

带时延约束的 FPGA 布线算法*

周 锋 童家榕 唐璞山

(复旦大学电子工程系 CAD 研究室 上海 200433)

摘要 基于 SRAM 编程结构的门海型 FPGA 连线上的时延较之 ASIC 来说比较大, 连线延迟不可预测. 在很多应用中必须对关键路径的时延加以定量限制(包括上限、下限和一组路径的时延差). 时延约束的实现需要布图算法来保证. 一般时延驱动的布线算法只能定性优化时延性能, 不能满足定量要求. 本文提出了高性能 FPGA 最短路径布线算法, 以它为主体的 FPGA 布线器能全面地考虑各种时延约束, 更好地利用布线资源, 对其它无时延约束的线网也可进行时延优化, 提高整个芯片的性能.

EEACC: 1130B

1 引言

FPGA 有多种编程结构^[1-10], 基于 SRAM 编程方式的门海型 FPGA (图 1) 中, SRAM 编程结构采用 MOS 晶体管作编程开关, 晶体管有上千欧姆的导通电阻, 由这些可编程晶体管串接构成的连线会有十几, 甚至几十纳秒的信号延迟, 使 FPGA 的工作频率受到很大限制. 所以, 时延驱动的布图算法对于要求较高的 FPGA 设计就成为必然. 然而, 仅有时延驱动对 FPGA 来说是不够的. FPGA 的信号延迟的不可预测性会导致布图以后某些信号不能在预定时刻到达, 引起错误的逻辑动作或产生严重干扰, 电路性能指标降低甚至无法正常工作. 一些应用场合例如 PCI 总线接口、数字信号处理等, 直接就对信号时延提出定量的要求. 很多情况下设计者必须对某些信号路径加上具体的、定量的时延约束, 不仅有时延上限的约束, 还可能有时延下限约束(例如, 为保证多个信号同时到达而有意延迟某些信号), 还可能有一组路径的时延之差的约束(例如 DataPath 电路). 时延驱动布图算法只能定性优化时延性能, 无法保证用户提出的定量要求得到

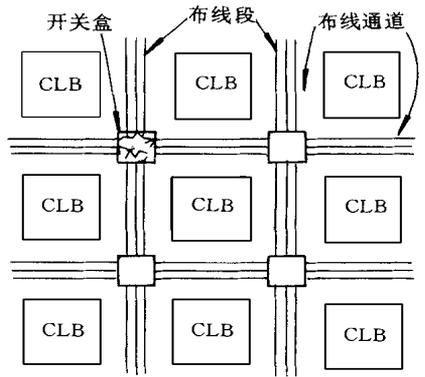


图 1 FPGA 结构

* 本课题得到国家“九五”重点科技攻关项目(96-738-01-09-01)资助

周 锋 男, 1966 年出生, 博士生. 从事集成电路计算机辅助设计的研究

童家榕 男, 1942 年出生, 教授, 博士生导师. 从事集成电路计算机辅助设计的研究

唐璞山 男, 1934 年出生, 教授, 博士生导师. 从事集成电路计算机辅助设计的研究

1998-03-02 收到, 1998-06-19 定稿

满足 本文提出的布线算法——RTC (Routing Algorithm with Time Constraints) 是考虑时延约束的 FPGA 布线算法 它能够满足用户提出的关键路径时延的各种定量要求, 包括时延上限、下限和时延差约束 同时对整个电路作资源利用率和性能的优化

2 时延约束对布线器的要求

我们提出“最短路径布线方案”来实现时延约束 布局阶段^[11]在优化布局的同时, 以线网的最短路径布线假设来预测关键路径时延, 使之满足相应的时延约束 这样, 布线阶段要保证关键路径上的线网以最短路径布线, 关键路径的时延要求才能得到满足 其它非关键线网优先用最短路径布线算法来布, 以缩短整个电路的平均时延, 提高性能

我们的研究表明, 在合理的模型下, FPGA 中一条连线的 Elmore 时延^[4]与它包含的编程开关数目之间可以用线性关系来描述 我们利用这一关系估算 FPGA 连线时延, 很好地兼顾了精度与算法复杂度之间的矛盾 根据具体的 FPGA 工艺参数, 对关键路径时延约束可以转化成连线对所含编程开关数目的约束, 便于布图算法处理

