A Specified Delay Accomplishing Clock Router Using Multiple Layers

Mitsuho Seki*, Kenji Inoue*, Kazuo Kato†, Kouki Tsurusaki†,
Shin'ichi Fukasawa†, Hitoshi Sasaki+ and Mutsuhito Aizawa+

* Hitachi Research Laboratory, Hitachi Ltd., † Semiconductor Dept., Hitachi Ltd., + Hitachi Engineering Co.

Abstract Clock routing to minimize the clock skew is very necessary to make high performance LSIs. Our clock routing method: (1) realizes the specified delay to each input terminal and provides a zero skew; (2) uses multiple routing layers for pin-to-pin routing; and (3) considers the delay arising from the resistance of a through-hole. Experimental results show that the delay is within 1% error compared to the specified delay and the skew can be controlled within pico second order.

1. Introduction

Because the clock frequency is rapidly increasing for design of high performance VLSIs, a clock skew minimizing layout technique is very important to reduce the cycle time. In particular, clock routing is the most important consideration for getting precise skew adjustment. As clock skew reduction routing methods, routing schemes to get the equal length from an output terminal to each input terminal are popular([1]-[4]). But these methods consider only balance of the routing length.

Clock routers which balance the delay between two terminals while considering the capacitance and resistance of segments have recently appeared([5]-[8]). But these routers use only one layer for pin-to-pin routing. They do not consider the delay arising from through-holes. In this sense, conventional clock routers do not consider the routing delay precisely. Furthermore, the clock delay minimization without any regard to the delay in other parts of an LSI does not always contribute to making an actual high performance LSI, because multiple clocks are usually used and there are timing relations between data signals and clock signals. So, we propose a routing method which realizes a specified pin-to-pin delay given to each input terminal in a multi-terminal clock net.

Our clock router directly controls the pin-to-pin delay which is specified from the layout constraints. When this specified delay to each terminal is the same, the clock skew becomes 0. And we consider the delay arising from the resistance of a through-hole and deal with different capacitance and resistance parameters on different layers. Because the resist- ance of a through-hole near an output terminal influences the net delay very much, it is important to consider the resistance of a through-hole.

We assume that the routing pattern of a net consists of one trunk and plural branches. We allow the routes to bend arbitrarily and use multiple layers. By utilizing this routing pattern assumption, the delay conditions to accomplish the specified pin-to-pin delay are calculated, and then routing is executed so as to meet these conditions.

2. Assumptions and Delay Calculation

2.1 Assumptions

The clock distribution logic is made before placement and routing, and is appropriately partitioned into nets by inserting buffers as shown in Fig.2.1. The net is assumed to consist of one output terminal and plural input ones. There is output resistance on an output terminal and input capacitance on an input terminal. Our clock routing is executed net-by-net and is capable of tolerating the situation that the specified pin-to-pin delay is different among input terminals. Generally, the same delays are given in the case of clock nets.

2.2 Delay Calculation

We use [9] which is based on Elmore[10] as a net delay calculation. The capacitance and resistance model of one segment is a π model. Because we use multiple layers for routing, we take both segments and through-holes into consideration as the elements of a net when calculating the net delay. The capacitance and resistance of a segment are assumed to be calculated by multiplying the segment length with the capacitance and resistance per unit length, respectively. Regarding a through-hole, its resistance is considered, but its capacitance is neglected.

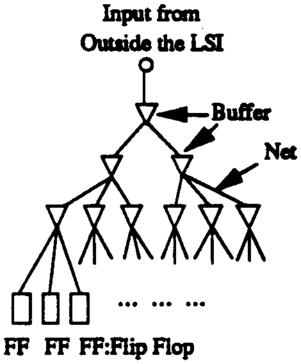
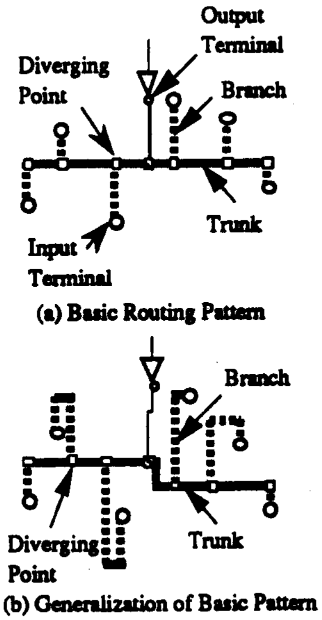


