考虑工艺参数变化的安全时钟布线算法*

蔡懿慈[1,2] 熊 焰[1] 洪先龙[2] 刘 毅[2]

(1 中国科学技术大学计算机科学与技术系, 合肥 230026; 2 清华大学计算机科学与技术系软件研究所, 北京 100084)

摘要 在超深亚微米(VDSM)工艺下, 由光刻工艺带来的光学邻近效应不可忽略, 时钟偏差受到光学邻近效应等工艺参数变化的影响非常严重. 提出了一种带缓冲器插入的安全时钟布线算法, 来防止因光学邻近造成线宽变化对时钟系统的影响. 该算法提出了"分支敏感因子"(BSF)的概念, 通过构造特殊的树型拓扑结构和布线过程中的缓冲器插入等操作, 达到总体布线长度和偏差灵敏度的平衡. 实验结果表明, 算法可以得到一个抗光学邻近效应工艺参数变化的可靠时钟布线树, 时钟偏差被有效地控制在合理范围之内.

关键词 时钟布线 工艺参数变化 时钟偏差 分支敏感因子 缓冲器插入

在同步集成电路设计中, 时钟信号负责同步单元的协同工作, 它直接决定着系统的正确性和可靠性. 当今的处理器时钟频率已达千兆赫兹, 时钟周期越来越短, 这要求更加精确地控制时钟信号的传递. 时钟信号到达同步单元的先后关系被定义为时钟偏差. 传统的时钟网络设计要求零偏差布线, 即信号同时到达所有同步单元[1], 后来发展为有限偏差布线和有用偏差布线. 文献[2,3]提出了DME算法(deferred-merge embedding), 以最小布线长度实现零偏差. 文献[4, 5]又把其推广到有限偏差布线. 时钟偏差在文献[6, 7]又具体划分为正偏差和负偏差.

超深亚微米工艺条件下, 连线宽度越来越细, 而连线厚度相对于线宽变化要小得多, 使线间耦合电容增大, 互连线延迟大大超过了门延迟. 除了传输路径长度、负载电容不对称等基本因素引起的时钟偏差之外, 连线宽度受光学邻近效应的影响也越来越严重, 从而导致时钟偏差. 由于芯片生产过程中的蚀刻程度不同、掩膜版对准误差, 及集成电路特征尺寸在 0.18 μm以下时, 光的衍射和折射现象严重产生光刻图像误差等不确定因素, 造成芯片上晶体管的实际尺寸、连线的

2005-01-28 收稿, 2005-06-23 收修改稿

* "八六三"国家高技术研究发展计划(2005AA1Z1230)及国家自然科学基金项目(批准号: 90307017)资助项目

实际宽度都会同期望值有一定的差别; 另外, 从静态模型中得到的一些晶体管参数往往也会和实际参数有所出入[8,10]. 所有这些工艺参数的变化和环境因素影响给现代微处理器中的时钟系统设计带来了冲击, 给系统的鲁棒性提出了更高的要求.

为了提高系统的抗干扰性, 文献[11, 12]分别采用网型和网、树混合型的拓扑结构来设计时钟网络. 但是由于占用了较多的布线资源, 给系统功耗带来了沉重的负担. 当前迫切需要一种可靠的树型时钟布线算法, 能够容忍工艺参数带来的变化. 文献[13]提出一种时钟树的拓扑结构算法, 将偏差要求严格的时钟端点对提前合并, 以减小时钟源点公共路径上的线宽变化影响, 但是文献中没有给出如何在物理布线中实现其拓扑结构. 文献[9]采用改变线宽的方法来使时钟树更加可靠, 但是它只考虑了零偏差的情况.

