



(12) 发明专利

(10) 授权公告号 CN 101931395 B

(45) 授权公告日 2014. 06. 25

(21) 申请号 201010203156. 1

CN 1839390 A, 2006. 09. 27, 全文.

(22) 申请日 2010. 06. 18

US 2007/0011628 A1, 2007. 01. 11, 全文.

(30) 优先权数据

CN 101114306 A, 2008. 01. 30, 全文.

61/218, 294 2009. 06. 18 US

CN 101393640 A, 2009. 03. 25, 全文.

(73) 专利权人 英赛特半导体有限公司

US 6009252 A, 1999. 12. 28, 全文.

地址 加拿大安大略省

审查员 孙艳兵

(72) 发明人 维亚切斯拉夫·L·扎瓦迪卡

爱德华·凯斯

(74) 专利代理机构 北京信慧永光知识产权代理

有限责任公司 11290

代理人 褚海英 武玉琴

(51) Int. Cl.

G06F 17/50 (2006. 01)

H03K 19/20 (2006. 01)

(56) 对比文件

CN 101063987 A, 2007. 10. 31, 说明书第

7-18 页, 附图 1-9.

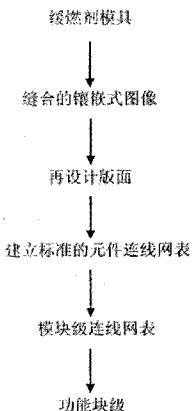
权利要求书2页 说明书12页 附图10页

(54) 发明名称

集成电路分析系统和方法

(57) 摘要

本发明使用结构数据开采方法和系统,结合分割提示和启示,以此在集成电路大门级中寻找高级元素库中功能模块的位置。在本发明的一个实施方法中,元素库是通过合成各种设计模块和约束块得以建立的。这个方法支持网络表和元素库之间以及元素库和网络表之间的特性匹配。本文说明的数据开采方法用于子图培养法,梯次说明集成电路网络表图形表达形式的特性。在本发明的一个实施方法中,我们使用了替补的杂乱脉冲干扰去表现子图的特性。此外,可以使用查寻到的高级功能块去代替具有匹配特性的目标网络表的相应部分,也可以对新产生的网络表进行注释。



1. 一种方法, 使用其中至少一个参考集成电路的至少一个参考网络表或其中的一部分, 分析取自至少第一个集成电路的至少一个目标网络表或其中一部分, 该方法包含以下步骤:

a) 说明所说目标网络表或其中一部分的特性, 以此获得所说目标网络表或其中一部分的多个特性;

b) 说明至少一个参考网络表或其中一部分的特性, 以此获得至少一个参考网络表或其中一部分的多个特性, 所述参考特性说明所述至少一个参考网络表的高级功能块的特性;

c) 将从 a) 步骤中得到的多个特性和从 b) 步骤中得到的多个特性相匹配, 并且

d) 提取从 c) 步骤中得到的匹配信息, 使用提取的匹配信息注释目标网络表中具有与所述参考网络表中一部分相匹配的特征的一部分, 由此在所述目标网络表中识别一个或多个所述高级参考功能块。

2. 根据权利要求 1 的方法, 至少第一个集成电路和至少一个参考集成电路相同。

3. 根据权利要求 1 的方法, 参考网络表从至少一个包含根据 VerilogTM 硬件说明语言 (VHDL) 完成的设计、功能块元素库和设计约束的小组的内容中合成而来。

4. 根据权利要求 3 的方法, 说明特性的 b) 步骤包含:

b1) 为合成的电路而提取 VHDL 小片;

b2) 为合成的电路合成已提取的小片;

b3) 为合成的小片计算杂乱脉冲干扰, 其中, 所述杂乱脉冲干扰是作用于目标的功能, 用任意长度的比特矢量表示, 且所述杂乱脉冲干扰的输出是具有固定比特长度的值;

b4) 比较计算出的杂乱脉冲干扰并且保留具有最大的独特杂乱脉冲的模块。

5. 根据权利要求 1 的方法, a) 或 b) 说明特性的步骤包含计算杂乱脉冲干扰。

6. 根据权利要求 1 的方法, c) 匹配的步骤包含不精确的匹配。

7. 根据权利要求 1 的方法, 该方法受分割暗示的指导。

8. 根据权利要求 1 至 7 中任何一种方法, 在步骤 c) 以后, 还有步骤包括:

d1) 用具有匹配的参考网络表中一部分的多个特性, 取代目标网络表一部分的多个特性。

9. 一种方法, 它使用取自至少一个参考集成电路的至少一个参考网络表或其中的一部分, 使用具有所说参考集成电路或其一部分特性的参考网络表或其中的一部分, 说明取自至少第一个集成电路的至少一个目