Fig.2.1 Clock Distribution Logic

3. Clock Routing Strategy

The delay cannot be calculated until the routing pattern is determined according to [9]. To realize the specified pin-to-pin delay, we think it is best to assume the basic routing pattern in advance, to derive conditions to accomplish the specified pin-to-pin delay using the basic pattern, and then to modify this routing pattern. In the basic pattern, a net has one trunk and plural branches to the terminals composing a net as shown in Fig.3.1 (a). And then, we produce a generalized pattern like Fig.3.1(b) by repeating U-shape modifications of the basic pattern.



(a) Basic Routing Pattern

(b) Generalization of Basic Pattern

Fig.3.1 Clock Routing Pattern

Permission to copy without fee all or part of this material is granted, provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

©1994 ACM 0-89791-690-5/94/0011/0289 $3.50

3.1 Trunk Routing

Given the terminals of one net, we search the locations of all the terminals and make the minimum rectangle which includes them. Then, the direction of the trunk is determined as a parallel line to the longer side of the rectangle. The trunk is basically located on the gravity point of the terminal distribution and is a straight line. But any trunk position satisfying the specified delay to each input terminal is permitted and bends are allowed if any prohibited areas are in the way.

After the trunk is drawn, the shortest route from the output terminal to the trunk is routed. If there are no obstacles, this route is a straight line, too. If this route is long, the net delay may be very large because the delay of this route is a product of the resistance of this route multiplied by the sum of all the capacitances of the net.

Next, we set diverging points on the trunk in order to route each branch from the trunk to each input terminal. The diverging point is usually the cross point made of the trunk and a perpendicular line from the input terminal. But, because of a routing rule or a prohibited area, these points are sometimes moved slightly.

3.2 The Delay of Every Part of the Net

We call a branch from the trunk to an input terminal an input branch. To accomplish the specified pin-to-pin delay, the precise condition to route an input branch should have to be sought. Then, we consider the branch length calculation model as in Fig.3.2. The basic routing pattern as shown in Fig.3.1(a) is assumed. A box in Fig.3.2 corresponds to a segment and the π model is assumed. Black circles show through-holes. White circles stand for diverging points on the trunk, and there is assumed to be a through-hole on it. Numbers are assigned to these diverging points. We let the diverging point to the output terminal be P. The extension of the box on the trunk from one segment to plural segments is easy. About the box on the input branch, its extension from one to plural segments is shown in the next chapter.

There are input branches and parts of the trunks on both sides of P in Fig.3.2. If we formulate the delay about one side, the other side is considered similarly. So, we consider the delay on the right side of P first.

We assume there are mr input branches on the right side of P. And we consider the delay of a right side input branch BR_j.

The input branch routing is assumed to be completed by: (i) putting a through-hole on the diverging point; (2) drawing the segment; and (3) putting a through-hole on the input terminal. We define notations as follows. The resistances of the through-holes are $rb1R_j$ and $rb2R_j$, respectively. Let the input capacitance, the length of an input branch, and the capacitance and resistance per unit length be cIR_j, XR_j, and cBR_j, rBR_j, respectively. Then, the delay DBR_j of the input branch BR_j and the entire capacitance $CABR_j$ of the right side input branch j are

$$DBR_j = cBR_j \cdot rBR_j \cdot XR_j^2/2 + (rBR_j \cdot cIR_j + cBR_j \cdot rb1R_j) \cdot XR_j$$
$$+ (rb1R_j + rb2R_j) \cdot cIR_j \qquad (3.1)$$