3 RTC 算法

FPGA 布线算法可分为两大类: 基于通道的布线算法和基于线网的布线算法

基于线网的布线算法^[5, 6, 8]以线网为单位布线, 以迷宫算法(因为 FPGA 阵列不大, 耗时可以接受)为基础 一般迷宫算法以找到路径为主要目标; 在 FPGA 中要考虑众多线网对有限的布线资源的需求竞争, 在许多可行的路径中选取最好的方案才是主要目标 这里有两个值得研究的问题: 第一, 能搜索到更多可供选择的路径方案无疑可以提高优化程度, 但算法复杂度也将随之提高; 第二, 选择路径的准则将直接影响布线的质量, 价格函数应尽可能准确地估计未布线网的走向, 结合当前资源情况, 作出合理的判断, 使后布的线网容易布通 这些是 FPGA 迷宫算法不同于普通迷宫算法之处 此外, 基于线网的布线算法都无法避免布线顺序对布线质量的影响, 可以单独或并用以下方法提高性能: 一, 布线之前为线网排序, 精心设计排序算法; 二, 应用“拆线重布”算法, 通过几个循环的重布使结果收敛于较优解

为了说明 RTC 算法, 首先定义一些概念

[定义 1] 如果某线网 (Net k) 以最短路径布线时在某含有 n 个通道的通道集 $\{c_1, c_2, \dots, c_n\}$ 中必占据其中一个, 那么就定义线网 k 对其中任意一个通道 (c_i) 的“需求程度”—— $ND(k, c_i) = 1/n$. 如果某通道 c_i 同时处于两个以上这样的通道集中, 则取其在各通道集中的 $ND(k, c_i)$ 中最大值为线网 k 对通道 c_i 的“需求程度”; 如果某通道 c_i 不处于任何一个这样的通道集中, 则 $ND(k, c_i) = 0$

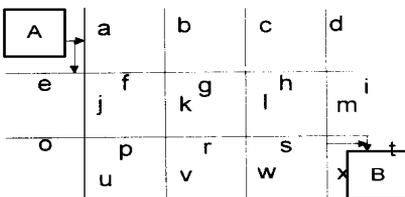


图 2 从 CLB A 到 CLB B 的连线可能经过的通道

例如, 图 2 中有线网 (记为 K) 从 CLB A 进通道 a 或通道 e , 最后从通道 t 或通道 x 进 CLB B. 这样通道 a 和通道 e 必用其一, $ND(K, a) = ND(K, e) = 1/2$; 同理 $ND(K, t) = ND(K, x) = 1/2$

如果按最短路径布线, 线网 K 只可能布在 $f, g, h, j, k, l, m, p, r, s$ 这几个通道内, 且不能重复 例如可走 $a-f-k-r-s-t$ 或 $e-f-g-h-m-x$ 等 无论怎样走, j, k, l, m 必居其一, $ND(K, j) = ND(K, k) = ND(K, l) = ND(K, m) = 1/4$; 而 $ND(K, f) = ND(K, g) = ND$

$$(K, h) = ND(K, p) = ND(K, r) = ND(K, s) = 1/2$$

[定义 2]如果某线网(Net k)布线时在某通道中有 n 条布线段(segment)可供选择,则定义线网 k 对其中任意一条布线段(s_i)的“需求程度”—— $NDS(k, s_i) = 1/n$

例如图 3 中 CLB A 的输出端子在通道 a 中有两个选择: 布线段 1 或布线段 3 则 $NDS(k, a1) = NDS(k, a3) = 1/2$

RTC 属基于线网的布线算法,以二端线网(由多端线网拆分而得)为单位依次布线 最短路径布线器是实现时延约束条件和提高布线时延性能的关键,它搜索出正在布的二端线网所有可能的用最短路实现方案,为每一种方案计算价格函数,取价格最小者为最后实现方案 本算法首先使用最短路径布线器布关键线网,然后用最短路径布线器布其它线网 最后用“改进 FPGA 迷宫算法”布仍未布通的线网 “改进 FPGA 迷宫算法”和最短路径布线器工作原理相似,不同之处是“改进 FPGA 迷宫算法”不局限于最短路径实现的方案,以找到路径为主要目标

