



(12) 发明专利申请

(10) 申请公布号 CN 102809711 A

(43) 申请公布日 2012. 12. 05

(21) 申请号 201210108006. 1

G01R 31/02(2006. 01)

(22) 申请日 2012. 04. 13

G01R 31/327(2006. 01)

(66) 本国优先权数据

201110392611. 1 2011. 12. 01 CN

(71) 申请人 中国电子科技集团公司第五十八研究所

地址 214035 江苏省无锡市滨湖区惠河路 5 号

(72) 发明人 李晓磊 徐彦峰 于大鑫

(74) 专利代理机构 无锡市大为专利商标事务所 32104

代理人 殷红梅

(51) Int. Cl.

G01R 31/08(2006. 01)

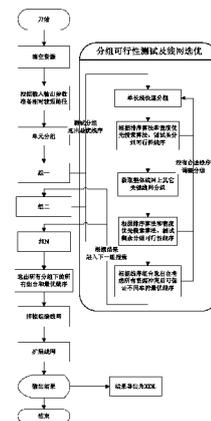
权利要求书 2 页 说明书 10 页 附图 3 页

(54) 发明名称

一种测试 FPGA 单长线及连接开关的扩展布线方法

(57) 摘要

本发明公开了一种测试 FPGA 单长线及连接开关的扩展布线方法,步骤包括:设置轮回行并初始化参数,快速分组,线序预估,二次过滤,线序拼接,回退判定,拆除重布,结果扩展,结果验证。其优点是:依据层次式 FPGA 的结构特点,提出了一种基于布通率的高可靠、高适应性、高效率的布线算法;该算法能快速进行资源配置和有效配置资源图的生成,对单长线及连接开关的资源覆盖率较高,且与具体硬件结构无关;该算法的实施极大地增强了自动化布线和测试能力,有效减少后期布线工作量,节约测试成本,提高测试效率,缩短测试周期,降低开发风险,有一定的应用和参考价值。



1. 一种测试 FPGA 单长线及连接开关的扩展布线方法,其特征是,步骤如下:

步骤一、设置轮回行并初始化参数:确定轮回行的数目为 2,并为后续布线搜索到的链路中间结果准备存储空间,准备完成后进入步骤二;

步骤二、快速分组:依据分组算法依次将 CLB 和单长线进行分组,获取轮回行所有 CLB 分组和第一组 CLB 分组对应单长线起始组合点;分组完成后进入步骤三,如果失败则退出算法,返回失败代码;

步骤三、线序预估:在选取完第一组 CLB 分组连接点后,通过获取轮回行所有关键布线 CLB 分组连接点的方法获取轮回行所有 CLB 分组的起始连接点,然后分别对这些连接点进行测试验证,测试该分组是否合适,是否可以选出最优线序;测试的方法是:依据宽度优先算法搜索并选取该 CLB 布线在最大链长下的所有可布通的线网,如能布通则表示该分组暂时可行,并分别记录 CLB1、CLB2、CLB3、CLB4 可布通的组合序号以便在后续回退时调整该组合序号选出相应线序组合;当预估完成后进入步骤四,否则,如果所有组合测试后均未发现可布通的线序组合则回退至步骤二重新分组;所述 CLB1、CLB2、CLB3、CLB4 分别指每个 CLB 分组的第一个 CLB、第二个 CLB、第三个 CLB 和第四个 CLB;

步骤四、二次过滤:经过步骤 3 预估过滤后,确定了所有关键布线 CLB 分组的起始连接点和所有可行的线序组合后,再做进一步进行过滤排除:从第一个 CLB 分组的 CLB1 和 CLB2 开始,依次选取相邻下个分组的 CLB1 和 CLB2,在选取 CLB1 和 CLB2 的过程中过滤水平六长线的占用,并同时采用线序选优的策略选取最优线序组合,以降低 CLB1 和 CLB2 分别对 CLB4 和 CLB3 布线线网的影响;在选取 CLB1、CLB2、CLB3 和 CLB4 的过程中,考虑相互之间水平六长线和行间单长线的占用影响,当过滤完成后进入步骤五,否则失败回退至步骤二重新分组;

步骤五、线序拼接:在完成所有关键布线 CLB 分组的布线后,需要对相邻关键布线 CLB 分组的布线进行拼接来完成整个线网的布线,在拼接过程中,依据单个 CLB 在线序预估阶段选出的可行线序索引进行,当拼接成功则进入步骤八,否则当某个 CLB 在重试完所有的可行线序索引后均未拼接成功,则进入步骤六进行回退判定;

步骤六、回退判定:在线序拼接失败后,进行布线 CLB 回退位置的选择,首先选择拼接失败 CLB 的顶部 CLB,调整顶部 CLB 选取的线序组合对底部 CLB 布线有影响的占用集合,同时测试底部 CLB 是否可以布通;如果可以则记录该线序索引值,进入步骤七进行拆除重布;如果调整完所有线序索引仍未布通,则回退至该拼接失败 CLB 的上一相邻关键布线 CLB 分组内具有相同组内索引的 CLB,同样重试该 CLB 选出的可行线序索引进行,同时测试是否可以布通,如是,则记录该线序索引值,进入步骤七进行拆除重布,如果重试完所有的可行线序索引后仍未布通,则继续按照上述方法循环回退,直至回退位置为首行的首个分组为止;如果仍未成功则回退至步骤二重新分组;

步骤七、拆除重布:对已布线网进行部分拆除,拆除部分为布线失败 CLB 位置至回退 CLB 位置,拆除这之间所有已布的线网,并在拆除过程中分析造成失败的冲突占用点和线序索引,为回退重布提供依据,拆除完成后根据回退 CLB 位置记录的可行线序索引值进行资源重建即重布,重建完成则回退至步骤五,从该回退 CLB 位置开始继续未完成的线序拼接;

步骤八、结果扩展:完成拼接后的链路即为完整链路,即已完成了轮回行布线,覆盖了轮回行的所有 CLB;此时轮回行以外的 CLB 部分,将搜索到的链路中间结果按照行奇偶对应

的原则扩展至整个芯片,在扩展的过程中严格遵守 XDL 的描述规则;

步骤九、结果验证:扩展完成后,即实现了整个 FPGA 的布线,将此时的中间结果直接导出为 XDL 格式文件,然后利用 XDL 命令,将导出文件转化为 NCD 网表进行结果验证。

2. 如权利要求 1 所述的测试 FPGA 单长线及连接开关的扩展布线方法,其特征在于,步骤二中所述的 CLB 和单长线分组方法,在型号为 XCV1000 的芯片上,其 CLB 分组、CLB 分组大小、单长线分组大小和单长线分组计算方式分别为:

CLB 分组:从公式

(列宽  $96 \times$  轮回行大小  $2$ ) / (CLB 分组大小 + CLB 被测分组大小)