本文分析了时钟树的各个不同部分对时钟偏差的影响, 推导了时延对不同部分连线宽度的灵敏度公式, 提出了“分支敏感因子”(BSF)的概念, 来反映连线宽度的变化对时钟偏差约束的影响程度. 在构造时钟树的拓扑结构时, 我们采用两种方法控制由于线宽变化导致的时钟偏差. 首先, 通过引入“优先合并方向”指导节点的合并过程, 减小了分支 BSF 因子, 从而避免了引起时钟偏差的连线生成. 第二, 采用适当的缓冲器插入限制了子树的负载大小, 减小了线宽变化对时延造成的影响, 进一步控制了线宽变化引起的时钟偏差. 实验结果表明, 算法可以得到一个抗线宽变化的可靠时钟布线树, 时钟偏差被有效地控制在合理范围之内. 余下部分的论文组织如下: 第 1 节介绍了问题定义和所采用的时延模型; 第 2 节分析了时钟树的各个不同部分对时钟偏差的影响, 推导了时延对不同部分连线宽度的灵敏度公式, 阐述了 BSF 因子的概念和计算方法; 在第 3 节中介绍了我们提出的时钟布线算法; 最后给出实验结果和结论.

1 问题描述

这里我们只考虑单时钟源点s_0的情况, $S = \{s_1, s_2, \cdots, s_n\}$代表同步单元对应的时钟端点集合. 假设$t_i$和$t_j$分别代表时钟信号到达端点$s_i$ s_j的时间, 那么s_i s_j之间的时钟偏差可以表示为$skew_{i,j} = t_i - t_j$. 如果s_i s_j对应的同步单元是相邻的, 即它们之间仅由组合逻辑电路相连, 且数据流向是从s_i到s_j, 为了避免电路功能错误, 它们之间的正负时钟偏差必须控制在一个合理范围之内[6]:

$$t_1 - t_2 \leqslant PSB_{1,2} = C_P - \mathrm{Max}(D_{\mathrm{logic}}) - D_{\mathrm{ff}} - D_{\mathrm{setup}}, \tag{1}$$

$$t_2 - t_1 \leqslant -NSB_{1,2} = \mathrm{Min}(D_{\mathrm{logic}}) + D_{\mathrm{ff}} - D_{\mathrm{hold}}, \tag{2}$$

其中 C_p 是时钟周期, D_{logic} 是组合逻辑电路延迟, D_{ff} 代表同步单元内延迟, D_{setup} 和 D_{hold} 分别是数据的建立时间和保持时间.

定义 1　带缓冲器插入的安全时钟布线树问题: 给定时钟源点 s_0, 时钟端点集合 $S=\{s_1, s_2, \cdots, s_n\}$, 并给出时钟端点之间的偏差约束集合 $C=\{t_i-t_j\in[NSB_{ij}, PSB_{ij}]\}$. 构造一时钟布线树 T, 树的根结点是时钟源点 s_0, 叶子节点是各个时钟端点, 并确定缓冲器插入的合理位置, 使得任意两时钟端点间的时钟偏差都满足集合 C 的约束要求, 并且受线宽变化影响最小.

1.1　符号说明

为了便于说明, 表 1 中列出了本文中所使用的参数标记符号及其含义说明:

表 1　参数标记符号说明

T_{Di}	从时钟源点到时钟端点 i 的时延值
i	在 RC 树中连接节点 i 到它的父节点的分支
$P(i)$	从根节点到端点 i 的节点集合(不包含根节点)
$U(i)$	从根节点到节点 i 的所有分支集合
R_i	分支 i 的电阻
R_{Cij}	从根节点到节点 i, j 的公共分支的电阻总和
C_i	分支 i 处的负载电容
C_{di}	下游电容, 节点 i 和它的子孙节点所有的负载电容总和
r_0, c_0	每μm^2 的单位电阻和单位电容
D_b R_b C_b	缓冲器单元的本征延迟, 驱动电阻和负载电容