标网络表或其中一部分的特性, 该方法包含:

a) 从所说目标网络表或其中一部分中选择一个种子部分;

b) 使用结构数据开采技术说明种子的特性, 以此获得种子的特性;

c) 将该种子和所说参考网络表或其中一部分相匹配, 如果匹配, 通过将邻近匹配的种子合并和重叠, 梯次培养所述种子, 重复步骤 b) 至 c);

d) 否则, 摈弃该种子。

10. 权利要求 9 的方法得到分割暗示的指导。

11. 权利要求 9 的方法, 其中有步骤 a), 包含选择布尔功能。

12. 权利要求 9 的方法, 其中有步骤 a), 包含选择行为型种子。

13. 权利要求 9 的方法, 其中有步骤 a) 包括, 根据种子在所说的目标网络表中的出现频

率选择一个种子。

14. 权利要求 9 的方法,其中有步骤 a) 包括,根据所说的目标网络表中的种子的独特性选择一个种子。

15. 权利要求 9 的方法,其中有步骤 a) 包括,考虑到输入和输出、或者时钟及其重组信号,从可能是一个功能块的网络表中甄选一小部分。

16. 权利要求 9 的方法,其中有说明特性的步骤 b),包含计算杂乱脉冲干扰。

17. 权利要求 9 的方法,其中有进行匹配的步骤 c),包含不精确匹配。

18. 权利要求 9 的方法,其中有培养策略的步骤 c),包含优先处理多个候选种子。

19. 权利要求 18 的方法,其优先处理取决于每个候选种子的大小。

20. 权利要求 18 的方法,其优先处理取决于所说目标网络表中每个候选种子的出现的次数。

集成电路分析系统和方法

发明领域

- [0001] 本发明涉及集成电路，特别是集成电路的分析系统和方法。
- [0002] 发明背景
- [0003] 正如能被对集成电路工艺有一般性技能的人领会的那样，例如，要通过成品来理解集成电路难度极大。例如，现代电子集成电路通过高度自动化的流程制作。举一个例子，集成电路的设计一般是用硬件说明语言写一个功能的高级说明，并在逻辑大门级进行合成。这个方法，加之日益广泛地使用了原先设计好的电路子块（例如硬件和 / 或如软件“宏功能”），推动了各中各样大门设计的程序制作。
- [0004] 这个高度自动化的过程也称之为自动路由，往往包括能自动在足够的空间中放置电子元件的软件程序。这个方法得到的结果未必是按逻辑排列或视觉上很美观的布局。因此表述相同功能模块的布局可能会不同。
- [0005] 虽然人们已经开发了集成电路的逆转工程技术，用以再制作集成电路的低级网络表（另外也称之为大门、原始级或单元级网络表），其自动化和准确性相当高，但是，要将这些网络表进行组织和分析，使之成为功能性的、模块化的和 / 或分等级的模块，目前还严重依赖专家大量的人工处理。这样做不仅费时，而且价格高昂。
- [0006] 人们在这个工艺中使用了不同的方法，提供从集成电路或集成电路的功能识别中自动提取逻辑。雷斯特提出的一个方法使用电路拆析程序来提取逻辑。（见 Lester, A. Bazargan-Saber, P. Greiner, A. 发表的题为“YAGLE，用于 CMOS VLSI 电路的第二代功能逻辑提取法”，登载于《第十次微电子国际大会议程》，1998 年，第 265–268 页）YAGLE 能够应用于网络表的提取，以此减少模拟时间。然而它并不能提供让人理解的结果。此外，YAGLE 方法极易造成网络表的误差。
- [0007] 另一个方法由 Boehner, M 提出，它提供了从晶体管到大门级的规律性提取法（见“LOGEX-CMOS 技术从晶体管到大门级的自动化逻辑提取法”，登载于《第 25 次 ACM/IEEE 自动化大会议程》，1988 年，第 517–522 页。这一方法的缺点之一是在高于大门级的更高级的逻辑提取中缺乏可行的规律性方法。
- [0008] 另一个已知的方法由 Ohlrich, M., Ebeling, C., Ginting, E. 和 Sather, L 发表。（见“子双子星座：使用快速子图同形算法识别子电路”，登载于《第 30 次国际自动化大会程序》，1993 年，第 31–37 页）虽然 Ohlrich, M 等人提出的方法是该领域最重要的算法之一，但它需要一个完全已知的基础元素库。因此从现代优化的时机变更和合成技术的角度，该方法不适合高级的功能识别。
- [0009] 因此，我们需要一个新的集成电路分析系统和方法，去克服已知方法中的缺点，至少为公众提供一个有用的替补方法。
- [0010] 我们提供这个背景资料披露了申请人认为同本发明相关的信息。我们没有承认，也不应猜测前面的任何信息构成同本发明相左的工艺。

