

文章编号: 2095-4980(2013)06-1000-05

快速片上互连缓冲器插入时序优化分析

王新胜^{a,b}, 刘兴春^b, 喻明艳^{a,b}

(哈尔滨工业大学 a.航天学院, 黑龙江 哈尔滨 150001; b.信息与电气工程学院, 山东 威海 264209)

摘要: 互连线延时已成为制约大规模集成电路性能的瓶颈, 而缓冲器插入能很好解决互连线延时。Van Ginneken(VG)算法是缓冲器插入互连时序优化的经典算法, 针对此算法的3个主要操作过程进行改进, 利用红黑树数据结构存储路由拓扑数据结构, 缩短数据结构的更新访问时间; 利用快速冗余判别和排序方法减小解方案数量和求解最优的复杂度。通过标准测试电路集ISCAS89中的电路对本文方法进行测试, 测试结果表明, 虽然随着电路规模增加, 改进方法和传统方法运行时间都相应增加, 但改进方法的优势更加明显; 且随着缓冲器库规模的增加, 其优势也越发明显, 如只有一种缓冲器的缓冲器库, 改进算法耗时为VG算法的73.28%, 当有8种和20种缓冲器的缓冲器库时, 耗时分别为VG算法的67.34%和63.05%。采用本文中的快速缓冲器插入算法, 能有效缩短基于缓冲器插入的大规模互连时序优化时间。

关键词: 片上互连; 缓冲器插入; 时序优化

中图分类号: TN47

文献标识码: A

doi: 10.11805/TKYDA201306.1000

Fast interconnect on chip buffer insertion timing optimization analysis

WANG Xin-sheng^{a,b}, LIU Xing-chun^b, YU Ming-yan^{a,b}

(a.School of Astronautics, Harbin Heilongjiang 150001, China; b.School of Information and Electrical Engineering, Weihai Shandong 264209, China; Harbin Institute of Technology)

Abstract: Along with large scale integrated circuit development, interconnect delay has become the bottleneck of restricting its performance. Buffer insertion is one of the ways to solve interconnect delay problem. VG algorithm is the classical algorithm of buffer insertion interconnection sequence optimization. The improvement is performed based on the three main operation processes of VG algorithm in this paper. Red-black tree data structure is employed to store and route topology data structure, therefore speed up the update and access of the data structure; fast redundancy discrimination and sorting method are adopted to decrease the quantity and the complexity of the optimal solution. The circuits included in standard test circuit International Symposium of Circuits and Systems in 1989 (ISCAS89) set are used to test the proposed method, and the test results show that although the running time of both the improved method and the traditional method is linearly increasing with the increasing of circuit scale, the advantages of the improved algorithm are still more obvious than traditional method. Meanwhile, the advantages are also more significant along with the buffer numbers increasing in the buffer library. In the case of one buffer in the buffer library, the running time of improved algorithm is 73.28% of VG algorithm, while in the case of 8 buffers and 20 buffers in the buffer library, the running time of improved algorithm is 67.34% and 63.05% of VG, respectively. The proposed method can effectively reduce buffer insertion time in the mass interconnection sequence optimization.

Key words: interconnection on chip; buffer insertion; timing optimization

随着大规模集成电路工艺尺寸的减小, 单个芯片上集成度的增加, 门延时越来越短, 而互连线的延时越来越长, 这在国际半导体技术路线图中有很好的诠释^[1]。另外, 在国际半导体技术路线图中还指出, 当特征尺寸减小到0.18 μm 以下时, 互连线延时可以占据总延时的50%以上。因此, 当今互连线延时已经成为约束大规模集成电

收稿日期: 2012-11-14; 修回日期: 2012-12-11

基金项目: 国家科技重大专项基金资助项目(2009ZX03003-005-01)

