

# 合理偏差驱动的时钟线网构造及优化<sup>\*</sup>

赵 萌 蔡懿慈 洪先龙 刘 穗

(清华大学计算机科学与技术系, 北京 100084)

**摘要:** 提出了一种新的时钟布线算法, 它综合了 top-down 和 bottom-up 两种时钟树拓扑产生方法, 以最小时钟延时和总线长为目标, 并把合理偏差应用到时钟树的构造中。电路测试结果证明, 与零偏差算法比较, 该算法有效地减小了时钟树的总体线长, 并且优化了时钟树的性能。

**关键词:** 合理偏差; 时钟布线; 时钟线网; 时钟树

**EEACC:** 2570      **CCACC:** 7410D

**中图分类号:** TN47      **文献标识码:** A      **文章编号:** 0253-4177(2003)04-0438-07

## 1 引言

随着工艺发展到深亚微米阶段, 物理设计中连线造成的延迟已超过了门延迟, 电路的性能越来越依赖于互连线的性能。在布图综合阶段, 时钟信号的分布对于同步系统的操作和性能都是至关重要的。不同寄存器的时钟信号延时的差别, 我们定义为时钟偏差。如果时钟偏差控制不当, 就会影响系统的性能甚至会使系统产生错误的操作, 比如, 在寄存器中产生一个错误的数据信号的锁存。同时, 确定时钟信号分布的时钟布线也应该尽量占用较小的布线区域, 以便减小寄生电容、电感效应对系统时延、串扰和功耗的影响并减小芯片尺寸。随着集成电路制造技术的发展, 芯片特征尺寸进一步变小, 长线网的连线电阻大幅度增加。由于时钟线网有很多的输出端口, 并横跨整个芯片, 在芯片上时钟线网一般很大。因此, 由于寄生电阻、电容和电感对时钟线网性能的影响就变得越来越严重。这就使得在最近几年里, 不论是工业界还是学术界都对时钟线网的设计进行了大量的研究工作。

迄今为止, 在高性能及低功耗的时钟布线领域里已经进行了积极有效的研究。文献[1]提出了自上

而下的构造时钟树拓扑的 BB 算法, 它利用启发式方法划分电路, 使子电路的总负载电容基本相等。文献[2~4]提出了自下而上的构造时钟树拓扑的 KCR 算法, 它迭代地连接“相近的”成对子树, 能够解决时钟汇点不对称分布的问题。两种算法都有一定的局限性: BB 算法对于小规模的时钟布线不能充分发挥优势, 而 KCR 在时钟线网很大时不能从全局综合考虑。由于实际工艺参数影响, 零偏差布线不能精确达到, 文献[5~7]提出了有限偏差(bounded-skew)时钟布线算法 BST。BST 算法设定所有的时钟偏差都在一个统一的偏差限定值下, 零偏差时钟树只是 BST 在限定值为零时的特例。然而, 在实际电路中所有的时钟偏差不能以统一的值来限定, BST 仍然与实际情况有一定误差。其次, 由于负偏差能够使电路在小于关键路径延迟的时间内工作<sup>[8]</sup>, 能够减小电路的功耗, 负偏差一般被称之为有用偏差。所以, 我们应当尽量提高电路中负偏差的数量。

本文针对以上问题提出利用合理偏差进行时钟线网的构造和优化, 并以最小线长和最小时钟延时为目标; 同时提高时钟线网中有用偏差的数目, 优化时钟树结构及性能。本文余下的内容是这样安排的: 第 2 部分介绍延时模型和一些基本概念; 第 3 部分给出算法流程; 第 4 部分详细阐述了算法思想; 实验

\* 国家重点基础研究发展规划(编号: G1998030403) 及国家自然科学基金(批准号: 60176016) 资助项目

赵 萌 硕士研究生, 主要从事 VLSI 时钟布线算法研究。

蔡懿慈 副教授, 主要从事计算机设计自动化的教学与科研工作。

2002-05-24 收到, 2002-12-05 定稿

©2003 中国电子学会

结果在第5部分中列出。

## 2 预备知识

### 2.1 延时模型

与集总RC模型相比, Elmore延时模型<sup>\*</sup>大大提高了精确性, 并给出一个简单限定的表达式<sup>[9]</sup>。对于每一段互连线本文均使用 $\pi$ 型延时模型。给定连线树T, 从时钟源点 $s_0$ 到时钟汇点 $s_i$ 的Elmore延时模型表示如下:

$$\tau(0, i) = R_0 C_0 + \sum_{e(u, v) \in \text{Path}(0, i)} r l_{u, v} \left( \frac{c l_{u, v}}{2} + C_v \right) \quad (1)$$

