.0	60.0	2.7	27.0	1.0	10.0
Example 3	5.6	56.0	1.6	15.9	0.12	1.2

Table 6.1: Three sets of clock network examples, each of which is implemented in two cases: (1) in a $1\mu m$ IC chip based on distributed RC line model; (2) in a thin film multi-chip module based on lossy transmission line model.

We apply the optimal sizing method on these three sets of clock networks. The clock network is assumed to be routed on the same layer of metal. For the case in the IC chips, we use $1\mu m$ technology where the branch widths are bounded between $1\mu m \sim 10\mu m$ and the increment is $0.5\mu m$. For the case in the thin-film multi-chip modules, branch widths are bounded between $10\mu m \sim 50\mu m$ and the increment is $1\mu m$.

Table 6.2 shows the typical electrical parameters of a $1\mu m$ technology IC chip and a advanced thin-film multi-chip module. The unit length wire resistance, capacitance and inductance are obtained based on the wire width in $1\mu m$ for the IC chip and the wire width in $10\mu m$ for the MCM. Based on these typical values and formulae shown in (3.1) and (3.2) in Section 3.2, we can derive the modeling of a clock branch with a given length and width. In (3.2), h is taken $15\mu m$ based on a advanced MCM design [12]. Note that terminals in IC chip examples have two different loading capacitances.

Figure 6.2 shows the plots of skew improvement when our optimal sizing method is applied on the Example 1 which has loops and Example 3. Example 2 is not illustrated here and it shows a similar skew improvement to Example 3. Initial skew is obtained when all branches of the clock network are assigned with the normal uniform width. The skew

(a)   (b)

Figure 6.4: Simulation waveforms at terminals of clock trees of Example 3 after using our sizing method. (a) in IC chip; (b) in MCM.

Examples	Clock Skew (*ns*)			Largest Path Delay (*ns*)		
	ORIGINAL	DME	OSM	ORIGINAL	DME	OSM
Example 1	0.257	–	0.052	2.314	–	2.563
Example 2	0.098	0.131	0.004	1.941	2.454	1.874
Example 3	0.268	0.234	0.026	5.215	6.445	2.563
Example 4	1.600	1.821	0.020	19.15	25.48	18.70

Table 6.4: Comparison results on testing examples in the case of $1\mu m$ IC chips by using the Deferred-Merge Embedding (DME) algorithm [6] which employs π-RC model and Elmore delay, and our Optimal Sizing Method (OSM). The performance results are also shown for the ORIGINAL clock network with the normal uniform width on which DME algorithm and OSM method are applied. Example 4 is the planar clock tree with 54 terminals shown in Figure 4(b).

7   Concluding Remarks

We have proposed an efficient measure to reduce the clock skew by assigning the clock network with variable branch widths. This measure has long been used for "H" clock tree. This paper formulates the optimal sizing of a general clock network as an optimization problem which minimizes the clock skew with a feasible set of widths. This feasible set of widths is decided by the process technology and routing resources. We presented an approach to determine the optimal widths of clock branches. An efficient algorithm is also proposed that assigns good initial widths on a clock tree which allow the later optimization process to converge faster.

Our approach is very flexible and can handle the general clock network including loops. The optimization method is independent of the modeling of the clock network. Currently,

we concentrate on the high speed clock network modeled as the distributed RC lines in chips and the lossy transmission lines in the thin-film multi-chip modules.

The experimental results show that our approach reduces both the clock skew and path delays in most of cases. These performance results are achieved based on our error function for optimization and an efficient algorithm to assign initial widths for the clock tree.

Although the modeling and testing examples in this paper are based on the single layer implementation of the clock network, our approach can easily be applied to a clock network implemented on different layers. In this case, the modeling of the clock network should take into consideration of the effect of vias and variable parameters of different layers.

The current density requirement due to *electromigration* sometimes also should be taken into consideration for a large size clock network [3]. The required current density provides a lower bound for the width of each branch. This lower bound of the branch can be specified in the constraints when doing the optimization.

8 Acknowledgement

References

[1] H. Bakoglu. *Circuits, Interconnections, and Packaging for VLSI.* Addison-Wesley Publishing Company, 1987.

[2] H. Bakoglu, J. T. Walker, and J. D. Meindl. A symmetric clock-distribution tree and optimized high-speed interconnections for reduced clock skew in ulsi and wsi circuits. In *Proc. IEEE Intl. Conf. on Computer Design*, pages 118–122, 1986.