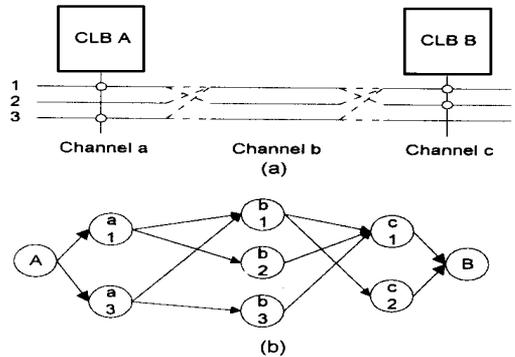


图 3 根据布线方案建立搜索图

路径方案的价格为路径包含的布线段的价格之和:

$$C_{Candidate_i} = \sum_{j=1}^{NS} C_{s_j}$$

其中 $C_{candidate_i}$ 是第 i 种方案的价格, NS 是路径包含的布线段数目, s_j 是路径中第 j 条布线段(segment). 例如上述图 3 中的例子 $NS = 3$ 在 $a1\ b1\ c2$ 的方案下, $a1, b1, c2$ 分别是 s_1, s_2, s_3 C_{s_j} 是每一个布线段 s_j 的价格:

$$C_{s_j} = CN_{s_j} + CD_{s_j}$$

第一项反映尚未布的线网对布线段 s_j 的潜在需求

$$CN_{s_j} = \sum_{k=1}^{Nu} (NDS(k, s_j) \times ND(k, c_{s_j}))$$

Nu 是未布二端线网的数目, $NDS(k, s_j)$ 是第 k 个未布二端线网对布线段 s_j 的需求程度, $ND(k, c_{s_j})$ 是第 k 个未布二端线网对布线段 s_j 所在通道 c_{s_j} 的需求程度

第二项反映布线段 s_j 所在的通道的拥挤程度

$$CD_{s_j} = N_r(c_{s_j}) + 1 + E[N_{exp}(c_{s_j})]$$

$N_r(c_{s_j})$ 代表布线段 s_j 所在通道 c_{s_j} 中已布的线网数; 加上 1 代表正在布的这个线网 $N_{exp}(c_{s_j})$ 是将要布在这个通道内的线网数 当前来说这是一个随机数, 求它的数学期望作估值 记 $E[x]$ 为 x 的数学期望

$$N_{exp}(c_{s_j}) = \sum_{k=1}^{Nu} N_{exp}(k, c_{s_j})$$

$N_{exp}(k, c_{s_j})$ 表示第 k 个未布线网将占用这个通道内布线段的数目, 也是随机数, 实际上只可能有两个取值: 0 或是 1, 服从零-壹分布 它为 1 (即布在此通道内) 的概率是 $ND(k, c_{s_j})$. 于是我们有:

$$E[N_{\text{exp}}(k, c_s, j)] = ND(k, c_s, j) \times 1 + (1 - ND(k, c_s, j)) \times 0 = ND(k, c_s, j)$$

根据概率理论, 可得:

$$E[N_{\text{exp}}(c_s, j)] = \sum_{k=1}^{Nu} E[N_{\text{exp}}(k, c_s, j)] = \sum_{k=1}^{Nu} ND(k, c_s, j)$$

对布线段 c_s 来说, 其它未布线网对它的潜在需求越大, 或它所处的通道越拥挤, 它的价格就越高, 它被取用的可能性越小

RTC 搜索全部可能的实现方案, 这在一般情况下是 NP 问题. 以简单的正方形布线区域为例 (图 4(a)), 图中边代表布线通道, 交点代表开关盒. 从左上角开始到右下角止以最短路径布线, 到一半路径长度 (记为 L) 时 (斜线上各点) 已有 2^L 种方案 (图 4(b)), 每个节点代表图 4(a) 中一个交点, 节点内数字是交点坐标, 最终仅布线通道的方案数就是 $O(2^L)$. 这里还没有计入每个布线通道内可供选择的布线段数目, 一个通道内不只有一个布线段