得到分组数目,再根据输入的分组偏移量分别获取所有分组所包含的 CLB 的具体行列值,所有这些分组称为关键布线 CLB 分组,每组又都包含 CLB1、CLB2、CLB3、CLB4 四个 CLB;将第一行的首个 CLB 分组作为第一个 CLB 分组,然后第一行按从左至右,第二行按从右至左的顺序依次将所有分组编号;而组序号从小到大相邻的 CLB 分组中有相同组内序号的 CLB 为起始和结束链路布线关系,第一组 CLB 的 CLB1 为第二组 CLB 的 CLB1 的起始链路布线 CLB,第二组 CLB 的 CLB1 为第一组 CLB 的 CLB1 的结束链路布线 CLB,同时第二组 CLB 的 CLB1 为第三组 CLB 的 CLB1 的起始链路布线 CLB;

CLB 分组大小:单长线数目  $24$  / 输入线序数目  $6 = 4$ ;

单长线分组大小:单长线数目  $24$  / 输入线序数目  $6 = 4$ ;

单长线分组:使用组合算法,将  $24$  根单长线每次选出  $6$  根,分四次选出分成四组,每组包含  $6$  根单长线起始连接点,然后将选出的四个分组编号,当后续检测到失败回退需要重新分组时,将当前分组编号加  $1$  继续选取下个分组;分组后的单长线依次分别对应到首个 CLB 分组的四个 CLB 上,作为相应 CLB 布线起始连接点。

3. 如权利要求 1 所述的测试 FPGA 单长线及连接开关的扩展布线方法,其特征在于,步骤四所述的线序选优的策略是:依据单链长最短、总链长最短、稀缺资源占用最少、单长线垂直底部占用连接点  $S_x$  或单长线垂直顶部占用连接点  $N_x$  最少的原则判定最优;其中,在进行第一行布线时考虑单长线垂直底部占用连接点  $S_x$ ,当进行第二行布线时考虑单长线垂直顶部占用连接点  $N_x$ ;  $x$  为  $0 \sim 23$  的任意整数值。

4. 如权利要求 1 所述的测试 FPGA 单长线及连接开关的扩展布线方法,其特征在于,步骤五所述对相邻关键布线 CLB 分组的布线进行拼接是将第一组的 CLB1、第一组被测 CLB 分组和第二组的 CLB1 进行拼接,第二组的 CLB1、第二组被测 CLB 分组和第三组的 CLB1 相拼接,并依次完成所有相邻 CLB 分组的 CLB1 的拼接,同理完成 CLB2、CLB3 和 CLB4 的拼接。

5. 如权利要求 1 所述的测试 FPGA 单长线及连接开关的扩展布线方法,其特征是,所述线序拼接是将相邻关键布线 CLB 分组的链路头、链路体和链路尾这三个部分依据连接对应关系进行对接,并最终形成完整的布线轨迹。

## 一种测试 FPGA 单长线及连接开关的扩展布线方法

### 技术领域

[0001] 本发明实现了一种测试 FPGA 单长线及连接开关的扩展布线方法,属 FPGA 自动化布线范畴。

### 背景技术

[0002] FPGA 芯片是一种布线资源数量庞大、种类繁多、可重复编程的超大规模集成电路,在实际应用中,发生于互连资源上的故障率远大于其它器件的故障率,那么为了解决发生于互连资源上的故障,尤其为了测试单长线及互连开关的连接情况,并快速有效地对故障进行查找、定位和解决,这里采用了一种 FPGA 单长线及直连开关的测试方法(专利申请号 201110125752.7)。

[0003] 在对该测试方法进行实施验证的过程中发现,由于手工费时费力且错误率高以及机时有限且极其昂贵等系列原因,极需一种更高效的布线算法来进行快速的资源配置和有效配置资源图生成以进行连续不间断的测试。

[0004] 近年来,各国学者提出过许多相关布线算法,其中迷宫算法是一种比较成熟的算法,其它如蚁群算法、全局寻优算法等都有一定的用途和群众基础,然而,这些算法在实施的过程中不是针对性不强,如单长线资源的覆盖率问题,就是理论到实践验证转换有一定难度,可操作性不强,另外有的算法在布线效率上的表现也不是很令人满意。

### 发明内容

[0005] 为了解决现有布线方法的不足、针对性不强以及其它如手工布线效率低下等问题,本发明依据层次式 FPGA 的结构特点,提出了一种测试 FPGA 单长线及连接开关的扩展布线方法,是一种基于布通率的高可靠、高适应、高效率的布线方法。

[0006] 按照本发明提供的技术方案,所述测试 FPGA 单长线及连接开关的扩展布线方法,步骤如下:

步骤一、设置轮回行并初始化参数:确定轮回行的数目为 2,并为后续布线搜索到的链路中间结果准备存储空间,准备完成后进入步骤二;

步骤二、快速分组:依据分组算法依次将 CLB 和单长线进行分组,获取轮回行所有 CLB 分组和第一组 CLB 分组对应单长线起始组合点;分组完成后进入步骤三,如果失败则退出算法,返回失败代码;

步骤三、线序预估:在选取完第一组 CLB 分组连接点后,通过获取轮回行所有关键布线 CLB 分组连接点的方法获取轮回行所有 CLB 分组的起始连接点,然后分别对这些连接点进行测试验证,测试该分组是否合适,是否可以选出最优线序;测试的方法是:依据宽度优先算法搜索并选取该 CLB 布线在最大链长下的所有可布通的线网,如能布通则表示该分组暂时可行,并分别记录 CLB1、CLB2、CLB3、CLB4 可布通的组合序号以便在后续回退时调整该组合序号选出相应线序组合;当预估完成后进入步骤四,否则,如果所有组合测试后均未发现可布通的线序组合则回退至步骤二重新分组;所述 CLB1、CLB2、CLB3、CLB4 分别指每个 CLB

分组的第一个 CLB、第二个 CLB、第三个 CLB 和第四个 CLB；

步骤四、二次过滤：经过步骤 3 预估过滤后，确定了所有关键布线 CLB 分组的起始连接点和所有可行的线序组合后，再做进一步进行过滤排除：从第一个 CLB 分组的 CLB1 和 CLB2 开始，依次选取相邻下个分组的 CLB1 和 CLB2，在选取 CLB1 和 CLB2 的过程中过滤水平六长线的占用，并同时采用线序选优的策略选取最优线序组合，以降低 CLB1 和 CLB2 分别对 CLB4 和 CLB3 布线线网的影响；在选取 CLB1、CLB2、CLB3 和 CLB4 的过程中，考虑相互之间水平六长线和行间单长线的占用影响，当过滤完成后进入步骤五，否则失败回退至步骤二重新分组；

步骤五、线序拼接：在完成所有关键布线 CLB 分组的布线后，需要对相邻关键布线 CLB 分组的布线进行拼接来完成整个线网的布线，在拼接过程中，依据单个 CLB 在线序预估阶段选出的可行线序索引进行，当拼接成功则进入步骤八，否则当某个 CLB 在重试完所有的可行线序索引后均未拼接成功，则进入步骤六进行回退判定；