发明总结

[0011] 本发明的一个目标是提供新集成电路的分析系统和方法。根据本发明的各个方面，本文翔实地说明了集成电路分析的不同方法和系统。

[0012] 本发明的另一个目标是，使用概率性模型结合分割暗示、手印匹配和图形匹配等手段，对于具有功能相等逻辑模块的集成电路，提供新的电路分析系统和方法，以此来表述集成电路中一系列大门逻辑。

[0013] 本发明的另一个目标是，通过培养出经过选择的、使用各种培养策略和分割暗示的子图，提供一种方法和系统，使之能梯次表示集成电路大门级网络表的特性。

[0014] 根据本发明的一个方面，我们提供一种方法，通过使用取自至少一个参考集成电路的至少一个参考网络表或其中部分，分析取自至少第一个集成电路的至少一个目标网络表或其中的一个部分。该方法由以下步骤组成：a) 说明所说目标网络表或其中一部分的特性，以获取所说多个目标网络表或其中的一部分；b) 说明所说至少一个参考网络表或其中的一部分的特性，以获取所说至少一个参考网络表或其中一部分的多个特性；c) 用步骤 a) 中获得的多个特性与步骤 b) 中获得的多个特性相匹配；d) 提取从步骤 c) 中获得的匹配信息。

[0015] 根据本发明的另一个方面，我们提供一种方法，通过使用取自至少一个参考集成电路的至少一个参考网络表或其中一部分，使用具有所说参考集成电路一部分特性的参考网络表或其中一部分，说明取自至少第一个集成电路的至少一个目标网络表或其中一部分的特性。该方法由以下步骤组成：a) 从所指目标网络表或其中一部分中选择一个种子部分；b) 使用结构数据开采技术说明这个种子的特性，以获得种子的特性；c) 将这个种子和所说参考网络表或其中的一部分加以匹配，如果匹配，就使用包括临近子图的培养策略培养出种子，并且重复步骤 b) 到 c)；d) 否则，摒弃这个种子。

[0016] 根据本发明的另一个方面，我们提供一种方法，使用取自至少一个参考集成电路的至少一个网络表或其中的一部分，使用具有所说参考集成电路杂乱脉冲干扰特性参考网络表或其中一部分，梯次说明表述大门级网络表或集成电路一部分的那个图形的特性。该系统包括这些步骤：a) 从该图形中选择一个种子子图；b) 计算该种子子图的杂乱脉冲干扰值；c) 将计算出的杂乱脉冲干扰值和所说参考网络表或其中一部分的杂乱脉冲干扰值相匹配，如果匹配，使用包括临近子图的培养策略培养出种子子图，并且重复步骤 b) 到 c)；d) 否则摒弃这个种子子图。

[0017] 根据本发明的另一个方面，我们提供一种系统，使用取自至少一个参考集成电路的至少一个参考网络表或其中一部分，分析取自至少第一个集成电路的至少一个目标网络表或其中一部分。该系统包括：本文实施的提供带电脑可读代码的储存器，可以通过 CPU 运行。该代码包括：a) 代码手段，它能说明所说目标网络表或其中一部分，以获得所说目标网络表或其中一部分的多个特性；b) 代码手段，它能说明至少一个参考网络表或其中一部分的特性，以获得所说至少一个参考网络表或其中一部分的多个特性；c) 代码手段，它能将从 a) 步骤中得到的多个特性和在 b) 步骤中得到的多个特性相匹配；d) 代码手段，它能从 c) 步骤中提取匹配的信息。

[0018] 根据本发明的另一个方面，我们提供一种系统，使用取自至少一个参考集成电路的至少一个参考网络表或其中一部分，分析取自至少第一个集成电路至少一个目标网络表或其中一部分。该系统包括：本文实施的提供带电脑可读代码的储存器，该代码可以通过

CPU 运行, 使用所说参考网络表或其中一部分, 使用具有所说集成电路或一部分特性的参考网络表或其中一部分, 说明该所说目标网络表或其中一部分的特性。该代码包括 :a) 代码手段, 它能从所说参考网络表或其中的一部分中能够选择种子部分 ;b) 代码手段, 它能使用结构数据开采技术说明该种子的特性, 以获得该种子的特性 ;c) 代码手段, 它能将该种子和所说参考网络表或其中一部分相匹配, 如果匹配, 使用包括临近种子的培养策略培养出该种子, 并且重复步骤 b) 到 c) ;d) 否则摒弃该种子。