1.2　时延模型

Elmore延迟模型是目前互连延迟计算中常用的计算模型, 尤其是在时钟网络的优化中. 它通过分布式电阻、电容的乘积求得, 是一阶阶跃响应时间的近似. 假设单位阶跃函数的瞬态响应为 $e(t)$, 其时间导数为 $e'(t)$, Elmore时延通常估算了 $e(t)$上升到 50%的时间:

$$T_{D_i}=\int_0^{\infty} te_i'(t)dt, \tag{3}$$

假设 S_0 是时钟的源点, S_i 是时钟端点, 那么从 S_0 到 S_i 的延迟可以表示为

$$T_{D_i}=t(s_0, s_i)=\sum_{k\in P(i)} R_k \bullet C_{dk}. \tag{4}$$

其中, $P(i)$代表从源点 S_0 到端点 S_i 路径上的中间节点集合(包括端点 S_i). 我们用 $Tree_k$ 表示以 k 为根节点的子树, R_k 和 C_{dk} 分别是 $Tree_k$ 的驱动电阻和负载电容总和.

将 Elmore 延迟 T_D 作为延迟来评价是因为 T_D 是脉冲响应(即源点输入的单位脉冲电压的响应)的一阶矩(first order moment)或平均值(mean). Elmore 延迟模型具有简单封闭的计算表达式; 计算结果的保真性, 即虽然 Elmore 模型计算出的延迟仅仅是精确时延时间的严格上界, 但在 Elmore 模型下求得的最优解往往很接近实际的最优解.

2 偏差灵敏度与 BSF 因子

2.1 时延灵敏度

根据 Elmore 时延模型的计算公式, 对时延值 T_{Di} 求某段连线 j 的电阻、电容参数偏导, 我们可以得到时钟端点 i 的时延值对该连线分支 R 和 C 参数变化的灵敏度:

$$\begin{aligned}\frac{\partial T_{Di}}{\partial R_j} &= C_{dj}, \quad j \in U(i),\\ &= 0, \quad \text{否则.}\end{aligned} \tag{5}$$

$$\frac{\partial T_{Di}}{\partial C_j} = R_{Cij}. \tag{6}$$

一段连线分支的线宽变化之后, 两类时钟端点的时延将受到影响: 如果时钟端点位于连线驱动的子树内部, 连线宽度改变引起驱动电阻值变化, 因此时延值对电阻的灵敏度是子树负载 C_{dj}; 子树外部时钟端点的时延也会受到影响, 因为从该时钟端点到时钟源点与该连线分支到时钟源点共享了一段路径, 时延值对连线电容变化灵敏度是公共电阻 R_{Cij}.

由偏导数的传递特性, 可以得到时延对连线分支线宽的偏导数计算公式:

$$\frac{\partial T_{Di}}{\partial w_j} = \frac{\partial T_{Di}}{\partial R_j}\frac{\partial R_j}{\partial w_j} + \frac{\partial T_{Di}}{\partial C_j}\frac{\partial C_j}{\partial w_j}. \tag{7}$$

连线宽度大小影响着连线分支的电阻和电容值, 若给定了常系数 r_0 和 c_0, 电阻和电容对线宽 w 的偏导数可以由下面公式计算得到

$$\frac{\partial R}{\partial w} = -\frac{r_0}{w^2}l, \qquad \frac{\partial C}{\partial w} = c_0 l. \tag{8}$$

将(8)式带入(7)式, 时延值对连线线宽的灵敏度可表示为

$$\frac{\partial T_{Di}}{\partial w_j} = \left(c_0\frac{\partial T_{Di}}{\partial C_j} - \frac{r_0}{w^2}\frac{\partial T_{Di}}{\partial R_j}\right)\times l. \tag{9}$$

若时延灵敏度为正值, 说明增加该段连线的线宽会增加时钟信号从源点传递到时钟端点 i 的时延; 相反, 如果时延灵敏度为负值, 则增加线宽会导致时延的减小.