步骤六、回退判定：在线序拼接失败后，进行布线 CLB 回退位置的选择，首先选择拼接失败 CLB 的顶部 CLB，调整顶部 CLB 选取的线序组合对底部 CLB 布线有影响的占用集合，同时测试底部 CLB 是否可以布通；如果可以则记录该线序索引值，进入步骤七进行拆除重布；如果调整完所有线序索引仍未布通，则回退至该拼接失败 CLB 的上一相邻关键布线 CLB 分组内具有相同组内索引的 CLB，同样重试该 CLB 选出的可行线序索引进行，同时测试是否可以布通，如是，则记录该线序索引值，进入步骤七进行拆除重布，如果重试完所有的可行线序索引后仍未布通，则继续按照上述方法循环回退，直至回退位置为首行的首个分组为止；如果仍未成功则回退至步骤二重新分组；

步骤七、拆除重布：对已布线网进行部分拆除，拆除部分为布线失败 CLB 位置至回退 CLB 位置，拆除这之间所有已布的线网，并在拆除过程中分析造成失败的冲突占用点和线序索引，为回退重布提供依据，拆除完成后根据回退 CLB 位置记录的可行线序索引值进行资源重建即重布，重建完成则回退至步骤五，从该回退 CLB 位置开始继续未完成的线序拼接；

步骤八、结果扩展：完成拼接后的链路即为完整链路，即已完成了轮回行布线，覆盖了轮回行的所有 CLB；此时轮回行以外的 CLB 部分，将搜索到的链路中间结果按照行奇偶对应的原则扩展至整个芯片，在扩展的过程中严格遵守 XDL 的描述规则；

步骤九、结果验证：扩展完成后，即实现了整个 FPGA 的布线，将此时的中间结果直接导出为 XDL 格式文件，然后利用 XDL 命令，将导出文件转化为 NCD 网表进行结果验证。

[0007] 具体的，步骤二中所述的 CLB 和单长线分组方法，在型号为 XCV1000 的芯片上，其 CLB 分组、CLB 分组大小、单长线分组大小和单长线分组计算方式分别为：

CLB 分组：从公式

$$(\text{列宽} \times \text{轮回行大小} - 2) / (\text{CLB 分组大小} + \text{CLB 被测分组大小})$$

得到分组数目，再根据输入的分组偏移量分别获取所有分组所包含的 CLB 的具体行列值，所有这些分组称为关键布线 CLB 分组，每组又都包含 CLB1、CLB2、CLB3、CLB4 四个 CLB；将第一行的首个 CLB 分组作为第一个 CLB 分组，然后第一行按从左至右，第二行按从右至左的顺序依次将所有分组编号；而组序号从小到大相邻的 CLB 分组中有相同组内序号的 CLB 为起始和结束链路布线关系，第一组 CLB 的 CLB1 为第二组 CLB 的 CLB1 的起始链路布线 CLB，第二组 CLB 的 CLB1 为第一组 CLB 的 CLB1 的结束链路布线 CLB，同时第二组 CLB 的 CLB1 为

第三组 CLB 的 CLB1 的起始链路布线 CLB；

CLB 分组大小：单长线数目 24 / 输入线序数目 6 = 4；

单长线分组大小：单长线数目 24 / 输入线序数目 6 = 4；

单长线分组：使用组合算法，将 24 根单长线每次选出 6 根，分四次选出分成四组，每组包含 6 根单长线起始连接点，然后将选出的四个分组编号，当后续检测到失败回退需要重新分组时，将当前分组编号加 1 继续选取下个分组；分组后的单长线依次分别对应到首个 CLB 分组的四个 CLB 上，作为相应 CLB 布线起始连接点。

[0008] 步骤四所述的线序选优的策略是：依据单链长最短、总链长最短、稀缺资源占用最少、单长线垂直底部占用连接点  $S_x$  或单长线垂直顶部占用连接点  $N_x$  最少的原则判定最优；其中，在进行第一行布线时考虑单长线垂直底部占用连接点  $S_x$ ，当进行第二行布线时考虑单长线垂直顶部占用连接点  $N_x$ ； $x$  为 0 ~ 23 的任意整数值。

[0009] 步骤五所述对相邻关键布线 CLB 分组的布线进行拼接是将第一组的 CLB1、第一组被测 CLB 分组和第二组的 CLB1 进行拼接，第二组的 CLB1、第二组被测 CLB 分组和第三组的 CLB1 相拼接，并依次完成所有相邻 CLB 分组的 CLB1 的拼接，同理完成 CLB2、CLB3 和 CLB4 的拼接。

[0010] 步骤五所述的线序拼接是将相邻关键布线 CLB 分组的链路头、链路体和链路尾这三个部分依据连接对应关系进行对接，并最终形成完整的布线轨迹。

[0011] 本发明的优点是：本方法能快速进行资源配置和有效配置资源图的生成，对单长线及连接开关的资源覆盖率较高，且与具体硬件结构无关；本方法的实施能极大地增强 FPGA 的自动化布线和测试能力，有效减少后期布线工作量，节约测试成本，提高测试效率，缩短测试周期，有一定的应用和参考价值。

## 附图说明

[0012] 图 1 是水平单长线测试整体结构示意图。

[0013] 图 2 是算法涉及基本概念示意图。

[0014] 图 3 是本发明的流程图。

[0015] 图 4 是 CLB 内部连接关系图。

## 具体实施方式

[0016] 为了进一步说明后续的实施步骤，这里结合附图从资源的抽取抽象、模型的建立、到算法理论基础、涉及的基本概念，再到算法流程所涉及的算法核心和布线策略，分别做简要陈述；涉及到的概念大致如下。

[0017] 1、资源等价描述

在设计 FPGA 布线算法前期，算法设计者需要从 FPGA 硬件设计者手中获取该芯片的结构描述文件，对文件中所描述的硬件结构及连接关系进行逻辑等价、逻辑抽象，并将中间结果存入数据库，以便在后续搜索路径的过程中依据该结果进行 FPGA 结构的重建，得到计算机可以识别的单元、图或树的结构，完成整个芯片的布线流程。

[0018] 这里需要进行等价描述的逻辑资源有：1) CLB (可配置逻辑模块)、IOB、BRAM 等逻辑单元块；2) 各个逻辑块的输入输出引脚连接度；3) 连接线之间的驱动方式；4) 单长线、六

长线、长线等布线资源 ;5) 其他相关布线资源。

#### [0019] 2、 依据设计模型

本方法依据的设计模型和理论基础,即配置完成 FPGA 芯片单长线及其直连开关的布线方法详见专利文献——一种 FPGA 单长线及直连开关的测试方法(专利申请号 201110125752.7)。

[0020] 模型如图 1 所示,包括 :CLB 1 ;IOB 2 ;第一级 CLB 分组 3 ;被测试的 CLB 4 ;整张图表示了水平单长线的整体布线架构,为了测试单长线及互连开关的连接情况,这里要求被测的 CLB 要么用直向单长线互连要么用斜向单长线互连,且每次布图只允许用其中一种连接方式。