[0019] 根据本发明的另一个方面, 我们提供一种系统, 使用其中至少一个参考集成电路的至少一个参考网络或其中一部分, 分析取自至少第一个集成电路至少一个目标网络表或其中的一部分。该系统包括 :本文实施的提供带电脑可读代码的储存器, 该代码可以通过 CPU 运行, 使用所说参考网络表或其中一部分, 使用具有所说参考集成电路或其一部分杂乱脉冲干扰特性的网络表或其中一部分, 梯次说明表述所说目标网络表或其中一部分图形的特性。该代码包括 a) 代码手段, 它能从该图形中选择种子子图 ;b) 代码手段, 它能计算出该种子子图的杂乱脉冲干扰值 ;c) 代码手段, 它能将计算出的杂乱脉冲干扰值和所说该参考网络或其中一部分的杂乱脉冲干扰相匹配, 如果匹配, 使用包括临近种子子图的培养策略培养出该种子子图, 并且重复步骤 b) 到 c) ;d) 否则摒弃该种子子图。

[0020] 附图概述

[0021] 我们使用供参考的附图说明本发明, 这些附图为 :

[0022] 图 1 说明集成电路逆转工程中涉及的一般性步骤 ;

[0023] 图 2 为集成电路大门级网络表和网络表图形表述的图例 ;

[0024] 图 3 是根据本发明一个实施方法制作的元素库流程图图例 ;

[0025] 图 4 是图 3 中制作的元素库内部特性结构的图例 ;

[0026] 图 5 是在目标集成电路中选择种子电路流程的图例 ;

[0027] 图 6 是根据本发明一个实施方法生长种子电路元素流程图例 ;

[0028] 图 7 是布尔确定图例 ;

[0029] 图 8 根据本发明一个实施方法形成行为型种子过程图例 ;

[0030] 图 9 这个图例说明根据本发明一个实施方法, 表示使用在目标开采流程中的一般性排列结构 ;

[0031] 图 10 和图 11 是在目标集成电路中寻找结构的过程图例 ;

[0032] 图 12 是梯次说明子图特性的流程图 ;

[0033] 图 13 是两个子图合二为一的图例 ;并且

[0034] 图 14 这个图例显示使用替代的杂乱脉冲干扰算法, 说明带标签的图形特性的过程。

[0035] 详述

[0036] 本文说明的不同实施方法, 应对各种难题, 包括某种分析低级网络表以此懂得集成电路中包含的功能, 特别是针对较大型的低级网络表, 我们举例说明了各种难题。为了便于说明, 本文中的“低级”、“大门级”和“单元级”等词语可以交换使用, 它们通常指较为低级的网络表, 需要进一步的处理, 以确定集成电路中各种层次、模块和 / 或功能。

[0037] 图 1 中的流程图说明根据本文透露的本发明中各实施方法, 分析集成电路所涉及的一般性步骤, 从再制作集成电路的布局和建立对其有表述性的标准单元网络表的起始步

骤,到建立模块级网络表和功能级说明。例如,在一个实施方法中,可以使用一个到多个人工操作 / 执行的和 / 或一个到多个部分或全部自动化的处理系统,自动从集成电路图形中提取单元。

[0038] 参考图 1 并根据本发明的一个实施方法,我们可以看到,集成电路的逆转工程始于给集成电路上缓燃剂,拍摄每层的图形并将所有的图形缝合在一起形成一个马赛克画面。这样我们就可以获得各组成部分之间的联系,以便再制作一个广阔的平面电路布局。标准的单元可以从集成电路的马赛克画面和平面布局中人工或自动化提取。一旦单元被提取后,它们就可以用人工或自动化手段连接起来,或者通过一种或多种辅助系统连接起来,以此制作经过分析过的装置的单元级网络表。单元级网络表可以用人工或自动化手段进一步分析,以便获取层次性的模块级网络表或高级的功能块图形。应该指出的是,虽然这种单元级网络表包含的装置处理、拍摄图形以及图形识别工艺造成了一些误差,但在整个过程中这些误差可以得到纠正,以便获得有用的结果。也就是说,虽然应用程序的设计通常容易出现网络表方面的误差,由于它们对于逆转工程中具有普遍的应用性,尽管在起始阶段的网络表中会出现误差,本文说明的方法和系统能够提供可以利用的结果。

[0039] 正如了解集成电路工艺的人们能理解的那样,本文说明的各实施方法中的方法和系统可以从起始点开始执行。例如,在一个实施方法中,我们可以对这种方法和系统加以配置,推动和 / 或促进图 1 中说明的过程所需的各个步骤,以及 / 或者其它相关的但没有明确说明的步骤。也就是说,我们可以对这种方法和系统加以配置,从加了缓燃剂的模具和缝合的马赛克图中,再次制作目标布局,进而制作参考性的标准单元网络表,并且从中产生某个模块级和 / 或功能块的说明。在其它实施方法中,我们可以从以前制作的标准单元网络表中,对这个方法和系统加以配置进行实施,以此产生目标集成电路模块和 / 或功能块的说明。