2.2 偏差灵敏度与 BSF 因子

定义 2 偏差灵敏度. 假设 k 是时钟布线树中的一段连线分支, 连线线宽为 w_k, i, j 是受偏差约束的一对时钟端点, 它们之间的时钟偏差对连线 k 线宽变化的灵敏度可以定义为

$$\frac{\partial Skew_{i,j}}{\partial w_k} = \frac{\partial T_{Di}}{\partial w_k} - \frac{\partial T_{Dj}}{\partial w_k}. \tag{10}$$

对于两个受偏差约束的时钟端点, 偏差灵敏度体现了各连线分支对这对偏差的相关程度. 偏差灵敏度越高, 表明如果该段连线宽度因工艺参数变化而改变, 会给它们之间的时钟偏差带来较大的影响. 通常情况下, 一段连线分支影响着很多对受偏差约束的时钟端点, 我们需要有一个总体的尺度进行评价.

定义 3　分支敏感因子(BSF). 对于时钟布线树中的连线分支 k, 所有受约束的时钟偏差对其线宽的加权灵敏度总和称为该段连线的分支敏感因子(branch sensitivity factor):

$$BSF(k) = \sum_{<i,j>\in C} \frac{\left|\partial Skew_{i,j} \middle/ \partial w_k\right|}{NSB_{i,j} + PSB_{i,j}}. \tag{11}$$

由于各个偏差约束的合理范围不一定相同, BSF 因子采用了对偏差灵敏度加权求和. 时钟网络设计得是否合理, 是否容易受线宽变化的影响, 各段连线上的 BSF 因子给我们提供了一个判断标准, 线段 BSF 因子越小, 该线段对线宽的变化越不敏感, 受线宽变化的影响越小; 反之, 如果线段 BSF 因子越大, 该线段对线宽的变化越敏感, 受线宽变化的影响越大. 线段的 BSF 因子与时钟的拓扑结构相关, 时钟网络的拓扑结构不同, 其各线段的 BSF 因子不同, 受线宽变化影响的程度也就不同. 下面用一个简单例子加以说明.

图 1 中构造了包含有四个时钟端点的两种不同“H 树”的拓扑结构, 其中每个端点处的负载都相同, 相当于单位连线电容的四倍, 端点 1 和 3 之间有时钟偏差的

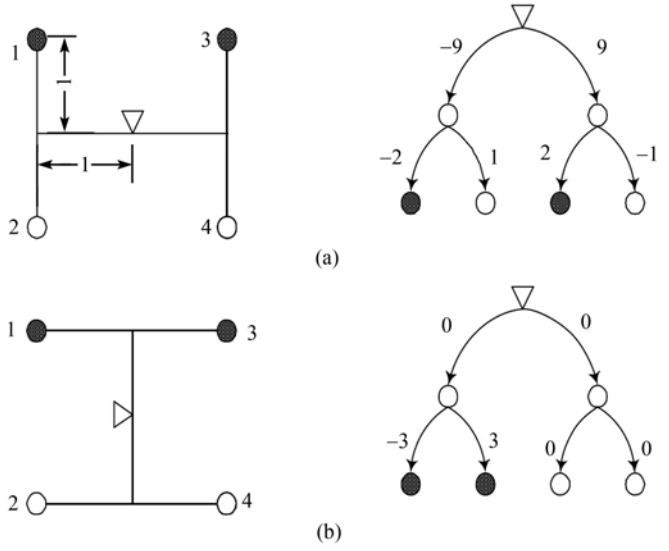

图 1　不同拓扑时钟树的 BSF 因子分布

约束条件限制. 根据公式(9)~(11), 可以计算出时钟布线树中的所有 6 段连线的 BSF 因子. 这里给出的不是具体的数值计算结果, 而是给出了相应的比例关系.