其中  $R_0$  是时钟源点的驱动电阻;  $C_0$  是驱动电路的总负载电容。在给定线宽统一的情况下,  $r$  和  $c$  分别表示单位线长的电阻和电容。令  $e(u, v)$  表示  $s_v$  和其父节点  $s_u$  的连线, 那么连线电阻  $rl_{u, v}$  和电容  $cl_{u, v}$  均与线长  $l_{u, v}$  成正比。 $C_v$  表示根节点在  $s_v$  的子树的总负载电容。 $\text{Path}(0, i)$  表示从源点  $s_0$  到  $s_i$  的路径。表达式(1)的第一项与T的总线长成线性关系, 而第二项与  $\text{Path}(0, i)$  的长度成二次方的关系。

### 2.2 合理偏差

考虑一个简单的同步电路, 如图1所示。不失一般性, 在本文中设电路由正相边缘触发的触发器组成, 时钟偏差将在两个时钟汇点例如  $C_{01}$  和  $C_{02}$  之间产生。 $C_{01}$  和  $C_{02}$  分别是触发器  $FF_{01}$  和  $FF_{02}$  的时钟端。图2说明了在两种不同时钟偏差下电路的工作情况。不论在哪种偏差情况下, 我们定义在给定的时钟频率下能产生正确数据的时钟偏差为合理偏差。如果偏差过大, 从  $FF_{01}$  到  $FF_{02}$  就会发生不正确的电路操作, 或者数据产生太早(也就是double-clocking), 或者数据产生太晚(即zero-clocking)。

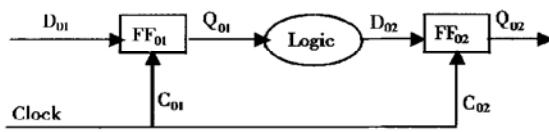


图1 简单的同步电路

Fig. 1 Simple synchronous circuit

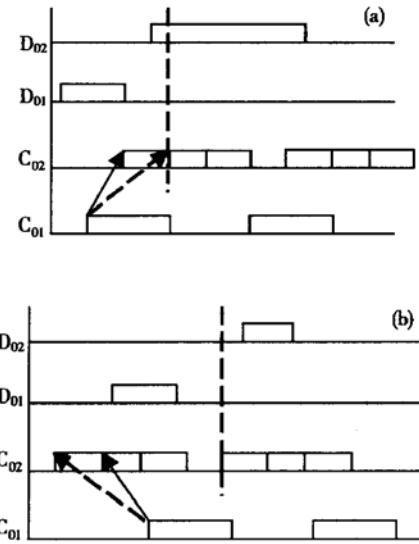


图2 (a)负偏差; (b)正偏差  
Fig. 2 (a) Negative skew; (b) Positive skew

一般来说, 为了保证在一个给定的时钟周期  $T_{\text{cp}}$  下正确的时钟操作, 在两个相邻的触发器  $FF_i$ 、 $FF_j$  之间合理的时钟偏差定义为:

负偏差限( $NSB_{ij}$ ): 在给定时钟频率下, 能够保证时钟正确操作的  $s_i$  和  $s_j$  之间的负偏差的最大值;

正偏差限( $PSB_{ij}$ ): 在给定时钟频率下, 能够保证时钟正确操作的  $s_i$  和  $s_j$  之间的正偏差的最大值。

如果  $FF_i$  和  $FF_j$  是相邻的, 那么

$$NSB_{ij} = \text{MIN}(d_{\text{logic}}) + d_{\text{ff}} - d_{\text{hold}} \quad (2)$$

$$PSB_{ij} = T_{\text{cp}} - \text{MAX}(d_{\text{logic}}) - d_{\text{setup}} - d_{\text{ff}} \quad (3)$$