[0040] 对集成电路实行逆转工程这个过程,可能会产生各种误差,如不规则的形状和线条厚度、登记误差和连接误差等,并且在处理由扩散和井坑等问题引起的平整问题时,这些误差还会进一步复杂化。因此,对集成电路具工艺有一般性技能的人能够理解,处理一个经过逆转工程的布局比处理一个原始的布局,即处理图像数据库系统 II 的格式 (DGSII),其难度更大。因此,集成电路工艺中的逆转工程往往涉及到让专家分析师手工识别集成电路设计中的每个独特的标准单元,并且识别它的功能。这样会让一个专家花上几个星期的时间识别完使用大型单元库做成的设计。

[0041] 一般来说,有可能通过一个或多个电脑程序(例如,软件和 / 或硬件带动的应用程序、算法和 / 或装置)对逆转工程的过程实行自动化,到达制作标准单元网络表这一步。在本发明的一个实施方法中,我们也执行了该系统和方法来分析集成电路,从低级网络表中 (NAND, NOR, DFF, MUX 等) 做提取。从提取的内容中可以作进一步的分析和网络表的组织。例如,在一个实施方法中,可以从低级的网络表中产生多达几十万个标准的单元,甚至可以达到上百万个半导体。然而,至少在某种程度上,有必要弥补低级网络表和人们可读的高级集成电路说明之间的差距。

[0042] 集成电路工艺中通常需要专家们的干预,作为对这种做法的一种改进,本文说明的本发明的实施方法,在识别模块和 / 或从集成电路的布局图形中,或者从以前获得的单元级网络表中,识别更高的集成电路表述特性,实现更大的自动化程度。

[0043] 为了可以跟踪对数字集成电路的分析,特别是大型数字集成电路,我们希望将它们设计时的同等自动化程度用于逆转工程。同样,根据本发明的一个实施方法,我们提供了一种依赖集成电路内部电路块功能的传播特性的分析系统和方法。该方法根据使用已知电路元件元素库的结构数据开采技术提出。例如,这个系统和方法从低级的网络表出发,依赖对普通数字模块(例如加法器、乘法器、计数器、移位寄存器、编码器、解码器、横杆、DACS、ADCs、CAMs等)的识别和层次表述,提供一种功能方面的更高分析。包含这种数字模块特性的合成的集成电路就具备这种功能。

[0044] 在一个实施方法中,这个方法和系统提供人们能够理解的结果,因而要获得所分析的集成电路的功能,就无需专家的进一步分析。

[0045] 在本发明的另一个实施方法中,可以实施该系统和方法而无需借助完全已知的基础元素库。在现代化的优化式时机性修改和合成技术的背景下,这种元素库并不总是存在。

[0046] 结构数据开采指的是从某些结构式的(如图形等)数据集中寻找和提取有用的信息。根据本发明的一个方面,结构数据开采法用于寻找一个网络表和另一个网络表直接有意义的结构性匹配。更具体地说,该方法用于查找集成电路目标和模块元素库之间、一个元素库和另一个元素库之间、一个集成电路目标和另一个集成电路目标之间的匹配。相对于一个目标或一个元素库而言,开采过程就是将目标或元素库作为结构数据集,其相应的信息与该目标或该元素库或它们的组成部分有关。我们可以用一个合适的机制来说明这个结构数据集及其相应信息的特性,该机制可以使用到结构数据集和相应的信息中去。我们可以将这种特性的结果加以对比,或者用来识别两个网络表之间的匹配。根据本发明的一个方面,可以使用图形来表述网络表,并且可以使用杂乱脉冲干扰来说明其中的图形和标签。

[0047] 如上所述,我们开发了逆转工程技术用于制作集成电路的低级网络表,这个过程具有较高的自动化程度和准确性。例如,我们提取标准单元来提供一个单元或大门级网络表。这样的大门级网络表可以用仿真模型的文本格式来表述。根据本发明的各实施方法,我们使用一个图形来做这种大门级网络表的模型,其中大门作为至高点来表述,将大门的功能编码在标签中,大门之间的联系作为边缘来表述,将针和输入输出信号流的方向编码在标签中。然而,将这种网络表组织和分析成为具有功能的模块和/或层次性的模块,我们目前仍然必须严重依赖大量专家的人工干预。至今在某种程度上,我们还没有找到合理的方法去弥补低级网络表和人工可读集成电路高级说明之间的差距。