从图 1 中(a)和(b)两种不同的拓扑结构计算得到的各线段 BSF 因子的结果比较可以看到, (a)中的线段 BSF 因子明显大于(b)中的线段 BSF 因子. 而图 1(a)与(b)的区别仅在于节点的合并顺序不同, (a)是将节点 1, 2 和 3, 4 分别先合并, 然后再合并其上一级的父亲节点; 而(b)是将接点 1, 3 和 2, 4 分别先合并, 然后再合并其上一级的父亲节点. 由此可以得到一个结论, 如果将受偏差约束的时钟端点优先进行合并, 得到的时钟树各段连线上的 BSF 因子较小, 如图 1(b)所示. 而按照此原则构造的时钟树, 其偏差受线宽变化的影响相对较小. 这与文献[13]的结论是相同的, 但要付出的代价是可能会带来整体布线树的线长增加.

3 安全时钟布线算法

为了减小时钟树各分支的DSF因子, 同时又不使时钟的总线长增加太大, 这里我们提出一种抗线宽变化的安全时钟布线算法, 算法由两个关键步骤组成: 在构造时钟布线树的过程中引入“优先合并方向”处理偏差灵敏度 DSF 和布线长度的平衡; 以适当的缓冲器插入进一步减小连线分支的 BSF 因子.

3.1 拓扑结构生成

为了降低时钟布线树的 BSF 因子, 需要尽早合并有偏差约束的时钟端点对, 而同时要避免增加连线长度. 这里我们引入“力矢量方法”思想进行这两方面的平衡, 下面给出节点的“引力”和节点的“优先合并方向”的定义.

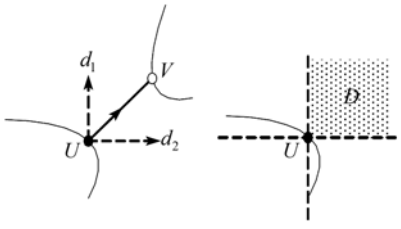

图 2 节点 U 的理想方向

定义 4 节点的“引力”. 如果两个时钟节点间有偏差约束, 我们就定义这两个时钟节点之间存在着吸引力, 引力的作用会使它们尽早合并, 以减小其所在时钟分支的 BSF 因子. 如图 2 所示, 节点 U 和 V 之间有时钟偏差约束, 则在节点 U 和节点 V 之间存在一个引力.

定义 5 节点的“优先合并方向”. 如果一个节点 U 与另外一个节点 V 之间存在偏差约束, 则定义节点 V 所在的象限为节点 U 的优先合并方向, 如图 2 中所示的阴影区域 D.

如果单纯用引力的模型进行节点的合并过程, 由于它是一种过于贪婪的方法, 会导致时钟线网线长的增加. 为此, 我们在引力模型上定义了“优先合并方向”, 在优先合并方向的指导下进行节点的合并过程, 可以达到既减小分支 BSF 因子又不使线长增加的目的. 我们重新定义节点之间的距离, 将加入偏差约束考

虑后的节点之间距离定义为加权距离函数:

$$Weight(i,j)=\begin{cases} dist(i,j)\times\beta, & j \text{ 在}i\text{的优先合并方向;} \\ dist(i,j)\div\beta, & j \text{ 在}i\text{优先合并方向的相反方向;} \\ dist(i,j), & \text{其他情况.} \end{cases} \tag{12}$$

这里 $0<\beta<1$, $dist(i,j)$表示两节点间的物理距离, 即为不同方向上的节点定义了合并的优先级, 在“优先合并方向”上的节点具有较高的优先级.

如图 2 所示, 节点 U 和节点 V 之间存在着引力. 在选择 U 的相邻节点进行合并时, 优先合并方向(图中阴影区域 D)上的时钟节点有着较高的优先级, 而并不是直接合并 U 和 V. 这样有引力的节点会尽早合并, 也不会增加过多连线.

3.2　缓冲器插入

在时钟网络的设计和优化中, 为了确保时钟信号的完整性, 即清晰的波形, 快速的上升/下降翻转, 常常需要插入大量的缓冲器, 但这种缓冲器的插入目的是为了能够驱动大负载电容, 加强时钟信号. 在时钟布线树抗线宽变化影响的设计中, 也可以利用缓冲器来屏蔽子树负载电容, 由于缓冲器的隔离作用, 当上游的连线分支宽度发生了改变, 不会波及到下游的时钟端点, 减小了分支的 BSF 因子, 从而可以减小由于线宽变化对偏差的影响.