[3] S. Boon, S. Butler, R. Byrne, B. Setering, M. Casalanda, and A. Scherf. High performance clock distribution for cmos asics. In *IEEE Custom Integrated Circuits Conference*, pages 15.4.1–15.4.5, 1989.

[4] R. K. Brayton, G. D. Hachtel, and A. L. Sangiovanni-Vincentelli. A survey of optimization techniques for integrated circuits. *Proc. of the IEEE*, 69(10):1334–1362, 1991.

[5] D. L. Carter and D. F. Guise. Effects of interconnections on submicron chip performance. *VLSI Design*, pages 63–68, Jan. 1984.

[6] F.Y. Chang. Waveform relaxiation analysis of nonuniform lossy transmission lines characterized with frequency-dependent parameters. *IEEE Trans. on Circuits and Systems*, 38:1484–1500, 1991.

[7] T.H. Chao, Y.C. Hsu, and J.M.Ho. Zero skew clock net routing. In *Proc. of 29th Design Automation Conf.*, pages 518–523, 1991.

[8] T.H. Chao, Y.C. Hsu, J.M.Ho, Kenneth D. Boese, and Andraw B. Kahng. Zero skew clock net routing. *IEEE Transactions on Circuits and Systems*, 39(11):799–814, November 1992.

[9] Jason Cong, Kwok-Shing Leung, and Dian Zhou. Performance-driven interconnect design based on distributed rc delay model. In *Technical Report, University of California, Los Angeles*, pages 1–36, 1992.

[10] J. Dobrowolski. *Introduction to Computer Methods for Microwave Circuit Analysis and Design.* Artech House, 1991.

[11] J. P. Fishburn. Clock skew optimization. *IEEE Transactions on Computers*, 39(7):945–951, 1990.

[12] Robert C. Frye. Balancing performance and cost in cmos-based thin film multichip module. In *Proceedings of IEEE Multi-Chip Module Conference*, pages 6–11, 1993.

[13] M. A. B. Jackson, A. Srinivasan, and E. S. Kuh. Clock routing for high-performance ics. In *Proc. of 27th Design Automation Conf.*, pages 573–579, 1990.

[14] A. Kahng, J. Cong, and G. Robins. High-performance clock routing based on recursive geometric matching. In *Proc. of 28th Design Automation Conf.*, pages 322–327, 1991.

[15] H. Liao, W. Dai, R. Wang, and F.Y. Chang. S-parameter based macro model of distributed-lumped networks using exponentially decayed polynomial function. *to appear in Proceedings of 30th ACM/IEEE Design Automation Conference*, 1993.

[16] H. Liao, R. Wang, W. Dai, and R. Chandra. S-parameter based macro model of distributed-lumped networks using pade approximation. *to appear in Proceedings of International Symposium on Circuits and Systems*, 1993.

[17] D.W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math.*, 11:431–441, 1963.

[18] A. Ralston and P. Rabinowitz. *First Course in Numerical Analysis.* McGraw Hill Book Company, NewYork, 1978.

[19] Curtis Ratzlaff, Nanda Gopal, and Lawrence Pillage. Rice: Rapid interconnect circuit evaluator. In *Proceedings of 28th ACM/IEEE Design Automation Conference*, pages 555–560, 1991.

[20] J. Rubinstein, P. Penfield, and M. A. Horowitz. Signal delay in rc tree networks. *IEEE Trans. on Computer-Aided Design*, CAD-2(3):202–211, 1983.

[21] R.-S. Tsay. Exact zero skew. In *Digest of Tech. Papers of IEEE Intl. Conf. on Computer Aided Design*, pages 336–339, 1991.

[22] H.T. Yuan, Y.T. Lin, and S.Y. Chiang. Properties of interconnection on silicon, sapphire, and semi-insulating gallium arsenide substrates. *IEEE Transactions on Electron Devices*, ED-31:639–644, April 1982.

[23] Qing Zhu and Wayne W.M. Dai. Perfect-balance planar clock routing with minimal path-length. In *Digest of Tech. Papers of IEEE Intl. Conf. on Computer Aided Design*, pages 473–476, 1992.

[24] Qing Zhu and Wayne W.M. Dai. Perfect-balance planar clock routing with minimal path-length. *Technical Report, UCSC-CRL-92-12, University of California, Santa Cruz.*, 1992.