$$CABR_j = cBR_j \cdot XR_j + cIR_j \qquad (3.2)$$

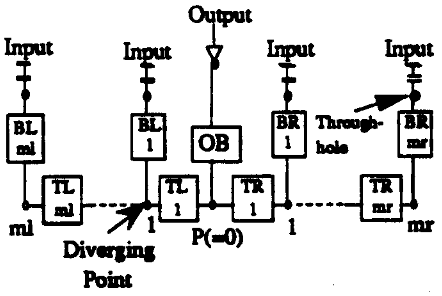


Fig.3.2 Branch Length Calculation Model

Next, we consider the delay of the box on the trunk in Fig. 3.2. The segment TR_j which is located between the diverging point j-1 and j is also assumed to be expressed by the π model, and its unit length capacitance and resistance are assumed to be cTR_j and rTR_j. The delay DTR_j of the segment TR_j is the product of the resistance of the segment TR_j multiplied by the summed capacitance of both segments at the input terminal side from the segment TR_j and the input capacitances We define the length of the segment TR_j as LTR_j. Then, DTR_j, the delay DQR_j from P to the diverging point j and the capacitance $CATR_j$ of the segment TR_j are,

$$DTR_j = LTR_j \cdot rTR_j \cdot (LTR_j \cdot cTR_j/2$$
$$+ \sum_{i=j+1}^{mr} LTR_i \cdot cTR_i + \sum_{i=j}^{mr} CABR_i) \qquad (3.3)$$

$$DQR_j = \sum_{i=1}^{j} DTR_i \qquad (3.4)$$

$$CATR_j = cTR_j \cdot LTR_j \qquad (3.5)$$

Finally, we consider the delay from the output terminal to P. When calculating this delay, because we have to consider the segments and through-holes on both sides of P, we define similar notations for the left side of P. We change R in the notations for the right side of P to L. And we assume there are ml branches on the left side of P. The route from the output terminal to the trunk, like an input branch on the right side of P, begins by putting a through-hole at the output terminal, draws the segment and finishes routing by putting a through-hole on the trunk.

We define the length of the route OB from the output terminal to the trunk as LOB, and the unit length capacitance and resistance as cLOB and rLOB, respectively. Because we assume we put a through-hole on the output terminal and P, the delay V1 derived from the output resistance r_0, the through-hole resistance r_{T1} on the output terminal and the through-hole resistance r_{T2} on P is expressed as follows.

$$CA = \sum_{j=1}^{mr}(CABR_j + CATR_j) + \sum_{j=1}^{ml}(CABL_j + CATL_j)$$
$$v1 = (r_0 + r_{T1})(cLOB \cdot LOB + CA) + r_{T2} \cdot CA \qquad (3.6)$$

Next, the delay V2 of the route OB itself is,

$$V2 = rLOB \cdot LOB \cdot \{ cLOB \cdot LOB/2 +$$
$$\sum_{j=1}^{mr}(CABR_j + CATR_j) + \sum_{j=1}^{ml}(CABL_j + CATL_j) \} \qquad (3.7)$$

If the route OB consists of plural segments, we only have to calculate the delay of each segment composing OB because the capacitances of the trunk and the input branches are already known. So, the delay DIR_j from the output terminal to the right side input terminal j and the delay DIL_j from the output terminal to the left side input terminal j are

$$DIR_j = V1 + V2 + DQR_j + DBR_j = MDR_j \ (j=1,\cdots,mr) \qquad (3.8)$$

$$DIL_j = V1 + V2 + DQL_j + DBL_j = MDL_j \ (j=1,\cdots,ml) \qquad (3.9)$$

$MDR_j(MDL_j)$ is the specified pin-to-pin delay from the output terminal to the right(left) side input terminal j.