下面通过一个例子对缓冲器插入的偏差灵敏度进行分析. 如图 3 所示, 假设时钟端点 i 和 j 存在偏差约束, 连线 a 是它们最近的公共父节点 V 的连线分支, 长度为 l, 负载为 C_L. 在没有缓冲器的情况下, 偏差 $skew_{i,j}$ 对分支 a 线宽的灵敏度为

$$\frac{\partial Skew_{i,j}}{\partial w_a}=\frac{\partial T_{Di}}{\partial w_a}-\frac{\partial T_{Dj}}{\partial w_a}=-\frac{r_0}{w_a^2}C_L\times l. \tag{13}$$

假设在子树 a 根节点处安置了一个缓冲器(R_b, C_b, D_b)对子树负载进行屏蔽, 则偏差 $skew_{i,j}$ 对分支 a 线宽的灵敏度变为

$$\left(\frac{\partial Skew_{i,j}}{\partial w_a}\right)_{\text{buffer}}=-\frac{r_0}{w_a^2}C_b\times l. \tag{14}$$

当子树负载 C_L 远大于缓冲器输入电容 C_b 时候, 缓冲器的插入会大大减小偏差 $skew_{i,j}$ 对连线分支 a 线宽的灵敏度. 由上面的分析可知, 缓冲器的插入可以减小线宽变化对偏差的影响, 提高时钟树抗干扰的能力. 尽管缓冲器的插入可以降低分支BSF因子, 但是过多的和不合理的缓冲器插入, 会带来芯片功耗的增加. 因此, 必须在合并两子树之前进行缓冲器插入位置的选择和

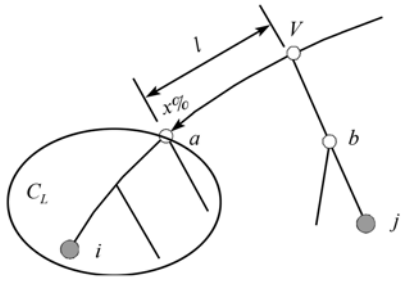

图 3　缓冲器插入

确定. 下面我们对缓冲器插入最佳位置进行推导.

如图 4 所示, 若在连线长度为 x%处已安置了一个缓冲器, 对子树负载进行屏蔽, 则偏差对该连线的灵敏度改变为

$$\frac{\partial \mathrm{Skew}_{i,j}}{\partial w_k} = R_b \times c_0 x\% \, l - \frac{r_0}{w_k^2} x\% \, l \times C_L - \frac{r_0}{w_k^2}(1-x\%)\, l \times C_b. \tag{15}$$

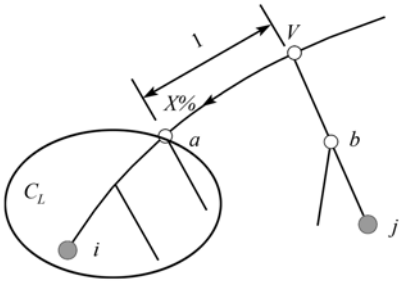

图 4 缓冲器插入对灵敏度的影响

考虑到负载 C_L 较大, 取 x%为零值, 灵敏度减小为 $-r_0 l C_b / w_k^2$. 由此可知, 缓冲器的最佳位置应该在子树的根节点处, 即如图 3 所示的 a 点.

根据上面的推导, 假设因光学邻近效应引起连线线宽由 w 变化到 $w \pm \Delta w$, 当 $\Delta w \cdot r_0 C_L \times l / w^2$ 接近最小偏差限时, 就执行缓冲器的插入操作以减小布线树的 BSF 因子.