[0048] 图2表明大门级网络表2500,它具有两个NOR2的大门2501,每个大门有两个输入2502和一个输出2503。

[0049] 根据本发明,图2是网络表2500的图形表述2600。图形2600有两个至高点2604和2605,表述两个门2501,它们分别具有NOR22602和NOR22603的功能类型。至高点2604和2605通过边缘2601相连接,该边缘表示2501两个门之间的连接:一个从输入到输出(I:0),一个从输出到输入(0:I)。较为有利的情况是,为了节省储存器并取得更为优化的结果,我们可以将网络表中和至高点、边缘、功能类型和连接相关的标签作为64比特的杂乱脉冲干扰储存起来。这种杂乱脉冲干扰可以从它们编码的文本串中计算出来。为了便于这里的说明,杂乱脉冲干扰是作用于目标X的功能h(X),用一个任意长度的比特矢量表示。这个杂乱脉冲干扰功能是个具有固定比特长度1的值。一般来讲,我们可以促使“假阳性”的可能性即X≠但是h(X)=h(Y)这个条件接近于2-1。同样,如果杂乱脉冲干扰比特长度1

够大,我们可以将目标之间的直接对比用与这些目标相应的杂乱脉冲干扰来取代。在本发明的一个优先实施方法中,我们使用了 64 比特的杂乱脉冲干扰,“假阳性”的匹配很低,可以忽略。这现象表明,为了比较网络表和其中相应的图形,相等的杂乱脉冲干扰意味着相等的图形和相等的网络表。

[0050] 正如能被对集成电路工艺有一般技能的人领会的那样,我们可以有多种方法实施一个低级网络表中相同的较为高级的功能。然而在本发明中,我们必须意识到,实施相同的高级功能通常将会有某些共同的特性(例如,网络表的小片、小布尔功能和 / 或共同的连接方式)。例如,这些特性可能表述了共同的硬件算法,如加法器(RCA 并列前缀加法器、进位跳跃式加法器等)、树状晶体结构、并列乘积产生器、乘法器、乘法 - 累加器等等。同样,在识别了这些较小的模块之后,我们可以通过观察这些较小的模块的存在和形式,去推断更高级模块的功能。

[0051] 在有些实施方法中,本发明使用了已知功能的数字电路元素库,这种电路由不同的合成工具,包括带有不同约束功能的稍有不同的各种储存器转换级。在一个实施方法中,我们使用了结构数据开发技术,以此寻找网络表各部分和已知功能电路元素库之间的匹配。例如,网络表可以用图像模型来表示,从中我们可以采用结构数据开采算法去寻找各种同形的子图(例如相同或相等的网络表小片)。我们不仅可以使用图像模型表述网络表,表述结构数据开采法,用于在大型的网络表中寻找同等的电路,用于制作自动化电路图,并且 / 或者寻找已知元素库元素的实例。根据有些实施方法,我们可以应用这些技术,从网络表中去提取更高级的功能。

[0052] 我们可以使用高级的功能块取代具有匹配特性的网络表中的相应部分,同样也可以用来表示新产生的网络表。

[0053] 例如,在一个实施方法中,我们将一个已知或独家的图像模型用于同形和结构数据的开采。在这个实施方法中,可以使用一个模型来减少至高点的数量,并且为标签的信息编码,这样可以减少结构数据开采中的组合时间。

[0054] 在有些实施方法中,我们可以使用一个或多个数据开采技术,包括但不限于杂乱脉冲干扰、提取指纹和一些替代方法。例如,一个 64 比特的整数“指纹”可以用来计算低级网表格中电路的一部分。我们还可以作为起点识别和使用同等的指纹,用于图形的匹配和说明电路的特性。

[0055] 在有些实施方法中,我们也可以使用结构的分割暗示,例如但不限于时钟树状电路、扫描链和 / 或具体的控制信号,用来寻找相关电路的位置,缩减需要计算的指纹数的组合生长。也可以使用至关重要的信号方法和 / 或反馈分析法。

[0056] 在有些实施方法中,这个系统和方法可以通过概论模型结合分割暗示、指纹匹配和图形匹配等手段,表述带有具体功能块的大型大门集。

[0057] 如上述介绍的一个实施方法,不同的合成工具和 / 或不同的方式可以产生具有已知功能的数字电路元素库,我们使用这样的元素库实施这个已透露的系统和方法。例如,我们可以编制标准的硬件算法和常用的高级模块去繁殖这个元素库;获取和 / 或开发结合这种常用高级模块的储存器转换级;使用不同的合成工具;产生有注释的大门和 / 或低级网络表,并且分析新产生的网络表,了解共同点、优化结果、时钟树状电路的分布和测试链。