3.3 Calculation of Branch Length

In this section, we show how to calculate each input branch length XR_j(right side), XL_j(left side). We define four vectors M, X, TB and Y as follows. M is the vector of the specified delay to each input terminal and X is the vector of each input branch length. Equations (3.8) and (3.9) are expressed as (3.16)

$$M=(MDR_{mr}, MDR_{mr-1}, \cdots, MDR_1, MDL_1, \cdots, MDL_{ml}) \qquad (3.10)$$

$$X=(XR_{mr}, XR_{mr-1}, \cdots, XR_1, XL_1, \cdots, XL_{ml}) \qquad (3.11)$$

$$TB=(TBR_{mr}/\log_e 2, TBR_{mr-1}/\log_e 2, \cdots, TBR_1/\log_e 2, TBL_1/\log_e 2, \cdots, TBL_{ml}/\log_e 2) \qquad (3.12)$$

$$TBR_j = D + \sum_{k=1}^{j}\{rTR_k \cdot cTR_k \cdot LTR_k^2/2 + rTR_k \cdot LTR_k \cdot (\sum_{i=k+1}^{mr} LTR_i \cdot cTR_i + \sum_{i=k}^{mr} cIR_i)\} + (rb1R_j + rb2R_j) \cdot cIR_j \qquad (3.13)$$

$$Y=(GR_{mr} \cdot XR^2_{mr}, \cdots, GR_e \cdot XR^2_e, \cdots, GL_{ml} \cdot XL^2_{ml})' \qquad (3.14)$$

$$GR_i = cTR_i \cdot rTR_i \quad i = 1, \cdots, mr \qquad (3.15)$$

$$M = A \cdot X + TB + (\log_e 2) \cdot Y/2 \qquad (3.16)$$

Details of the matrix A are omitted, but matrix A is a positive real matrix. So, the eigenvalue of A exists in the x>0 plane. In order to seek the solution in the x>0 plane, from [11], we only have to investigate the solution of the following equation.

$$M = A \cdot X + TB + (\log_e 2) \cdot F(X)/2 \qquad (3.17)$$

$$F(X) = (GR_{mr} \cdot XR_{mr} \cdot |XR_{mr}|, \cdots, GR_e \cdot XR_e \cdot |XR_e|, GL_e \cdot XL_e \cdot |XL_e|, \cdots GL_{ml} \cdot XL_{ml} \cdot |XL_{ml}|) \qquad (3.18)$$

This equation has at most one solution (derived from [11]) and the solutions in the x>0 plane in the two equations (3.16) and (3.17) are the same. We execute the input branch routing making use of this input branch length.

4. Input Branch Routing Method

In the derivation of the input branch length, we assumed that the input branch was a straight line. But, the actual routing uses plural segments and through-holes. In that case, we cannot accomplish the specified pin-to-pin delay if we simply use the necessary input branch length because the unit length capacitance and resistance are different in each layer and there exists delays arising from through-holes. So, we derive the necessary capacitance and delay conditions from the necessary input branch length, and then, we route the input branch so as to meet them.

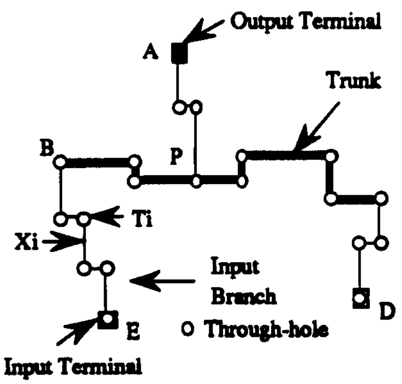


Fig.4.1 Branch Routing Model

4.1 The Conditions of the Branch Routing

The conditions for the input branch routing are explained by using Fig. 4.1. First, we consider the input branch delay, in other words, the delay between B and E. By comparing the case where the input branch is a straight line with the case where the input branch is routed with n segments, we derive the conditions of the input branch routing. When considering a certain input branch, we denote Z as the necessary input branch length, I_0 as the input capacitance, and R, C as the unit length resistance and capacitance, respectively. The straight line branch delay, B-E delay 1, is