算法的基本流程如下表所示:

表 2 布线程序的高层次描述

输入:	S: 时钟端点集合, C = [*NSB, PSB*]: 端点之间的偏差约束集合
输出:	时钟布线树 *T(S)*
程序:	
	While |S| > 1
{	
	计算 S 中的节点的优先合并方向
	根据综合权值来选择配对的节点
	缓冲器插入检查
	配对的子树进行合并操作
	更新节点集合 S
}	

4 实验结果

我们在 SUN Enterprise-450 工作站上用 C/C++语言实现了上述布线算法, 并对 ISCAS89 提供的 2 个测试用例和文献[1]中五个例子进行了测试(由于 r1~r5 这五个例子是为零偏差布线准备的, 并没有提供偏差约束条件, 我们随机构造了相应的偏差约束). 表 3 给出了测试电路的具体信息.

工艺参数变化对连线宽度的影响由两部分组成: 总体分布式变化和局部随机变化:

$$w = w_0 + \lambda x + \theta y + \delta, \tag{16}$$

上述公式中 w_0 是正常情况下的理想线宽, w 是芯片制造出来的实际线宽, λ和θ是总体分布式变化沿 x 和 y 方向的系数, 局部随机变化δ 可以认为服从正态分布: $N(0, \sigma^2)$.

表 3　测试电路信息

电路	时钟端点数目	偏差约束数目
s1423	74	77
s5378	179	174
r1	267	170
r2	598	261
r3	862	373
r4	1903	874
r5	3101	1255

在 0.18 μm 工艺条件下, 我们假设一般的连线宽度 w_0 为 0.54 μm, 标准偏差 σ 为 0.108 μm, 单位电阻和单位电容分别取 0.004Ω/μm 和 0.02 fF/μm. 缓冲器的参数定为 D_b = 5ps, R_b = 10Ω, C_b = 500 fF. 表 4 给出了一般时钟布线和我们安全时钟布线的布线结果比较.

表 4　布线结果比较

电路	普通时钟布线			安全时钟布线					
	总线长/μm	违反偏差约束数目	时钟树 BSF 因子	总线长/μm	增加	违反偏差约束数目	减小	时钟树 BSF 因子	缓冲器数
s1423	106378	10	1.02×10^{-4}	116541	10%	2	80%	3.01×10^{-5}	2
s5378	170775	16	5.28×10^{-4}	192815	13%	7	56%	1.70×10^{-4}	10
r1	129387	17	4.26×10^{-4}	142780	10%	3	82%	1.06×10^{-4}	2
r2	256440	42	1.62×10^{-3}	280297	9%	8	81%	1.58×10^{-4}	4
r3	329807	47	4.65×10^{-3}	367710	11%	13	72%	3.79×10^{-4}	12
r4	673186	54	2.20×10^{-2}	765521	14%	10	81%	1.75×10^{-3}	24
r5	996702	106	6.01×10^{-2}	1128271	13%	19	82%	2.60×10^{-3}	34

可以看到经过特殊拓扑结构的选取、缓冲器插入等优化策略, 大大降低了布线树的 BSF 因子, 违反偏差约束的现象也减小了 60~80%. 结果说明得到的时钟布线树对工艺参数变化的灵敏度被限制在一定范围之内, 结果更加稳定安全. 带来的总体布线长度增加不到 15%, 优于网型或混合型拓扑的时钟布线.

为了便于观察 BSF 因子对时钟布线树的影响, 我们详细比较了工艺参数变化后 r5 中的时钟偏差分布. 图 5 显示了所有 1255 个受约束的时钟偏差经过归一化的分布情况, 图中纵坐标单位是合理偏差限的倍数. 可以看到一般时钟布线算法的布线结果受工艺参数变化影响较大, 很多原本合理的偏差超出了它们的合理偏差限制范围(图中用虚线显示正、负偏差限). 采用我们的安全时钟布线算法, 在降低了布线树连线分支的 BSF 因子之后, 尽管受工艺参数变化影响互连线的线宽发生了改变, 但对时钟偏差影响较小.