[0058] 根据本发明的一个方面,图 3 用例子说明了创建具有已知功能数字电路元素库的

过程。使用仿真模型硬件说明语言 (VHDL) 3001 和 DesignWare™ 已知参考功能块 3002 设计的储存器级电路, 具有各种设计制约 3003, 如速度、区域、动力、排错 / 电路扫描和额外的缓冲等。一开始, 我们使用这种储存器级电路合成了一个大门级的平面仿真模型 Verilog™ 网络表 3004。从这个 Verilog™ 网络表 3004, 用于已知功能块的 Verilog™ 小片提取了出来 3005, 并且巩固成为独特的 Verilog™ 小片 3006。下一步, 我们对 Verilog™ 小片的杂乱脉冲干扰作了比较 3007。在这个过程的末尾, 元素库包含了具有独特杂乱脉冲干扰 3008 的参考功能块。

[0059] 图 4 说明本发明如何描述使用杂乱脉冲干扰的功能组成部分的元素库。图 4 中的表格显示了上述元素库产生这个步骤 3008 前的结构。在图 4 中, 各种已知的功能块 3202 在不同的储存器转换级位置得到了识别。例如, 组成部分 3203 位于储存器转换级 (I17, I40 和 I44), 杂乱脉冲干扰代码为 903。这个组成部分 3203 具有两个可能存在的加法器 75 的前任组成部分, 被杂乱脉冲干扰代码 493 所识别。反过来, 杂乱脉冲干扰代码 493 的加法器 75, 在不同的储存器转换级位置具有 70 个加法器 75 类前任, 被杂乱脉冲干扰代码 257 和 258 所所识别。在本发明的一个方面, 由图 3 显示的 3008 步骤所示, 为了促进逆转工程的集成电路的特性化, 只保留元素库中功能组成部分的独特杂乱脉冲就足够了。

[0060] 在有些实施方法中, 我们使用了不同的算法去识别和处理整体的结构, 如扫描链、时钟树状电路的分布和作为分割暗示使用的母线。

[0061] 在有些实施方法中, 我们使用了不同的算法去识别和提取小型的布尔功能, 包含到图像模型中。

[0062] 在有些实施方法中, 我们使用了不同的结构数据开采技术处理大型的数字网络表, 主要是调整优化的算法, 以减少需要计算的子图指纹的组合数字的增长。在有些实施实例中, 我们可以使用算法优化法、领域知识的应用 (例如周期和关键信号) 和 / 或分割暗示等。

[0063] 在有些实施方法中, 为了优化这个系统和方法, 可以使用某些算法, 在已经有注释的元素库中, 为相同功能的模块寻找匹配。可以重复这个过程直到找到足够密度的匹配。

[0064] 在有些实施方法中, 我们使用了概论模型, 将网络表的开采结果同功能块网络表分割暗示结合起来。

[0065] 为了评估在特定背景下该方法或系统的准确性, 例如作为系统测试的一个形式或作为校准, 我们可以从一个测试媒介的布局中提取低级的网络表, 并将一个或多个上述的算法应用与比较。

[0066] 根据本发明的不同实施方法, 下列内容提供同本主题相关的不同例证, 识别实施本发明的各个方面、步骤和考虑。

[0067] 在一个实施方法中, 为了自动或半自动地识别或说明集成电路中不同组成部分或方面的特性, 我们使用现有或者独家的元素库去查寻已经被识别的组成部分, 并说明它们的特性。为了有效地查寻和说明已经识别的组成部分的特性, 我们设定了标准去界定构成元素库条目的匹配, 同时证明一系列相关的因素, 包括计算的负荷和时间、在一个特定的适用查寻空间的可行性、潜在的匹配多样性和查寻要求等。如有可能的话, 要说明对于一项应用或所需的结果, 这种查寻是否有必要。

[0068] 为了实行结构数据的开采, 本发明使用了标准单元 (如参考网络表) 的元素库。标

准单元元素库通常包含 50 至 250 这种单元,可以包括带普通布局(如标准高度和位于 Vdd 和 Vss 之间的矩形)的 30 个 PMOS/NMOS 晶体管。在多种情况下,在同一个设计中,各个单元会多次得到使用。同样,根据本发明的一个实施方法,我们可以从具有相应单元级的网络表中再次大量制作单元元素库。

[0069] 我们可以使本发明适应,使之用于各种不同标准单元的匹配过程。从理论上讲,标准单元的匹配就是一个直截了当的精确匹配过程。然而在实践中,使用真实的逆转工程布局,在有些实施方法中,我们通常使用一个近似表去照顾到 GDSII 形状的变异、缺失的功能、各图像层之间(例如井坑和扩散)的错位、路由的单元区的应用、光亮和对比度的变异等等。例如,我们可以使本发明适应,使 NOR 大门的变异和在 32 比特的 DVD 微控制器中发现的倒置输入单元相匹配。