#### [0021] 3、 算法理论模型

在基础数据库搭建完成后,我们需要对搜索算法的基本概念作一了解,依据设计模型,在 24 根 D0 信号进入第一组 CLB 后,需要分别连接到水平或者垂直的单长线连接点上,接着 CLB 的输出线通过 GRM 的内部互连线连接到外部资源,然后通过下一组的 CLB 的内部互连线连接到 CLB,形成完整的单条线网 ;在整个过程中,最重要的有两个环节,一个是从 CLB 引出线网的过程,一个是在 CLB 中结束线网的过程 ;这两个过程,这里我分别称作为链路头布线和链路尾布线,剩下的从 CLB 横穿的部分我称之为链路体布线,最后,我们将链路头、链路体和链路尾拼接形成整条线网 ;行首 / 尾采用蛇形或者反蛇形布线方式连接。

[0022] 如图 2 所示,图中序号分别代表的资源为 :轮回行 5 (将在标题 4 说明);链路头 6,链路的起始部分 ;第一组 CLB 分组 7 ;链路体 8,链路的横穿部分 ;链路尾 9,链路的结束部分 ;第二组 CLB 分组 10 ;行首 / 尾连接方式 11,连接方式包括蛇形和反蛇形连接 ;第 N-1 组 CLB 分组 12,第 N 组 CLB 分组 13 (以下标题 8 中将详述分组目的和方法);横穿 CLB 的单长线互连方式 14,横穿的方式有直线和斜线两种方式,分别为了测试直向和斜向开关。

#### [0023] 4、 轮回行

轮回行即有一定轮回周期的行,连接点在从起始连接点经过一个轮回行周期的布线变换后,重又回到起始的连接点,这里变换所需要经过的行即为轮回行,为了规范期间,这里通过特殊变换使得轮回行大小为 2,且连接起始点均从奇数行开始 ;至于轮回行大小设置为 2 的原因,见以下标题 7。

#### [0024] 5、 最大搜索链长

在依据宽度优先算法对单线网布线的过程中,由于不考虑驱动和时延等因素的影响,搜索链长是无节制增长的,为了控制增长,控制算法复杂度,这里需要引入最大链长的概念,以限定单条链路的搜索长度,即 CLB 内部单链长的最大长度不能超过设置的最大链长值,这里我们控制的最大链长值为 10,初始值为 5,后期在搜索线网失败的情况下将逐步放宽该初始化值至最大值,直至布线完成。

#### [0025] 6、 算法流程

本方法的特色为采用了轮回行扩展模式布线,下面将算法流程的核心大致做一描述,首先将整个芯片结构按轮回行模式进行划分,接着按照一定算法完成轮回行布线,由于是多线网问题,所以轮回行布线实际包含两个部分 :单线网布线和整体布线。前者采用宽度优先搜索算法完成对单线网布线,后者在调用前者的基础上通过综合选优完成所有线网布线。在整个布线过程中,我们采用快速分组、线序预估、多点布线、二次过滤、线序选优、线序

拼接、回退判定、拆除重布及结果扩展等一整套布线方法和策略,有效降低后期布线资源的冲突和拥挤,并尽量避免线序对布线质量的影响,充分保证布通率。在完成轮回行布线后,依据奇数行和偶数行相照应的原则进行扩展,最终完成整个芯片的布线。

[0026] 如图 3 算法流程图所示,分别说明了算法的流程和关键步骤,由于只是为了说明算法的大致流程,故对其中某些步骤作了合并简化处理,对某些策略也未作标示。

#### [0027] 7、 设置轮回行

由于所实施的系列 FPGA 结构和市面上大部分商用 FPGA 结构均采用层次化对称式结构,经研究发现,在该系列 FPGA 结构下,前两行已基本具备了整个 FPGA 结构的特征,如 CLB 内部布线,换行布线,链路体组装,水平单长 / 六长线的占用排除等;唯一要解决的就是垂直六长线在后期布线的资源冲突问题(由于仅为了测试单长线开关的连通性,所以排除了长线资源),只要解决了该问题轮回行的条件就能满足;办法有两种:1)、后续行依次进行组合调整;2)、在前两行就排除掉可能引起冲突的占用点;第一种方法可行但相对耗时,第二种方法在考虑到六长线的隔 6 进入 CLB 的特点,那么可能引起冲突的后续奇数行仍然对应奇数行,偶数行对应偶数行,所以只要保证自身 S、N 的对应数字不同时存在相同值就可以了。在分析完可行性之后,我们决定采用效率较高的第二种方法。

#### [0028] 8、 快速分组

由于我们采用的是整组线序布线策略,那么在轮回行设定完成之后,首先要面对的就是分组;分组有两种:第一种是 CLB 分组,目的是为了便于组间回退、线网拼接及多点布线;第二种是水平单长线分组,目的是为了便于控制线序和选取最优;CLB 分组相对简单,分组的大小和个数依据芯片的宽度、输入节点个数等通过一定的公式计算得来,以 XCV1000 为例(以下举例均以该设备为例),这里我们要求输入点的个数必须被 24 整除且商为偶数,我以 6 作为输入点个数,那么其计算方式分别为:

CLB 分组:从公式

$(\text{列宽} 96 \times \text{轮回行大小} 2) / (\text{CLB 分组大小} + \text{CLB 被测分组大小})$

得到分组数目,再根据输入的分组偏移量分别获取所有分组所包含的 CLB 的具体行列值,所有这些分组称为关键布线 CLB 分组,每组又都包含 CLB1、CLB2、CLB3、CLB4 四个 CLB;将第一行的首个 CLB 分组作为第一个 CLB 分组,然后第一行按从左至右,第二行按从右至左的顺序依次将所有分组编号;而组序号从小到大相邻的 CLB 分组中有相同组内序号的 CLB 为起始和结束链路布线关系,第一组 CLB 的 CLB1 为第二组 CLB 的 CLB1 的起始链路布线 CLB,第二组 CLB 的 CLB1 为第一组 CLB 的 CLB1 的结束链路布线 CLB,同时第二组 CLB 的 CLB1 为第三组 CLB 的 CLB1 的起始链路布线 CLB;

CLB 分组大小:单长线数目 24 / 输入线序数目 6 = 4;

单长线分组大小:单长线数目 24 / 输入线序数目 6 = 4;

单长线分组:使用组合算法,将 24 根单长线每次选出 6 根,分四次选出分成四组,每组包含 6 根单长线起始连接点,然后将选出的四个分组编号,当后续检测到失败回退需要重新分组时,将当前分组编号加 1 继续选取下个分组;分组后的单长线依次分别对应到首个 CLB 分组的四个 CLB 上,作为相应 CLB 布线起始连接点。

[0029] 单长线分组和 CLB 分组具有一定的内在联系,首先,它们的组大小相同,且单长线的分组序号对应 CLB 的组内序号;其次,每个关键布线 CLB 分组内的 CLB 布线均是针对有对