路性能的主要瓶颈,互连线的延时优化显得特别重要。

在互连线延时优化方面,主要有缓冲器插入^[2]、电流模^[3]和低摆幅^[4],其中缓冲器插入是应用最为广泛的优化方法。同时,文献[5]指出 65 nm 工艺下,大约 12%的局部互连线插入缓冲器,模块之间通信所需要的标准单元中,大约 15%插入缓冲器;进入 32 nm 的工艺,这 2 个方面的数据分别变为 29%和 70%。由此可见,在互连时序优化中缓冲器插入方法的重要性越来越大,复杂度越来越高,寻找更快更优的缓冲器插入方法变得十分必要。缓冲器插入优化最经典的算法是 VG 算法^[6],该算法主要包括 3 个主要操作:1) 增加线;2) 增加缓冲器;3) 冗余检查。Shi 等人针对 VG 算法提出了改进算法^[7],主要包括 4 个方面:1) 有效数据存储结构;2) 缓冲器占优判断准则;3) 快速冗余检查;4) 快速融合算法。Li 等人在此基础上又提出了一种新的记账式方法进一步加快判断缓冲器插入的优化^[8]。同时也出现了其他针对互连线延时和功耗优化的方法^[9-11]。

本文在 VG 算法基础上,针对其操作的 3 个主要过程,利用红黑树数据结构存储路由拓扑数据结构,利用快速冗余判别和排序方法减小冗余解的数量和寻找最优的复杂度,最后对其进行了实现,同时通过标准测试电路集 ISCAS89 对其进行了测试。

1 问题描述

本文采用二叉树形式来表示要优化的互连电路,如图 1 所示,将电路记为 $T=(V,E)$,其中 V 指电路中的标准单元或者互连线的交叉点; E 指电路中 2 个节点间的互连线。定义每个节点的驱动电容为 C ,要求的到达时间为 Q ,同时定义缓冲器参数 R_{bi}, C_{bi} 和 D_{bi} ,这 3 个参数分别为缓冲器的输出电阻、输入电容和固有延时,而互连线上的电阻和电容定义为 $R(e)$ 和 $C(e)$ 。

VG 算法主要过程就是添加互连线,添加缓冲器和冗余判断。添加互连线过程,假设在子树 $T(V_1)$ 驱动点 V_1 处插入一段互连线 $E=(V,V_1)$,形成一个新树 $T(V)$,如图 2 所示。那么对应于 $T(V_1)$ 的每一个候选解方案 α_i , $T(V)$ 都会产生一个新的解方案, C 和 Q 的计算如式(1)和式(2)所示。

$$C(V, \alpha_i) = C(V_1, \alpha_i) + C(e) \quad (1)$$

$$Q(V, \alpha_i) = Q(V_1, \alpha_i) - R(e)C(V_1, \alpha_i) - R(e)C(e) / 2 \quad (2)$$

当在树的根节点 V 处插入一个缓冲器 B 时,对应于 $T(V)$ 的每个候选解方案 α_i , 变为新的解方案,同时 C 和 Q 的计算如式(3)和式(4)所示。

$$C(V) = C(B) \quad (3)$$

$$Q(V, \alpha_i) = Q(V_1, \alpha_i) - K(B) - R(B) \times (C(e) + C(V_1, \alpha_i)) - R(e)C(V_1, \alpha_i) - R(e)C(2) / 2 \quad (4)$$

冗余检查指 2 个候选的解方案 α_1 和 α_2 , 如果 $C(V, \alpha_1) \leq C(V, \alpha_2)$, 并且 $Q(V, \alpha_1) \geq Q(V, \alpha_2)$, 则说明候选方案 α_1 优于候选方案 α_2 , 即方案 α_2 是冗余,需要剔除。