$$\text{B-E delay } 1 = R \cdot Z \cdot (C \cdot Z/2 + I_0) \qquad (4.1)$$

The actual branch route is supposed to consist of n segments and n through-holes. We let T_i be the resistance of the through-hole i and X_i the length of the segment i composing the input branch. T_1 stands for the through-hole resistance on the trunk and T_{n+1} does on the input terminal. X_1 is the length of segment touching the trunk. c_i and r_i are the unit length capacitance and resistance of segment i, respectively. We give the n segments an input branch delay, B-E delay2, as

$$\text{B-E delay } 2 = \sum_{j=1}^{n+1} T_i \cdot (\sum_{j=i}^{n} c_j \cdot X_j + I_0) + \sum_{i=1}^{n} r_i \cdot X_i \ (c_i \cdot X_i/2 + \sum_{j=i+1}^{n} c_j \cdot X_j + I_0) \qquad (4.2)$$

Next, we consider the delay between A and B. When routing an input branch, the route between A and B is fixed. So, only the capacitance between the input branch B-E influences the delay between A and B. Then, B-E capacitance 1 (the straight line) and B-E capacitance 2 (n segments) are

$$\text{B-E capacitance } 1 = C \cdot Z \qquad (4.3)$$

$$\text{B-E capacitance } 2 = \sum_{i=1}^{n} c_i \cdot X_i \qquad (4.4)$$

We have to route the branch such that (4.1)=(4.2) and (4.3)=(4.4). These conditions are those of the delay and capacitance. These have to be considered separately, but that is difficult. So, we change the condition of the capacitance to that of the delay. We consider the sum of the resistance between A and B, UR.

$$UR = O_0 + \sum_{k=1}^{m} RF_k \cdot Y_k + \sum_{k=1}^{m} T_k \quad (4.5)$$

Here, we define O_0 as the output resistance, Y_k as the length of each segment length from A to B, RF_k as the unit length resistance of Y_k, and T_k as each through-hole resistance between A and B. Then, A-B delay 1 (straight line), A-B delay 2 (n segments) are

$$\text{A-B delay 1} = UR \cdot C \cdot Z + S \quad (4.6)$$

$$\text{A-B delay 2} = UR \cdot (\sum_{i=1}^{n} c_i \cdot X_i) + S \quad (4.7)$$

S is the delay that is determined without any relation to the input branch to be routed now.

Now, we define the delay between A and E as A-E delay. We let A-E delay 1 be the straight line A-E delay and A-E delay 2 be the actual route A-E delay. Then,

$$\text{A-E delay 1} = \text{A-Bdelay1} + \text{A-Edelay1} = (4.1) + (4.6) \quad (4.8)$$

$$\text{A-E delay 2} = \text{A-Bdelay2} + \text{A-Edelay2} = (4.2) + (4.7) \quad (4.9)$$

So, we route an input branch such that (4.1) + (4.6) = (4.2) + (4.7) is satisfied. We evaluate this condition when routing an input branch.

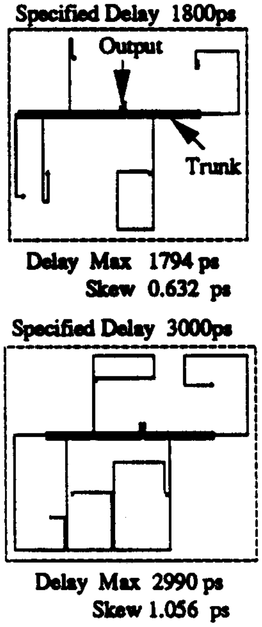
4.2 Input Branch Modification

We execute an initial input branch routing from the diverging point on the trunk to the input terminal as short as possible. Because (4.9) is known after the initial branch routing, we can evaluate the difference between (4.8) and the (4.9). If there is a difference, we modify the route of the input branch until (4.8)=(4.9) or the difference between (4.8) and the (4.9) becomes very small. The modification of the input branch route is done by U-shape extension or reduction against the initial input branch route. The search of the input branch route uses the line search method of [12].