应分组序号单长线的初始或变化的线序进行布线的。

#### [0030] 9、 线序预估

在选取完第一组 CLB 分组连接点后,我们依据整体获取轮回行所有关键布线 CLB 分组连接点,获取的方法如下(以第一个 CLB 分组的 CLB1 上的水平单长线起始点 W0 为例):

根据是测被测 CLB 的直向开关还是斜向开关分情况获取,如果要测直向开关则只有横穿 CLB 一种,如果要测斜向开关则分斜向上和斜向下两种,一共分三种情况,这里的三种情况又都包含 CLB 内部布线和 CLB 相邻布线,由于 CLB 相邻布线的连接关系无非是  $E_x \rightarrow W_x$  或者  $W_x \rightarrow E_x$  ( $x$  为  $0 \sim 23$  之间的可能值)的情况,三种情况的规律一致,不做过多讲述。这里把重点放在 CLB 内部布线上:a、直向,这种情况和外部情况类似, W0 到相邻 CLB 分组的 CLB1,连接起始点还为 W0 ;b、斜向上, E0 在第一个 CLB 中的连接情况为  $W_0 \rightarrow N_x \rightarrow E_y$  ( $x, y$  为  $0 \sim 23$  之间的整数值,该值取决与前面对单长线内部互连关系的抽象),到相邻 CLB 分组中 CLB1 的起始点为上边  $E_y$  经 CLB 相邻布线得到  $W_y$  ( $y$  值为  $0 \sim 23$  之间的整数值,该值为 W0 依据数据库中的单长线互连关系变换获取);c、斜向下跟斜向上基本类似,不同的是 CLB 内部  $W_0 \rightarrow N_x \rightarrow E_y$  需要变为  $W_0 \rightarrow S_x \rightarrow E_y$  ( $x, y$  为  $0 \sim 23$  之间的整数值,该值取决与前面对单长线互连关系的抽象),其它同斜向上情形。其中,  $E_x$  ( $E_y$ ) 表示单长线水平右边占用的一个连接点,  $W_x$  ( $W_y$ ) 表示单长线水平左边占用的一个连接点,  $S_x$  表示单长线垂直底部占用的一个连接点,  $N_x$  表示单长线垂直顶部占用的一个连接点。

[0031] 通过上述方式可以获取第一分组 CLB1 的 W0 经过变换后到第二个分组的连接初始值,依次可以获取第一分组 CLB1 的其它连接点到第二个分组的连接起始点,通过相邻获取的方式获取所有关键布线 CLB 分组中 CLB1、CLB2、CLB3、CLB4 的连接起始点。

[0032] 所以在确定是以上 3 种情况的任意一种时,就可以获取所有关键布线 CLB 分组下的所有起始连接点。然后进行分组测试验证,测试该分组是否合适,是否可以选出最优线序,如不合适则回退到第 8 步进行快速分组调整;测试的方法就是搜索并选取最大链长下的所有可能线网,如能布通则表示该分组暂时可行,否则调整分组继续测试直至可行,然后分别记录第一组、第二组、第三组的组合序号以便在后续回退时调整该序号选出相应分组。这个过程我称之为布线预估阶段。

#### [0033] 10、 多点布线

在线序预估阶段我们依据 CLB 分组,获取了所有关键布线 CLB 分组连接点线序,然后同时对多个分组关键点同时进行预估和布线,这里提到的多点是轮回行在 CLB 分组后具有相同组内序号的关键布线 CLB 分组;通过对这些关键布线 CLB 分组进行同时布线预估可以更快速的判定劣质线序并进行淘汰,以便高效高质量的进行线序分组。后期同样依据这些关键布线 CLB 分组进行布线和拼接,提高布线效率。这里提及的多点预估和多点布线统称为多点布线。

#### [0034] 11、 二次过滤

二次过滤是对线序预估后选取的线序进行第二次过滤淘汰,即行间和 CLB 间占用冲突问题,包括水平六长占用和垂直单长占用等;由于冲突是相互的,那么调整谁改变谁呢?

在 CLB 分组后,每组的 CLB 我按照先后给定了序号,第一行按从左至右排序,第二行从右至左排序,每组都从 1 开始。如分组大小为 4,则第一行第一组为 CLB1\_R1C1、CLB2\_R1C2、CLB3\_R1C3、CLB4\_R1C4,第二行最左边一组为 CLB4\_R2C1、CLB3\_R2C2、CLB2\_R2C3、CLB1\_R2C4

(注:CLB<sub>m</sub>R<sub>x</sub>C<sub>y</sub>, CLB<sub>m</sub>代表某个分组内的第 m 个 CLB, m 为 CLB 组内序号, x 和 y 分别代表行和列的序号), 由于第一行的 CLB1、CLB2、CLB3、CLB4 分别和第二行的 CLB4、CLB3、CLB2、CLB1 一一对应, 这里我们固定 CLB1 和 CLB2, 分别调整 CLB3 和 CLB4, 在选取 CLB3 和 CLB4 的过程中过滤行间单长线冲突; 当 CLB 的横穿数目大于 6 时, 则不考虑水平六长占用冲突, 否则需要在排除完行间冲突后再进行水平六长占用的排除, 如果在排除过程中发现有布不通的情况则同样需要回退至快速分组阶段重新选取线序; 由于选取 CLB1 和 CLB2 的结果直接影响到 CLB3 和 CLB4 的分组和选优, 所以在选取 CLB1 和 CLB2 过程中需要对结果进行判优, 那么如何选取最优呢?

## 12、 线序选优

在线序预估阶段, 为了判断水平单长线组合的可行性, 我们采用了全排列算法依次对线序进行预估, 并依次记录可行线序; 前期线序的选取直接影响后续线序的选取, 特别是 CLB1 和 CLB2 分组的线序选取, 为了给这些线序评优, 这里我们引入评分选优策略: 依据单链长最短、总链长最短、稀缺资源占用最少、单长线垂直底部占用连接点 S<sub>x</sub> 或单长线垂直顶部占用连接点 N<sub>x</sub> 最少的原则判定最优; 其中, 在进行第一行布线时考虑单长线垂直底部占用连接点 S<sub>x</sub>, 当进行第二行布线时考虑单长线垂直顶部占用连接点 N<sub>x</sub>。

[0035] 依据该策略分别对每组可行线序进行打分, 并按照分值越高越优秀的原则对所有线网排序, 这个综合评优的过程还要兼顾链路头和链路尾的总分值进行。

## [0036] 13、 线序拼接

在完成连接点线序选优、预估和二次过滤后, 我们已对分组关键布线 CLB 分组下的单个 CLB 分别提取了所有可布通的线序集合, 但单个 CLB 的布通不代表整个链路是可行的, 这时的布线是不完整的, 依据算法模型, 完整的链路同时包括链路头、链路体和链路尾, 所以这时需要对相邻关键布线 CLB 分组的组内和组间的 CLB 进行布线拼接, 拼接的过程中随时都有布不通的可能性, 那么在这种情况下发生时, 就需要考虑回退和拆除重布, 把搜索节点从当前点回退到回退点, 并拆除这之间的已布线网。