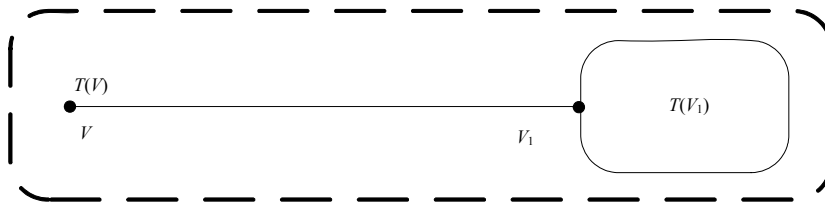


Fig.2 Adding interconnect subtree

图2 互连线子树添加

2 关键技术

2.1 红黑树技术

本文采用红黑树来保存候选解方案列表,这里称为候选树。红黑树实际上是一种特殊的二叉搜索树,它在传统二叉树数据结构中增加一个颜色域来表征这个点是红色节点还是黑色节点。正常情况下查找、插入和删除操作

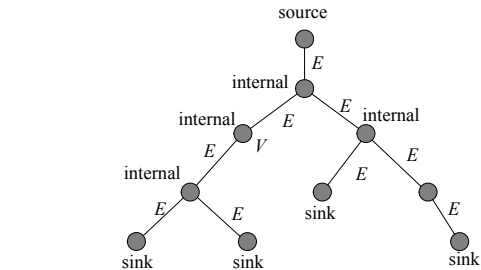


Fig.1 Binary tree representation of integrated circuit interconnection

图1 集成电路中互连电路的二叉树表示形式

的时间复杂度均为 $O(\log_2 n)$ 。红黑树使用颜色域的约束,使最坏情况下的时间复杂度仍为 $O(\log_2 n)$,其证明参见文献[12]。

2.2 快速冗余判别技术

本文把冗余判别技术分为 2 类:1) 基于缓冲器判别技术;2) 基于线的判别技术。基于缓冲器的判别技术第一次被 Shi 等人提出^[7],其定义为:对于树 $T(V)$ 的任意候选解方案 α_1 和 α_2 ,如果 $Q(V, \alpha_1) \geq Q(V, \alpha_2)$,并且 $C(V, \alpha_1) \leq C(V, \alpha_2)$,则 α_1 基于缓冲器判别优于 α_2 ,那么 α_2 就是冗余解方案,限于篇幅,在此不予证明,详见文献[7]。基于线的判别技术,主要运用在每次候选树解树添加线的过程中,直接利用式(1)和(2)进行判断,直接继承了 VG 算法的冗余判别技术。

算法:快速互连时序优化算法
 1、输入待优化的电路;
 2、生成 $T(V)$ 所有非冗余候选解树 $A(V)$
 3、开始优化:
 1) 如果 V 是终端节点,更新候选解树 $A(V)$ 以保存 $T(V)$ 的唯一解,并返回 $A(V)$;
 2) 如果 V 不是终端节点
 a) 如果树 $T(V)$ 由树 $T(V_1)$ 和互连线 (V, V_1) 构成,对 $A(V_1)$ 中的节点进行修改,使得其包含互连线 (V, V_1) 的信息以生成树 $A(V)$ 。然后删除冗余的解方案,并返回 $A(V)$;
 b) 如果树 $T(V)$ 由树 $T(V_1)$ 和缓冲器插入点 V 构成,在 $A(V_1)$ 中搜索具有最大的 $Q(V_1, \alpha)$,然后生成新的候选解方案,并插入到候选 $A(V_1)$,删除冗余的解方案,生成新的候选树 $A(V)$,返回 $A(V)$;
 4、是否达到源端,否返回 3;
 5、结束

Fig.3 Fast interconnect timing optimization algorithm

图 3 快速互连时序优化算法

3 算法实现

该算法的输入是所有节点的位置信息、节点之间的连接关系、互连终端要求的到达时间和负载,将这些信息读入到基于红黑树的二叉树 $T(V, E)$ 中。二叉树生成之后,从互连终端开始自底向上构造非冗余候选树 $A(V)$,同时对候选树进行比较排序,所以树生成过程也就是优化的缓冲器插入过程。算法具体实现如图 3 所示。