5　结论

工艺参数变化和环境的不确定因素会导致实际的连线宽度与设计值产生误

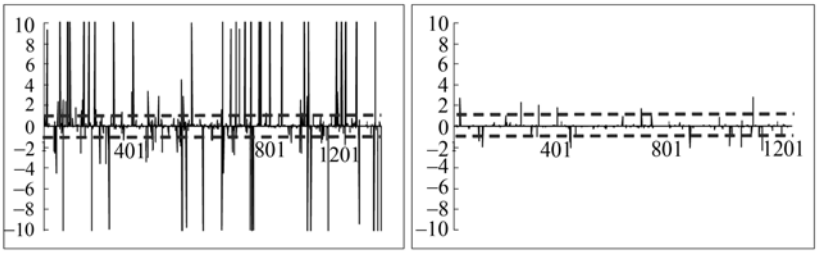

图 5 工艺参数变化后 r5 中的偏差分布情况
(a) 普通时钟布线结果, (b) 安全时钟布线结果

差, 影响时钟系统的信号传递时延. 本文提出了一种考虑工艺参数变化的安全时钟布线算法. 它引入布线树中连线分支敏感因子概念, 来反映布线树对线宽变化的灵敏度. 通过特殊的拓扑结构和适当的缓冲器插入, 限制了时钟偏差受连线分支线宽变化的影响, 增加了时钟布线树的可靠性. 进一步的工作要考虑缓冲器单元本身受工艺参数变化的影响问题.

参 考 文 献

1 Tsay R S. Exact zero skew. In: Proceeding of International Conference on CAD, 1991. 336~339
2 Chao T H, Hsu Y C, Ho J M, et al. Zero skew clock routing with minimum wire length. IEEE Transactions on Circuit and System, 1992, 39(11): 799~814[DOI]
3 Edahiro M. Minimum path-length equi-distant routing. In: Proc of Asia-Pacific Conf on Circuits and System, 1992. 41~46
4 Huang D J H, Kahng A B, Tsao C W A. On the bounded-skew clock and Steiner routing problems. In: Proceeding of 32nd Design Automation Conference, 1995. 508~513
5 Cong J, Kahng A B, Koh C K, et al. Bounded-skew clock and Steiner routing. ACM Trans on Design Automation of Electronics System, 1998, 3(3): 341~388[DOI]
6 Xi J G, Dai W W M. Useful-skew clock routing with gate sizing for low power design. In: Proceeding of 33rd Design Automation Conference, 1996. 383~388
7 Tsao C W, Albert Koh, Cheng K. UST/DME: A clock tree router for general skew constraints. IEEE/ACM International Conference on Computer-Aided Design, 2000, 400~405
8 Pavman Z H, Tony M, James D M. Characterization and modeling of clock skew with process variations. In: IEEE Custom Integrated Circuit Conference, 1999. 441~444
9 Pullela S, Menezes N, Pillage L T. Reliable non-zero skew clock trees using wire width optimization. In: Proceeding of 30th Design Automation Conference, 1993. 165~170
10 Liu Y, Nassif S R, Pileggi L T, et al. Impact of interconnect variations on the clock skew of a gigahertz microprocessor. In: Proceeding of 37th Design Automation Conference, 2000. 168~171
11 Desai M P, Cvijetic R, Jensen J. Sizing of clock distribution networks for high performance CPU chips. In: Proceeding of 33rd Design Automation Conference, 1996. 389~394
12 Su H, Sapatnekar S S. Hybrid structured clock network construction. In: International Conference on Computer-Aided Design, 2001. 333~336
13 Velenis D, Friedman E, Papaefthymiou M C. A clock tree topology extraction algorithm for improving the tolerance of clock distribution networks to delay uncertainty. In: Proc ISCAS, 2001. 4422~4425