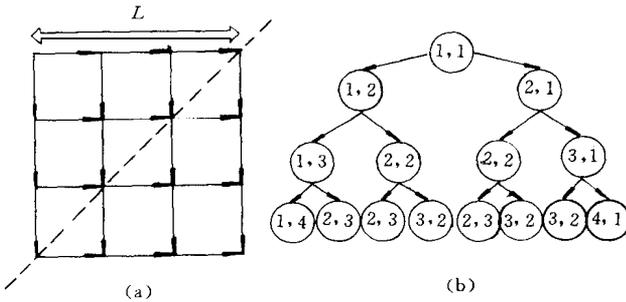


图 4 全搜索方案的搜索路径

我们发现不同的方案之间有许多重复路径. 例如图 3 中的例子: (不失一般性, 这里把问题简化在一个布线通道方案内, 有多个布线通道方案的情况可简单推广). 图 3(a) 中即便只有一个布线通道方案, 考虑通道内的布线段, 有 $a_1, b_1, c_2, a_3, b_3, c_1$ 等 6 种实现方案. 然而在 a_1, b_1, c_2 和 a_3, b_1, c_2 两种方案中 b_1, c_2 部分是重复路径. 实际上在搜索路径时常碰到已搜索过的路径, 为了利用前面

计算的结果避免重复计算, 我们建立“搜索图”. 搜索图中源结点和目标结点分别表示二端线网的起始和终止 CLB, 其它每一个结点表示一个布线段, 结点之间的边表示从边两端所接的布线段之间可以用编程开关相连, 边的方向表示允许的搜索方向 (规定边的方向在最短路径布线器中是必须的). 如图 3(b) 即是图 3(a) 构造的搜索图. 我们对搜索图作深度优先搜索来代替迷宫搜索.

当一个结点被搜索过以后, 从这个结点开始往后直到终点的各方案中价格最小的路径及其价格就已经求出了, 我们把这些信息记录在这个结点中. 这样, 当从其它路径搜索再次到达这个结点时, 就可直接利用记录, 不必重新计算, 极大地降低时间复杂度. 例如图 3 中从 b_1 开始往后有 b_1, c_1 和 b_1, c_2 两种方案, 搜索完 b_1 及其后面的子图后, 从 b_1 开始往后的最小价格方案已经求出, 把这个方案及其价格记录在 b_1 结点里. 以后从 a_3 结点搜索再次到达 b_1 时, 不必往下搜索, 直接读取最小方案价格加入运算即可. 事实上, c_1 和 c_2 的价格在 b_1 之前就已算出并存储在这两个结点中, 以后从 b_2 搜索到 c_1 或 c_2 时, 可直接利用结果. 图的深度优先搜索算法复杂度为 $O(e+n)$, e 是图中的弧数目, 这里是牵涉到的编程开关数目; n 是图中结点的数目, 这里代表搜索到的布线段总数.

“改进 FPGA 迷宫算法”解决最短路径布线器无法布通的线网. 它逐步放宽路径长度的限制, 例如, 放宽至最短长度的 2 倍、8 倍. 仍然限定长度以控制算法复杂度. 实验表明, 超过 80% 的遗留线网能在 2 倍最短长度内布通; 而如果不控制算法复杂度, 布通同样线网将增加

耗时数十倍

线网预排序算法决定布线顺序。排序算法计算二端线网的排序价格, 线网依排序价格从小到大排序, 排序价格小的线网先布

$$\text{Corder}(i) = M \ln \text{Length}(i) + \alpha \times \text{Scale}(i)$$

式中 $M \ln \text{Length}(i)$ 是线网 i 的最短路径布线长度。长度短的线网先布, 避免短线绕长路, 提高时延性能; 而对长线网来说, 由于跨度大, 可选择的方案更多。所以这样做有利于提高布通率。Scale(i) 是二端线网 i 所在的多端线网的端子数目。Scale(i) 大的线网分布广, 有较多的等效连接点。同属这一多端线网的后布二端线网只需接到这些线网上任意一点, 所以较容易布通。α 是权重因子。

4 结果

带时延约束的布线算法需和带时延约束的布局算法(我们开发的另一软件 PTC) 配合使用。取 MCNC Partition93 中四个例子, 加上时延约束进行测试。工艺映射(Mapping) 和路径时延计算利用 Xilinx 公司软件 XACT 5.0 完成。布局布线分别采用 PTC-RTC 和 XACT5.0 完成以作比较(XACT5.0 不能考虑路径之间的时延差约束)。结果比较见下表, 其中“MAX”列指整个芯片中最长的路径时延; $\Delta(d1, d2, \dots)$ 指路径 $d1, d2, \dots$ 之间的时延差。XC3000 资源条件下布线, 布线通道容量为 5。RTC 在 SUN SPARC_5 上运行, 用时不超过 1 分钟。