上述算法在 $T(V)$ 由左右 2 棵子树 $T(V_1)$ 和 $T(V_2)$ 构成时,假设 $T(V_2)$ 的候选解方案个数大于或等于 $T(V_1)$ 的候选解方案,先计算 $T(V_1)$ 决定的候选解方案 $A(V_1)$,接着计算 $T(V_2)$ 决定的候选解方案 $A(V_2)$,最后把 $A(V_1)$ 插入到 $A(V_2)$ 中,删除冗余的解方案,最后得到树 $T(V)$ 的候选解方案树 $A(V)$ 。

4 仿真结果

本文用 C 语言实现了 VG 算法和快速互连时序优化算法,并将这 2 种算法应用于不同大小的电路优化,使用的计算机平台为 Dell Poweredge T610 服务器, CPU 是 Intel(R) Xeon(R),主频 2.4 GHz, 96 GB 内存。使用 3 种缓冲器库文件 Blib1, Blib8, Blib20 来评估算法的性能,它们分别包含 1 个、8 个和 20 个不同尺寸和驱动能力的缓冲器,表 1 列出了其中 8 种缓冲器的参数,为了简便,其他 12 种缓冲器参数在此没有列出,并且每种缓冲器的固有延时均设定为 66.36 ps,相应电容和电阻参数值来源于工业 180 nm 工艺典型参数,在保证 $R \times C$ 乘积不变的条件下来构造得来。

表 1 仿真用缓冲器参数

Table1 Simulation buffer parameters

buffer type	1	2	3	4	5	6	7	8
input resistance/ Ω	1 120	980	840	700	560	420	280	140
input capacitance/fF	7.89	9.01	10.52	12.62	15.78	21.03	31.55	63.10

表 2 ISCAS89 标准测试电路互连线条数和候选缓冲器插入点数

Table2 Numbers of interconnect wires and candidate buffer insertion points based ISCAS89

test circuit	s27	s1196	s5278	s9234	s13207	s15850	s35932	s38584
number of interconnect	22	1 041	4 440	8 221	11 955	14 329	30 317	34 486
insertion point number of candidate buffer	910	41 830	177 680	328 910	479 520	574 160	1 220 670	1 380 420

测试电路采用的是标准测试电路集 ISCAS89,根据电路的规模选用的电路为 s27, s1196, s5278, s9234, s13207, s15850, s35932 和 s38584。表 2 中列出了所用的标准测试电路中互连线条数和候选缓冲器插入点数。

图 4 给出了 VG 算法和本文方法在使用 Blib1, Blib8 和 Blib20 缓冲器库时的优化运行时间情况,从图中可以

看出随着电路规模的增加,新算法相对于 VG 算法的运算速度优势越来越大。

表 3 给出在使用不同大小的缓冲器库时,新算法与 VG 算法运行时间之比。可以看出新算法相对 VG 算法的运算速度优势随着缓冲器库中缓冲器种类的增加而增大,说明本算法对多种缓冲器插入更为有效。同时,在电路规模相同时,缓冲器种类为 8 的新算法运行时间为缓冲器种类为 1 的 20 倍左右,而当缓冲器种类为 20 时,此数字扩大到接近 100 倍。