否则

$$NSB_{ij} = PSB_{ij} = \infty \quad (4)$$

其中  $d_i$  和  $d_j$  分别是时钟源点到达触发器  $FF_i$  和  $FF_j$  的时间; 而  $\text{MAX}(d_{\text{logic}})$  和  $\text{MIN}(d_{\text{logic}})$  分别表示触发器  $FF_i$ 、 $FF_j$  之间组合单元的最长和最短路径延迟;  $d_{\text{ff}}$  是通过触发器的传输延迟;  $d_{\text{setup}}$  和  $d_{\text{hold}}$  分别是触发器锁存数据的建立时间和保持时间;  $T_{\text{cp}}$  为时钟周期。

### 2.3 问题描述

一般来说, 时钟布线问题可以公式化地表示为如下形式:

问题的输入: 时钟源点  $s_0$ , 时钟汇点(或者说是寄存器)集合  $\{s_1, s_2, \dots, s_n\}$ , 时钟汇点之间的偏差约

\* 由于 Elmore 延迟模型不能解决电感对于延迟的影响, 因此本模型的使用范围是在 GHz 的操作频率之下。

束集合:  $\{[\text{NSB}_{ij}, \text{PSB}_{ij}] \mid d_i - d_j \in [\text{NSB}_{ij}, \text{PSB}_{ij}]; i, j = 1, 2, \dots, n\}$ .

问题的目标: 利用节点间的偏差约束关系, 构造一个最小成本的时钟树  $T$ , 树的根结点是时钟源点  $s_0$ , 叶子节点是各个时钟汇点  $s_1, s_2, \dots, s_n$ . 其中构造成本是时钟树总线长和时钟汇点最大延时的加权和.

说明: 为了提高问题的实际应用性, 我们是在曼哈顿(Manhattan)网格上走线.

### 3 算法流程

一般的时钟树产生算法分为两个步骤: 抽象拓扑产生和拓扑嵌入. 本文的算法将这两个步骤结合起来, 即将拓扑构造和合并子树根节点同时完成. 采用这种拓扑和连线结合方法的优点是: 一方面可以得到理想布线结果, 另一方面使两个步骤之间信息的反馈更加迅速, 减小了复杂度, 提高了算法执行的速度. 我们从叶子节点开始, 每次产生拓扑(找到要合并的节点)后, 就按照新产生的拓扑合并时钟树的节点.

合理偏差驱动的时钟布线算法流程如图 3 所示. 其中, LSTProduceTopology 是利用综合方法产生时钟树的拓扑; 它兼顾了自上而下和自下而上两种算法的优点, 不仅能够总体把握时钟汇点的情况,

```

Input: 时钟源点  $s_0$ 
Sink= 时钟汇点(或者说是寄存器)集合 $\{s_1, s_2, \dots, s_n\}$ 
SB= 时钟汇点之间的偏差约束集合
Output: 时钟布线树  $T$ 
Procedure ConstructLST( $s_0$ , Sink, SB)
    待处理节点集合 temp_s= Sink;
     $T^0= \{s_0, \text{temp\_s}\}$ ;
    While ( $|\text{temp\_s}| > 1$ )
    {
         $T^* = \text{LSTProduceTopology}(\text{temp\_s})$ ;
         $T^0 = \text{LSOptimize}(T^*, T^0, \text{SB})$ ;
        ( $\text{temp\_s}, T^0$ ) = MergeTreeNode(temp_s,  $T^*$ ,  $T^0$ );
         $T^0 = \text{ConstructionOptimize}(T^0)$ ;
        SB= Update(SB);
    }
     $T = T^0$ 

```

图 3 合理偏差驱动的时钟布线算法

Fig. 3 Legitimate skew driven clock routing algorithm

还能使产生的时钟布线树线长明显减小. MergeTreeNode 是根据拓扑  $T^*$  暂时构造当前时钟树  $T^0$  为零偏差形式, 并且节点集合 temp\_s. LSOptimize 是根据合理偏差约束集合 SB 优化时钟树; 它充分利用了组合逻辑电路的合理偏差, 更加接近电路的实际情况; 由于合理偏差的引入, 时钟树的线长得到了优化; 而且增加了负偏差, 减小了时钟线网的功耗<sup>[8]</sup>. ConstructionOptimize 是优化时钟树的结构; 它消除时钟树结构的不平衡, 保证了时钟线网总线长的不增长或减小. Update 是更新时钟线网的合理偏差限集合.