5. Experimental Results

This algorithm was implemented in C language on the SUN SparcStation10.

Experimental results for one net are shown in Fig.5.1. The net used in the experiments consisted of 6 terminals (1 output and 5 inputs). We gave the same specified delay to all input terminals. The test delays were 3ns and 1.8ns, respectively. Routing was executed by using double metal layers. From the two routing patterns in Fig.5.1, we saw that the detour length of the input branches changed depending on the specified pin-to-pin delay. The delay after routing was 0.3% less than the specified pin-to-pin delay, but the difference was very slight. The skew was surprisingly small because it was around 1 ps and within 0.035% of the specified pin-to-pin delay. And the calculation time was about 0.42 s.



Specified Delay 1800ps

Delay Max 1794 ps
Skew 0.632 ps

Specified Delay 3000ps

Delay Max 2990 ps
Skew 1.056 ps

Fig.5.1 Routes for 1 net

In another experiment, the clock signal was distributed to 50 FFs and there were three consecutive nets until the clock signal reached the FFs. The specified pin-to-pin delays were 206ps to the 1st level net, 161ps to the 2nd level net and 74ps to the 3rd level net, respectively. So, the target clock path delay without gate delay was 441ps. The routing result was that the maximum clock path delay after routing was 439ps and the clock skew was 4.6ps. The delay was very close to the target delay and the clock skew was surprisingly slight.

6. Conclusion

We described the following clock routing strategy: (1) realizing a specified delay given to each input teminal; (2) using multiple layers to route; and (3) considering the delay derived from the resistance of a through-hole and dealing with different parameters on different routing layers. We presented the routing results that the realized delay was within a slight error compared to the specified delay and the routing skew could be controlled within pico second order. With this routing strategy, the clock skew is slight and multiple clocks behave correctly with the right phase difference.

References

[1]M.A.B. Jackson, et al., "Clock Routing for High-Performance ICs", 27th DAC, pp573-579, 1990

[2]A. Kahng, J. Cong & G. Robins, "High Performance Clock Routing Based on Recursive Geometric Matching", 28th DAC, pp322-327, 1991

[3]M. Edahiro, "A Clock Net Reassignment Algorithm Using Voronoi Diagram", ICCAD, pp420-423, 1990

[4]Q. Zhu et al, "Perfect-balance Planar Clock Routing with Minimal Path-length", ICCAD, pp473-476, 1992

[5]R.S. Tsay, "Exact Zero Skew", ICCAD, pp336-339, 1991

[6]T.H. Chao, Y.C. Hsu & J.M. Ho, "Zero Skew Clock Net Routing", 29th DAC, pp518-523, 1992

[7]S. Pullela, et al., "Reliable Non-Zero Skew Clock Trees Using Wire Width Optimization", 30th DAC, pp165-170, 1993

[8]W. Khan, et al., "Zero Skew Clock Routing in Multiple-Clock Synchronous Systems", ICCAD, pp464-467, 1992

[9]J. Rubinstein, P. Penfield & M.A. Horowitz, "Signal Delay in RC Tree Networks", IEEE Trans. on Computer-Aided Design, vol. CAD-2, No.3, pp202-211, July, 1983

[10]W.C. Elmore, "The Transient Response of Damped Linear Networks with Particular Regard to Wide-Band Amplifiers", J. of Applied Physics, vol.19, No.1, pp55-63, January, 1948

[11]I.W. Sandberg & A.N. Willson, "Some Theorems on Properties of DC Equations of Non-Linear Networks", BSTJ, vol.48, pp1-34, 1969

[12]H. Kitazawa, "A Line Search Method for High Routing Rate", J. of Information Processing Society of Japan, vol.26, No.11, pp1366-1375, 1985 (in Japanese)

[13]S. Kodama & N. Suda, "Matrix Theory for System Control", Measurement Automation Society, The 2nd Edition, 1985, Ohm Press (in Japanese)