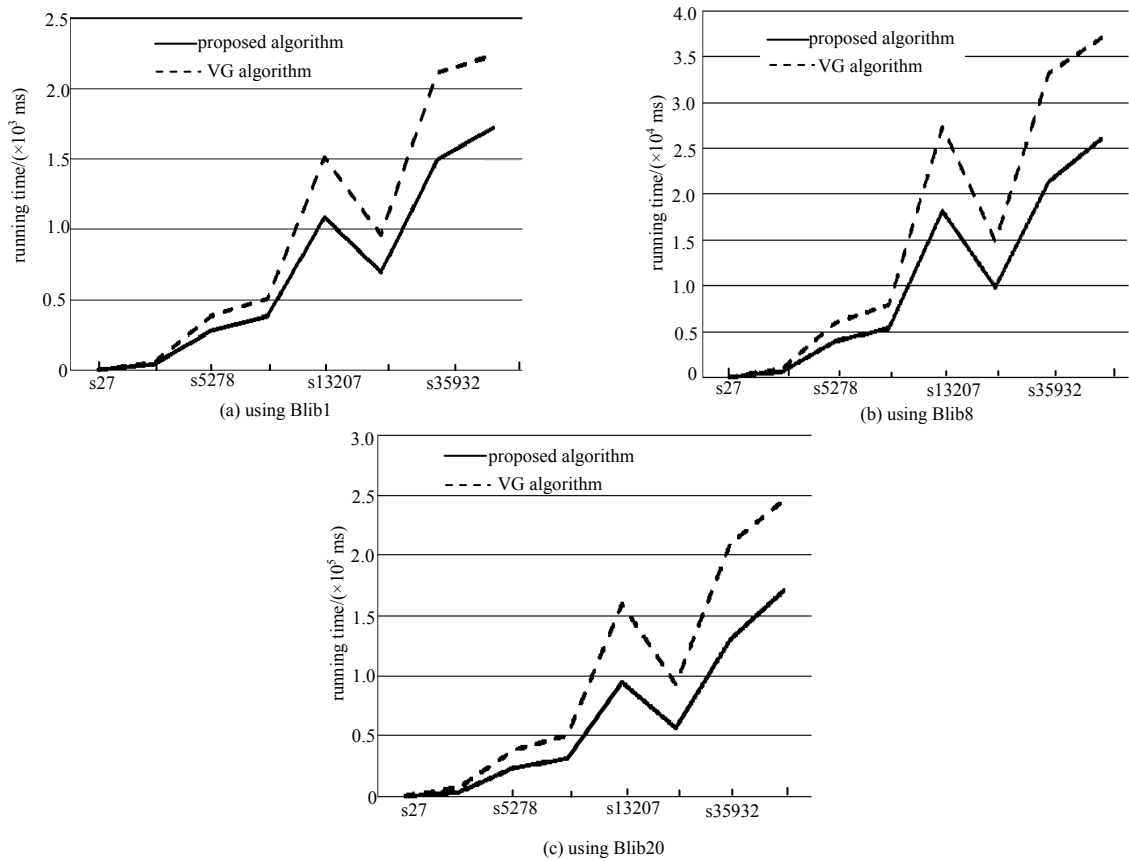


Fig.4 Comparison of running time between proposed algorithm and VG
图 4 新算法和 VG 算法运行时间对比

表 3 新算法与 VG 算法对不同规模的电路运行时间之比

electric circuit	bufferlib1/(%)	bufferlib8/(%)	bufferlib20/(%)
s27	74.02	70.68	64.17
s1196	73.25	67.59	61.45
s5278	71.39	65.94	63.38
s9234	75.42	67.80	62.72
s13207	71.87	66.43	59.63
s15850	72.32	65.81	61.38
s35932	70.92	64.12	62.09
s38584	77.05	70.35	69.60
average ratio	73.28	67.34	63.05

5 结论

本文针对片上互连的缓冲器插入时序优化进行研究,在 VG 算法的基础上,针对其 3 个操作过程的特点,使用红黑树数据结构存储候选解方案,并且把缓冲器插入冗余判别和互连线插入冗余判别分开。通过 ISCAS89 标准测试电路仿真测试,结果表明新算法相对 VG 算法的运算速度优势随着电路规模的增加而增大,同时也随着缓冲器库中缓冲器种类的增加而增大,当缓冲器库中只有一种缓冲器可选时,新算法耗时大约为 VG 算法的 73.28%,而当缓冲器库中缓冲器可选数目分别为 8 时和 20 时,新算法运行时间大约是 VG 算法运行时间的 67.34%和 63.05%。