### 4 算法详述

#### 4.1 时钟树拓扑生成

时钟树的拓扑生成有两种方法: 自上而下和自下而上. 自上而下方法<sup>[1, 9]</sup>从全局综合考虑, 可以总体把握时钟汇点的情况, 但对于小规模结果较差; 自下而上方法<sup>[2]</sup>从底层着手, 优先合并距离最近的节点, 产生的时钟布线树线长会明显减小, 但是当节点数目较多的时候, 问题规模较大, 而且不易控制节点之间的偏差约束.

我们采用自上而下和自下而上相结合的拓扑生成算法, 算法流程见图 4. 其中的 Top-down 算法见文献[1, 9], Bottom-up 算法见文献[2, 3]. 本算法综合了以上两种方法的优点: 在高层时钟线网很大时, 采用自上而下的拓扑构造, 可以从全局综合考虑优化; 当到达底层, 时钟汇点数目较少时, 采用自下而上的算法, 对于底层的时钟布线也能充分考虑优化. 因此, 避免了小规模时自上而下方法的结果较差和节点数目较多时自下而上方法对于偏差约束很难控制的缺点.

```

Input:  $S$ : 时钟汇点集合, 临界数目 group
Output: 最小 cost 的时钟布线树拓扑结构  $T^*$ 
Procedure LSTproduceTopology ( $S$ )
    While ( $|S| > 2$ )
    {
        If ( $|S| > \text{group}$ ) Top-down( $S$ )
        Else Bottom-up( $S$ )
        把生成的父节点加入集合  $S$  更新拓扑结构  $T^*$ 
    }

```

图 4 合理偏差时钟树 LST 拓扑生成算法

Fig. 4 Legitimate skew clock tree topology generation algorithm

## 4.2 合理偏差的利用

一般来说, 在零偏差的时钟树中任意一个父节点到两个子树时钟汇点的延时相等。这样来说, 构造零偏差时钟树时父节点的位置就惟一确定下来。我们的算法是利用合理偏差使父节点的位置自由度增大, 根据目标函数的要求, 运用图形处理方法选出最优的父节点位置。这时, 父节点的位置范围由正负偏差限决定。根据预备知识 2.2 中正负偏差限的定义, 推导出时钟树中任意两个节点的正负偏差限:

如果节点  $s_a$  和  $s_b$  是时钟汇点, 那么

$$\text{NSB}_{ab} = \text{NSB}_{ij} \quad (5)$$

$$\text{PSB}_{ab} = \text{PSB}_{ij} \quad (6)$$

如果节点  $s_a$  和  $s_b$  其中至少一个不是时钟汇点, 而是子树根节点, 即么

$$\begin{aligned} \text{NSB}_{ab} &= \min(d(a, i) - d(b, j) + \text{NSB}_{ij}, d(a, k) \\ &\quad - d(b, l) + \text{PSB}_{lk}) \end{aligned} \quad (7)$$

$$\begin{aligned} \text{PSB}_{ab} &= \min(d(b, j) - d(a, i) + \text{PSB}_{ij}, d(b, l) \\ &\quad - d(a, k) + \text{NSB}_{lk}) \end{aligned} \quad (8)$$

其中  $s_i$  和  $s_k$  是以  $s_a$  为根节点的子树的子节点;  $s_j$  和  $s_l$  是以  $s_b$  为根节点的子树的子节点; 而  $d(x, y)$  表示节点  $s_x$  到节点  $s_y$  的延迟时间。

我们用  $\text{mid}$  来表示父节点的位置,  $\text{mid}$  是父节点到左子节点(也就是左子树根节点)距离占左右子节点距离的比例。

零偏差时, 父节点位置:

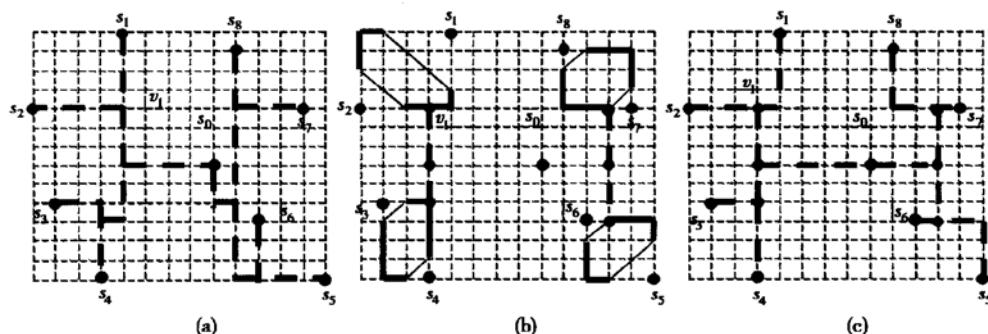


图 5 (a) 零偏差时钟树; (b) 利用合理偏差来优化; (c) 合理偏差时钟树

Fig. 5 (a) Zero-skew clock tree; (b) Optimizing with legitimate skew; (c) Legitimate skew clock tree

$$\begin{aligned} \text{mid} &= (t_2 + r \times C_2 \times l + r \times c \times l \times l/2 - t_1) \\ &/ (r \times c \times l \times l + r \times l \times (C_1 + C_2)) \end{aligned} \quad (9)$$

利用合理偏差时, 父节点位置范围:

$$\begin{aligned} \text{mid1} &= (-\text{NSB} + r \times C_2 \times l + r \times c \times l \times l/2) \\ &/ (r \times c \times l \times l + r \times l \times (C_1 + C_2)) \end{aligned} \quad (10)$$

$$\begin{aligned} \text{mid2} &= (\text{PSB} + r \times C_2 \times l + r \times c \times l \times l/2) \\ &/ (r \times c \times l \times l + r \times l \times (C_1 + C_2)) \end{aligned} \quad (11)$$

其中  $l$  是左右子节点的距离;  $t_1$  和  $t_2$  分别是左右子节点到左右子树叶子节点的延时;  $C_1$  和  $C_2$  分别是左右子节点的负载电容;  $r$  和  $c$  分别是单位线电阻和单位线电容; NSB 和 PSB 是左右子节点的正负偏差限。采用合理偏差时, 父节点的位置  $\text{mid}$  在  $\text{mid1}$  和  $\text{mid2}$  之间。

图 5(a) 给出了零偏差时钟树的最佳实际布线图。可以得出, 最佳零偏差布线的总线长为 48 个方格长, 时钟汇点延时均为 11 个单位延时(用线长延迟模型近似)。图 5(b) 给出了利用合理偏差来调整时钟树的过程。图中的黑实线是利用合理偏差限算出的父节点的可能位置; 然后, 再根据布线目标选出最佳的父节点位置。例如, 图 5(b) 中  $s_1$  和  $s_2$  父节点  $v_1$  的位置就发生了移动, 它是考虑  $s_3$  和  $s_4$  的父节点合理位置范围得到的最佳位置。图 5(c) 给出了最后得到的合理偏差时钟树布线图。它不仅满足任两个时钟汇点的偏差要求, 而且总线长为 43 个方格长, 除了  $s_1$  点为 13 个单位延时以外, 其它时钟汇点延时均不大于 11 个单位延时。总体来说, 合理偏差时钟树布线算法达到了总线长减小, 大部分时钟汇点延时不增长, 而且时钟布线树的功耗也有一定程度的减小。

### 4.3 时钟树的优化

本算法对两种结构的时钟树进行了优化。图 6(a)是对布线过程中形成的不平衡的时钟树结构进行调整。由于构造时钟布线树时两棵子树并不一定要求在根节点处合并，使得布线树的拓扑构造有更大的灵活性，因此，我们在每次的迭代过程中对时钟树的不平衡结构进行如图 6(a)的优化，保证在不增大布线成本的条件下，使左右子树尽量平衡。图 6(b)是对时钟树节点的合并顺序进行调整，如图 6(b)所示，原来的时钟树是  $s_1$  和  $s_2$  先进行合并，调整后是  $s_1$  和  $s_3$  先进行合并；这种优化方法减小了布线成本，且使左右子树时延更加平衡。

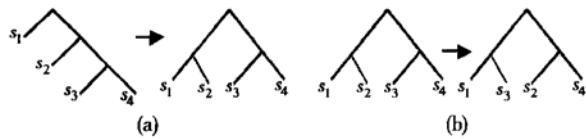


图 6 (a) 不平衡时钟树结构的变化; (b) 减小布线成本的拓扑结构优化

Fig. 6 (a) Optimization to remove unbalance; (b) Optimization to reduce routing cost