关键路径	s510xc3x						c432xc3x						
	a	b	c	d	$\Delta(b, c, d)$	MAX	a	b	c	d	$\Delta(a, b)$	$\Delta(a, b, c, d)$	MAX
关键线网数	6	5	5	4		—	2	3	5	5			—
时延约束值/ns	< 70				< 10	—					< 10	< 20	—
RTC 结果/ns	63	56	60	53	7	75	18	25	32	23	7	14	84
XACT5.0/ns	66	56	37	67	30	79	66	31	56	36	35	35	91

关键路径	c499xc3x						s420xc3x									
	a	b	c	d	$\Delta(a, b, c, d)$	MAX	a	b	c	d	e	f	g	h	$\Delta(a \sim h)$	MAX
含关键线网数	4	4	5	5		—	3	3	3	3	3	3	3	3		—
时延约束值/ns					< 4	—									< 10	—
RTC 结果/ns	31	31	33	33	2	84	16	18	23	24	17	21	15	19	9	95
XACT5.0/ns	10	25	14	36	26	86	7	28	18	10	22	13	12	20	21	96

5 结论

RTC 是一个能非常有效地实现用户提出的各种时延要求的 FPGA 布线算法。它先进的全搜索算法和价格函数的设计能更好地考虑后面未布线网的需求和走向, 提高布线质量——包括时延性能和布通率。同时, 它是一个定通道容量的算法, 符合 FPGA 结构的实际情况, 具有实用前景。

参 考 文 献

- [1] Sudip K. Nag and Rob A. Rutenbar, "Performance-Driven Simultaneous Place and Route for Island-Style FPGAs", The 1995 International Conference on CAD, 332~ 338
- [2] Tanay Karnik and Sung-Mo Kang, "An Empirical Model for Accurate Estimation of Routing Delay in FPGAs", The 1995 International Conference on CAD, 328~ 331.
- [3] Jon Frankle, "Iterative and Adaptive Slack Allocation for Performance-driven Layout and FPGA Routing", 29th ACM/IEEE Design Automation Conference 1992, 536~ 542
- [4] W. C. Elmore, J. Appl Phys, 1948, **19**(1), 55~ 63
- [5] Yu-Sheng Lee *et al*, IEEE Trans Comput Aided Des Integr Circuits Syst, 1997, **16**(2): 179~ 185
- [6] Yu Sun *et al*, IEEE Trans Comput Aided Des Integr Circuits Syst, 1997, **16**(1): 486~ 627.
- [7] Stephen Brown *et al*, IEEE Trans Comput Aided Des Integr Circuits Syst, 1992, **11**(5): 620~ 627.
- [8] Amit Chowdhary *et al*, "Detailed Routing of Multi-Terminal Nets in FPGAs", 7th International Conference on VLSI Design, January 1994, 237~ 242
- [9] Yuliang Wu *et al*, IEEE Trans Comput Aided Des Integr Circuits Syst, 1996, **15**: 33~ 34
- [10] S. Thakur *et al*, IEEE Trans Comput Aided Des Integr Circuits Syst, 1997, **16**(1): 32~ 46
- [11] 周锋, 童家榕, 唐璞山, "带时延约束的FPGA 布局算法", 已被《计算机辅助设计与图形学学报》接收

Routing Algorithm for FPGA With Time Constraints

Zhou Feng, Tong Jiarong, Tang Pushan

(CAD Institute, Electronic Engineering Department, Fudan University, Shanghai 200433)

Received 2 March 1998, revised manuscript received 19 June 1998

Abstract The unpredictable path delay of the Sea-of-gate structure and SRAM-based FPGA makes it necessary to impose time constraints on some paths in order to achieve a successful system design. Combined with the PTC (Placement with Time Constraints), the proposed FPGA routing algorithm-RTC (Routing with Time Constraints) can deal with not only the constraints of path delays, but also the constraints of delay differences among a group of paths. A high performance Shortest Distance Router (SDR) is proposed, in which, with the linear time complexity, all possible shortest-distance routing methods for the routing net are valued and the best solution is finally choosed. Probability theory is adopted in the cost function to predict the final channel density in order to guide the routing more reasonably.

EEACC: 1130B