参考文献:

- [1] Semiconductor Industry Association. International Technology Roadmap for semiconductors[EB/OL]. [2007-01]. <http://www.itrs.net/Links/2007ITRS/Home2007.htm>
- [2] Chen J Y, Jone W B, Wang J S, et al. Segmented bus design for low-power systems[C]// 12th IEEE Workshop on Signal Propagation on Interconnects. Avignon:IEEE, 2008:25-29.
- [3] Mohammad M T, Nasser M. Low-power and high-performance techniques in global interconnect signaling[J]. Microelectronics Journal, 2009,40(10):1487-1495.
- [4] Zhang H, Rabaey J. Low-Swing Interconnect interface Circuits[C]// International Symposium on Low Power Electronics and Design(ISLPED'98). CA, USA:[s.n.], 1998:161-166.
- [5] Osler P J. Placement Driven Synthesis Case Studies on Two Sets of Two Chips: Hierarchical and Flat[C]// Proceedings of International Symposium on Physical Design. San Diego:IEEE, 2004:190-197.
- [6] Van Ginneken L P P. Buffer Placement in Distributed RC-tree Networks for minimal Elmore delay[C]// Proceedings of Int. Symp. Circuits and Systems. New Orleans:IEEE, 1990:865-868.
- [7] Shi W, Li Z. An $O(n \log n)$ Time Algorithm for Optimal Buffer Insertion[C]// Design Automatic Conference(DAC'2003). San Diego:IEEE, 2003:580-585.
- [8] Li Z, Shi W. $O(mn)$ Time Algorithm for Optimal Buffer Insertion of Nets with m Sinks[J]. IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, 2012,31(4):427-441.
- [9] Narasimhan A, Sridhar R. Variability Aware Low-Power Delay Optimal Buffer Insertion for Global Interconnect[J]. IEEE Trans. on Circuits and Sys. I, 2010,57(12):3055-3063.
- [10] 张妮娜, 窦衡. 并行化改进遗传算法的 FPGA 高速实现方法[J]. 太赫兹科学与电子信息学报, 2012,10(1):107-113. (ZHANG Nina, DOU Heng. Methodology of realizing FPGA for improved parallel genetic algorithm[J]. Journal of Terahertz Science and Electronic Information Technology, 2012,10(1):107-113.)
- [11] 李海军, 王玉萍, 黄耀群. 一种基于 FPGA 节省资源实现 FIR 滤波器的设计方法[J]. 太赫兹科学与电子信息学报, 2010,8(4):455-458. (LI Haijun, WANG Yuping, HUANG Yaoqun. A method of resources saving to realize FIR filter based on FPGA[J]. Journal of Terahertz Science and Electronic Information Technology, 2010,8(4):455-458.)
- [12] 潘金贵, 顾铁成, 李成法, 等. 算法导论[M]. 北京:机械工业出版社, 2006. (PAN Jingui, GU Tiecheng, LI Chengfa, et al. Introduction to Algorithms[M]. Beijing:China Machine Press, 2006.)

作者简介:



王新胜(1978-), 男, 山东省威海市人, 博士, 主要研究方向为大规模集成电路可靠性.
email:xswang@hit.edu.cn.

刘兴春(1989-), 男, 山东省潍坊市人, 硕士, 主要研究方向为大规模集成电路设计.

喻明艳(1964-), 男, 杭州市人, 教授, 主要研究方向为多核处理器架构和并行编程.

勘 误

《太赫兹科学与电子信息学报》2013 年第 5 期《能散度与发射度模块在太赫兹 FWTWT 模拟中的应用》(作者:董烨,等)基金项目之一“国家自然科学基金资助项目(1130501)”,更改为“国家自然科学基金资助项目(11305015)”。

由此带来的不便,敬请谅解!

《太赫兹科学与电子信息学报》编辑部
2013 年 12 月 10 日