[0037] 线序拼接就是将相邻关键布线 CLB 分组的链路头、链路体和链路尾这三个部分依据连接对应关系进行对接, 并最终形成完整的布线轨迹, 其具体拼接方法为: 将第一组的 CLB1、第一组被测 CLB 分组和第二组的 CLB1 进行拼接, 第二组的 CLB1、第二组被测 CLB 分组和第三组的 CLB1 相拼接, 并依次完成所有相邻 CLB 分组的 CLB1 的拼接, 同理完成 CLB2、CLB3 和 CLB4 的拼接。

## [0038] 14、 回退判定

在线序预估阶段, 我们记录了关键布线 CLB 分组内的 CLB 下所有可行的排序索引值, 当在失败 CLB 重试完所有的排序索引值都没有完成拼接时, 我们即认为需要回退; 关于回退位置, 我认为有两个, 第一个: 上级 CLB, 因为上一个相邻关键布线 CLB 分组内具有相同组内序号的 CLB 直接影响该失败 CLB 的选取, 比如第四组的 CLB1 受第三组的 CLB1 的影响。第二个: 顶部 CLB, 因为顶部占用直接影响了底部 CLB 的线序选取; 在回退到顶部 CLB 前, 我们还需要记录该失败 CLB 的原始占用, 在重新选取线序后, 将更新过的占用与原始占用比对, 做出合理判断, 选择影响最小的优秀线序, 使后布的线网更容易布通; 回退到回退点后, 我们调整回退点的 CLB 排序索引值并重试, 若成功则从调整点顺序布下去, 若失败则同样先是回退到上级 CLB, 再回退到顶部 CLB, 依次循环直至第一行第一组的某个 CLB, 仍然失败则

对具有相同 CLB 组内序号的 CLB 进行分组调整并重试。当重试完所有分组仍无法布通时，我们则回退到 8 步重新进行分组。

#### [0039] 15、 拆除重布

关于拆除和重布；在回退过程中，我们对部分已布线网采取拆除策略，这里讲的拆除并非完全整个布线的拆除，仅是对从当前 CLB 到回退 CLB 之间的已布线网进行拆除；重布前，我们需要对因资源冲突造成布线失败的原因进行综合分析；在重布的过程中，我们采用失败的线网优先，被拆线网靠后的原则进行线网资源重建。由于拆除重布过程没有拆掉所有线网，所以能够有效减少拆线资源量，节省运行时间，提高整体布线效率。

#### [0040] 16、 结果扩展

在前两行布线成功的前提下，我们把前两行搜索成功的经验按奇数行和偶数行相照应的原则扩展至整个芯片，依次应用到所有行和列。在扩展的过程中我们严格遵循 XDL 描述规则，依据 XDL 所规范要求的 PIP、NET、INST、INPIN、OUTPIN 等标准形式进行格式化，PIP 和 NET 依据实际搜索的器件间的连线而定，包括 Slice、IOB、TBUF 等各种形式的 INST 按布线中间结果和输入输出参数形成，INPIN 和 OUTPIN 按实际的参数值进行格式化，并最终形成具有标准规范的中间结果。

#### [0041] 17、 网表验证

扩展完成后的数据，其格式已经完全符合了 XDL 规则，故可直接将布线结果导出为 XDL 格式文件，再利用 XDL 命令，将已包含了布线所产生的资源配置信息和线网信息的 XDL 文件转化为 NCD 网表，进行验证。

[0042] 在简要说明了算法所涉及的概念、设计模型、算法核心及布线策略后，下面再结合具体的布线流程对本发明的实施步骤作更进一步说明，在实施过程中，确定完 FPGA 芯片的型号后，我们将该芯片的型号信息作为算法的输入参数输入，此外，还要选取的输入参数包括链路起始点组合和链路结束点组合，这里我们分别选取 Slice1 的 4 根信号，自左至右分别为 {S1\_Y/S1\_YQ/S1\_X/S1\_XQ}；Slice0 的 2 根信号，自左至右分别为 {S0\_X/S0\_XQ} 作为输出；以及 6 根输入信号自左至右为 {S1\_G\_B1/S1\_BY/S1\_F\_B1/S1\_BX/S0\_F\_B1/S0\_BX}。这里选取的输入输出信号应一一对应。

[0043] 关于输入输出对应性的问题，在图 4 所示的 CLB 内部连接关系图上有很好的体现，每组 CLB 的输出到下一组的 CLB 输入都有固定的对应关系，如图上所示的 Slice0 的两组 {S0\_X/S0\_XQ, S0\_Y/S0\_YQ} 信号，左侧 CLB 在输入右侧 CLB 的过程中，X 与 XQ, Y 与 YQ 应一一对应串联。

[0044] 除以上的参数外，本算法还可灵活设置的附加条件包括：最大搜索链长，被测 CLB 分组大小、链路体穿过的 CLB 数，链路尾连接方式（蛇形和反蛇形），过滤管脚及偏移 CLB 数目等。

[0045] 在设置完参数及搜索条件后，本发明依据理论模型和一定步骤进行链路搜索，步骤大致如下：

步骤一、设置轮回行并初始化参数：确定轮回行的数目为 2，并为后续布线搜索到的链路中间结果准备存储空间，准备完成后进入步骤二；

步骤二、快速分组：依据分组算法依次将 CLB 和单长线进行分组，获取轮回行所有 CLB 分组和第一组 CLB 分组对应单长线起始组合点；分组完成后进入步骤三，如果失败则退出

算法,返回失败代码;

**步骤三、线序预估:**在选取完第一组 CLB 分组连接点后,通过获取轮回行所有关键布线 CLB 分组连接点的方法获取轮回行所有 CLB 分组的起始连接点,然后分别对这些连接点进行测试验证,测试该分组是否合适,是否可以选出最优线序;测试的方法是:依据宽度优先算法搜索并选取该 CLB 布线在最大链长下的所有可布通的线网,如能布通则表示该分组暂时可行,并分别记录 CLB1、CLB2、CLB3、CLB4 可布通的组合序号以便在后续回退时调整该组合序号选出相应线序组合;当预估完成后进入步骤四,否则如果所有组合测试后均未发现可布通的线序组合则回退至步骤二重新分组;所述 CLB1、CLB2、CLB3、CLB4 分别指每个 CLB 分组的第一个 CLB、第二个 CLB、第三个 CLB 和第四个 CLB;