## 5 实验结果

上述算法在 SUN e450 工作站上已经实现，并

对实际电路进行了测试。表 1 是部分测试电路数据，来自北卡州立大学(见网站 [www.cbl.ncsu.edu/CBL-Docs/iscas89.html](http://www.cbl.ncsu.edu/CBL-Docs/iscas89.html))。表 2 为测试结果，其中 Top-down 算法见文献[1]，Bottom-up 算法见文献[2, 3]，综合拓扑的零偏差布线算法见文献[10]。互连线的单位电阻和单位电容分别为  $0.004\Omega/m$  和  $20pF/m$ 。

表 1 测试电路数据

Table 1 Test circuits data

电 路	s1423	s5378	s15850
时钟汇点数	74	179	597
合理偏差约束数	77	174	317
布线区域长、宽/ $\mu m$	14000/10000	13000/12000	16000/15000

从表 2 中看出本文实现的合理偏差布线及优化算法与前三种算法相比，时钟布线总线长有了明显改善，最大时延与前三种算法相比也是最优。由于有合理偏差(量级为  $10^{-12}$ )的严格限制，最大的时钟偏差接近零偏差布线的结果，对于较大电路(如 s15850)还有显著减小；而且本算法的速度较快，运行时间与前三种算法相比基本不变。不同的是，本算法与实际电路更加贴近，考虑了实际的情况；同时，本算法的结果还包含负偏差，达到了减小功耗的目的<sup>[8]</sup>。图 7 给出了本算法在 s5378 和 s15850 电路上的布线结果。

表 2 不同算法的结果比较

Table 2 Compared results with different algorithms

电 路	算 法	Top-down 算法	Bottom-up 算法	综合拓扑的零偏差布线	本文算法
s1423	总线长/ $\mu m$	119465	110551	107015	99636
	最大时延/ns	$5.36 \times 10^{-11}$	$6.01 \times 10^{-11}$	$4.62 \times 10^{-11}$	$4.53 \times 10^{-11}$
	最大偏差/ns	$2.32 \times 10^{-13}$	$2.12 \times 10^{-11}$	$1.94 \times 10^{-12}$	$4.93 \times 10^{-12}$
	运行时间/s	0.10	0.07	0.11	0.11
	负偏差个数	—	—	—	34
s5378	总线长/ $\mu m$	208036	177282	173368	164591
	最大时延/ns	$1.28 \times 10^{-10}$	$1.10 \times 10^{-10}$	$1.09 \times 10^{-10}$	$1.02 \times 10^{-10}$
	最大偏差/ns	$4.38 \times 10^{-12}$	$1.03 \times 10^{-11}$	$2.08 \times 10^{-12}$	$1.24 \times 10^{-11}$
	运行时间/s	0.43	0.23	0.55	0.56
	负偏差个数	—	—	—	70
s15850	总线长/ $\mu m$	576950	435409	434010	433736
	最大时延/ns	$4.04 \times 10^{-10}$	$4.19 \times 10^{-10}$	$4.26 \times 10^{-10}$	$3.71 \times 10^{-10}$
	最大偏差/ns	$2.64 \times 10^{-10}$	$5.85 \times 10^{-12}$	$2.57 \times 10^{-11}$	$3.89 \times 10^{-12}$
	运行时间/s	3.35	1.36	3.40	3.37
	负偏差个数	—	—	—	195