**步骤四、二次过滤:**经过步骤 3 预估过滤后,确定了所有关键布线 CLB 分组的起始连接点和所有可行的线序组合后,再做进一步进行过滤排除:从第一个 CLB 分组的 CLB1 和 CLB2 开始,依次选取相邻下个分组的 CLB1 和 CLB2,在选取 CLB1 和 CLB2 的过程中过滤水平六长线的占用,并同时采用线序选优的策略选取最优线序组合,以降低 CLB1 和 CLB2 分别对 CLB4 和 CLB3 布线线网的影响;在选取 CLB1、CLB2、CLB3 和 CLB4 的过程中,考虑相互之间水平六长线和行间单长线的占用影响,当过滤完成后进入步骤五,否则失败回退至步骤二重新分组;

**步骤五、线序拼接:**在完成所有关键布线 CLB 分组的布线后,需要对相邻关键布线 CLB 分组的布线进行拼接来完成整个线网的布线,在拼接过程中,依据单个 CLB 在线序预估阶段选出的可行线序索引进行,当拼接成功则进入步骤八,否则当某个 CLB 在重试完所有的可行线序索引后均未拼接成功,则进入步骤六进行回退判定;

**步骤六、回退判定:**在线序拼接失败后,进行布线 CLB 回退位置的选择,首先选择拼接失败 CLB 的顶部 CLB,调整顶部 CLB 选取的线序组合对底部 CLB 布线有影响的占用集合,同时测试底部 CLB 是否可以布通;如果可以则记录该线序索引值,进入步骤七进行拆除重布;如果调整完所有线序索引仍未布通,则回退至该拼接失败 CLB 的上一相邻关键布线 CLB 分组内具有相同组内索引的 CLB,同样重试该 CLB 选出的可行线序索引进行,同时测试是否可以布通,如是,则记录该线序索引值,进入步骤七进行拆除重布,如果重试完所有的可行线序索引后仍未布通,则继续按照上述方法循环回退,直至回退位置为首行的首个分组为止;如果仍未成功则回退至步骤二重新分组;

**步骤七、拆除重布:**对已布线网进行部分拆除,拆除部分为布线失败 CLB 位置至回退 CLB 位置,拆除这之间所有已布的线网,并在拆除过程中分析造成失败的冲突占用点和线序索引,为回退重布提供依据,拆除完成后根据回退 CLB 位置记录的可行线序索引值进行资源重建即重布,重建完成则回退至步骤五,从该回退 CLB 位置开始继续未完成的线序拼接;

**步骤八、结果扩展:**完成拼接后的链路即为完整链路,即已完成了轮回行布线,覆盖了轮回行的所有 CLB;此时轮回行以外的 CLB 部分,将搜索到的链路中间结果按照行奇偶对应的原则扩展至整个芯片,在扩展的过程中严格遵守 XDL 的描述规则,扩展完成后进入步骤九进行结果验证;

**步骤九、结果验证:**扩展完成后,即实现了整个 FPGA 的布线,将此时的中间结果直接导出为 XDL 格式文件,然后利用 XDL 命令,将导出文件转化为 NCD 网表进行结果验证。

[0046] 本方法采用了轮回行布线、整体扩展的模式进行了 FPGA 布线;在布线过程中,又

利用整体分组、多点布线、综合评优等多套策略,有效降低了后期资源冲突几率和回退率,极大地提高了布线质量和布线效率;拆除重布的过程采用部分而非整体拆除的机制,大大减少了拆线数量,显著提高了运行速度;实验结果表明,与其它相关算法相比本方法具有布线效率高、单长线资源覆盖率高、布通率高等优势。