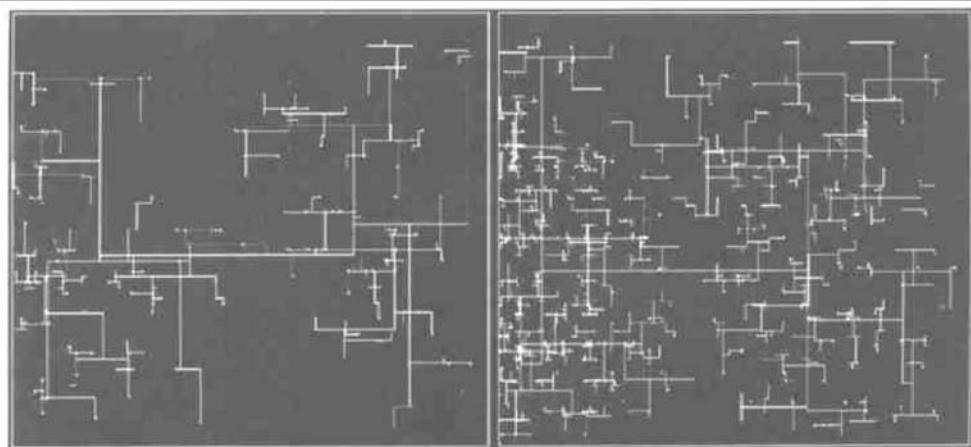


图 7 s5378 和 s15850 电路上的布线结果图

Fig. 7 Routing results on s5378 and s15850

## 6 结论

本文提出了一种新的时钟树布线算法。它考虑了实际逻辑电路中的偏差要求，并利用合理偏差使时钟树的布线条数减小，最大时延减小，时钟线网中的负偏差增多，并把合理偏差与综合拓扑生成算法、时钟树的优化有机地结合在一起，避免了布线的盲目性，提高了布线质量，算法的速度较快。

## 参考文献

- [1] Chao T H, Hsu Y C, Ho J M. Zero skew clock net routing. 29th ACM/IEEE Design Automation Conference, 1992: 518
- [2] Kahng A B, Cong J, Robins G. High-performance clock routing based on recursive geometric matching. Proc ACM/IEEE Design Automation Conf, 1991: 322
- [3] Cong J, Kahng A B, Robins G. Matching-based methods for high-performance clock routing. IEEE Trans Comput-Aided Des Integr Circuits Sys, 1993, 12: 1157
- [4] Cong J, Kahng A B, Robins G. On clock routing for general cell layouts. Proc IEEE Int ASIC Conf, 1991, 14: 5. 1
- [5] Boese K D, Kahng A B. Zero skew clock routing trees with minimum wirelength. Proc IEEE 5th Int ASIC Conf, Rochester, 1992: 1. 1. 1
- [6] Edahiro M. Minimum skew and minimum path length routing in VLSI layout design. NEC Res Dev, 1991, 32: 569
- [7] Cong J, Koh C K. Minimum-cost bounded-skew clock routing. Proc IEEE Int Symp On Circuits and Systems, 1995, 1: 215
- [8] Fishburn J P. Clock skew optimization. IEEE Trans Comput, 1990, 39(7): 945
- [9] Li Zhiyan, Yan Xiaolang. High speed multilevel staged clock routing. Chinese Journal of Semiconductors, 2000, 21(3): 290 (in Chinese) [李芝燕, 严晓浪. 高速多级时钟网布线. 半导体学报, 2000, 21(3): 290]
- [10] Liu Yi, Zhao Meng, Hong Xianlong, et al. A clustering-based algorithm for zero-skew clock routing with buffer insertion. Chinese Journal of Computer Aided Design and Graphics, 2002, 14(2): 97 (in Chinese) [刘毅, 赵萌, 洪先龙, 等. 一种基于结群的零偏差时钟布线算法. 计算机辅助设计与图形学学报, 2002, 14(2): 97]

## Legitimate Skew Driven Clock Tree Routing and Optimization<sup>\*</sup>

Zhao Meng, Cai Yici, Hong Xianlong and Liu Yi

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

**Abstract:** A new clock routing algorithm is proposed, which synthesizes top-down and bottom-up clock tree topology generation algorithms. This algorithm aims at minimum clock delay and total wire length and utilizes legitimate skew to construct a clock tree. The experimental results show that, compared with zero-skew clock routing, this algorithm effectively reduces the total wire length of clock tree and optimizes the performance of clock tree.

**Key words:** legitimate skew; clock routing; clock net; clock tree

EEACC: 2570 CCACC: 7410D

Article ID: 0253-4177(2003)04-0438-07

\* Project supported by State Key Development for Basic Research of China(No. 1998030403) and National Natural Science Foundation of China (No. 60176006)

Zhao Meng graduate student. She is engaged in the research on clock routing algorithm of VLSI.

Cai YiCi associate professor. She is engaged in the research and teaching of computer design automation.

Received 24 May 2002, revised manuscript received 5 December 2002

©2003 The Chinese Institute of Electronics