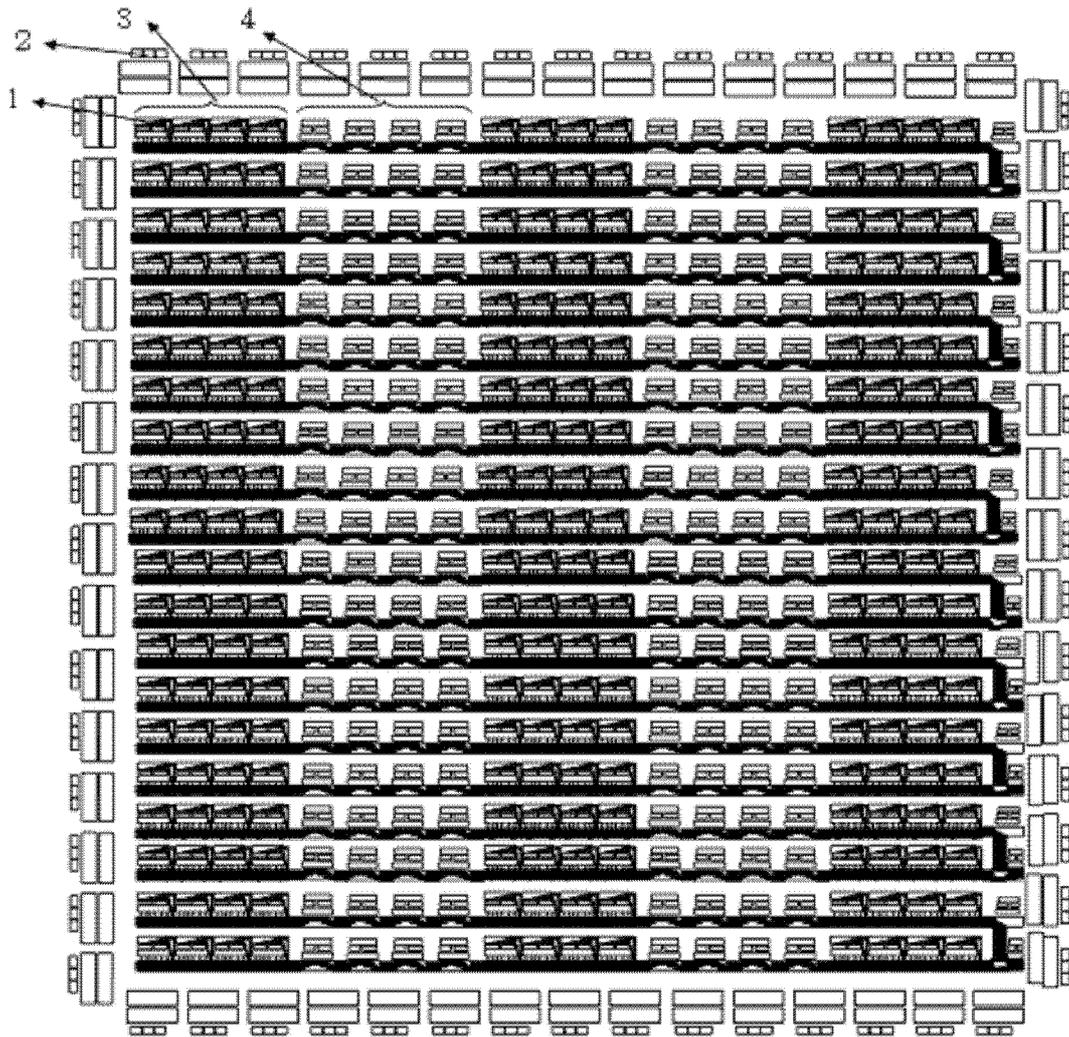


图 1

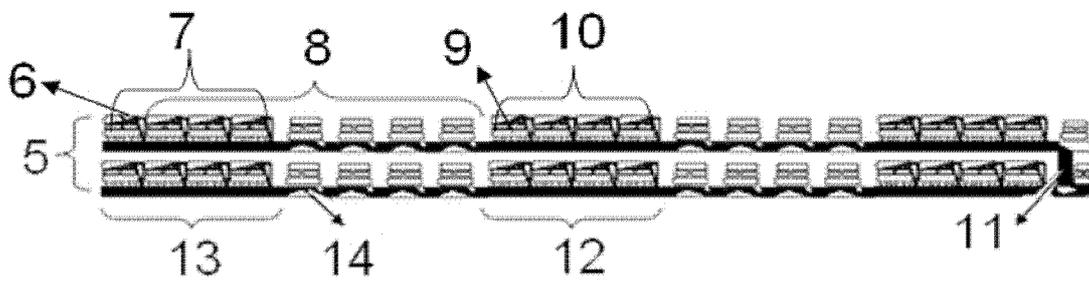


图 2

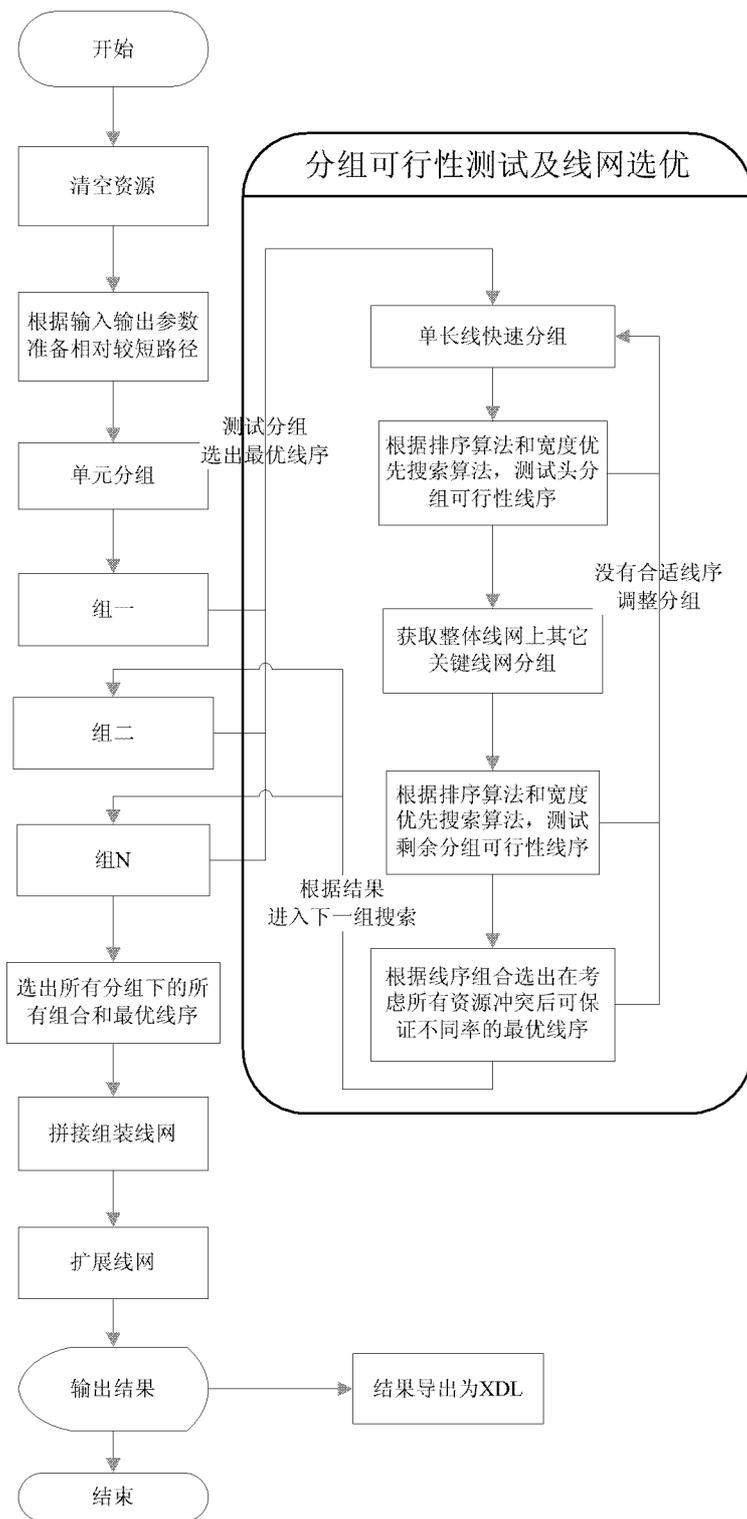


图 3

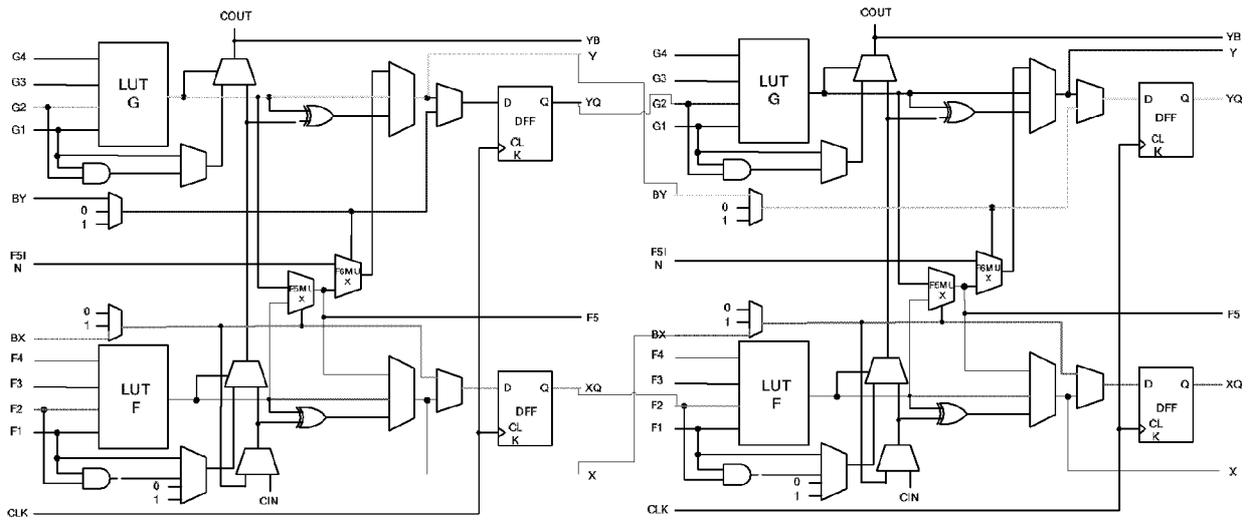


图 4