[0070] 我们可以使本发明适应,使之为找到集成电路标准单元的查寻空间效力。查寻空间指的是集成电路单元级网络表的范围。在这个范围内,我们可以识别功能模块。在最坏的情形中,标准单元的查寻空间包含图像的像素数。然而在实践中,通过预先识别标准单元的行,建立大概的单元界限,对齐突出的特性(如角落和垂直互连通路)等,我们可以减少这种查寻空间,从而通过加快标准单元的提取获得更多易处理的查寻空间。一般来讲,查寻空间包含标准单元列阵的多个行或栏。

[0071] 一般来讲,模块由标准单元组建。例如,简单的模块会包括,但不限于,一个 4 位输入的加法器、一个 4 位输入的 MUX、一个 8 位输入的存储器等。较为复杂的模块包括,但不限于,一个 32 比特的加法器、一个乘法器和多位元移位器等。

[0072] 在建立模块元素库时,根据应用项目的情况,可以应用各个方面和 / 或复杂组合。例如,在有些设计和 / 或者某个特定的目标中,一个特定的模块(或者参考网络表)只可以出现一次。同样,由于合成工具、单元元素库和选择的参数、速度、区域和动力等的不同,大门级变异很多,因此某个特定的模块只能在行为级明确地加以定义,如 16 比特的加法器。也可能出现额外的电路,如排错 / 扫描电路、额外的缓冲以及高 / 低设定的输入等。此外,有些实例中缺乏标准的布局,因此大门可以和其它模块的大门相连。一般来讲,这些模块可能会比较复杂,包括各类单元的 100 多个组成部分。

[0073] 考虑到这些复杂因素,并根据本发明的一个实施方法,我们可以在大门级网络表中获取和 / 或识别模块的表述特性。例如,在一个实施方法中,某个模块可以直接用图形来表述,带有 N 个节点(大门),M 个边缘(电线),边缘可以有方向(如输入或输出)。使用这个方法,我们考虑识别模块的各种方法和过程。一个方法是图形匹配,它通常要求使用定义完美的查寻形式去找到匹配。通常讲,图形匹配具有最糟的匹配评估时间,其比例是 M^2 。另一个方法是使用形式匹配,这样任何一个节点可以同图中的任何别的节点相连接,因此通常就不存在查寻形式,因为每个节点的可能通道都必须得到评估,导致最糟的处理时间,其比例为 N。另一个方法是使用模块查寻空间,这要求解决子图同形的问题并且在大图中找到小图。模块查寻空间方法可能会是一个很复杂的问题,因为它在多项式时间中或者在数学词汇中,都没有已知的解决办法,一个无法控制的问题。

[0074] 总结起来,有些模块识别问题的困难包括,从一个特定的目标中建立完整的元素库,有可能的匹配造成潜在的组合数爆炸式增长,带有模块和元素库规模的查寻空间,以及在行为级匹配的难度等。在行为级里,目标只有在大门级才得到说明。

[0075] 同样,根据本发明的一个实施方法,我们提供了一个包含合成设计的元素库,例如,包含已知功能的模块,使用不同的合成工具和 / 或方法,尽量要做得彻底,建立高效的查寻,使用结构数据开采将元素库和目标加以匹配。在一个实施方法中,我们可以使用模块代替具有匹配特性的目标网络表中的相应部分,并且可以在新产生的网络表中加注。

[0076] 根据本发明的一个实施方法,结构数据开采过程涉及在目标集成电路中选种,培养种子和分析匹配的种子。为了便于这个说明,种子是小型、重复性的、非同一般的电路小片,如 ALU 模块等。培养种子的过程始于在同模块元素库匹配的目标集成电路中选择初始的种子。一开始选择小种的优点是计算不会太昂贵。在这个阶段,元素库中起初的匹配不能保证模块的匹配,有可能出现和元素库的多种匹配。

[0077] 在一个方面,结构数据开采允许采用比较实际的方法去寻找元素库和目标之间有意义的结构匹配。

[0078] 参照图 5,并根据本发明的一个方面,培养种子过程的第一步是在目标集成电路中选择初始种子。下一步,这个过程涉及将从模块元素库选出的种子进行匹配。模块元素库可以是任何现有的元素库,包括从目标集成电路中产生的元素库。选择初始种子和初始匹配的步骤可以重复多次,可以是按顺序重复,也可以平行重复,以便获取一套匹配的种子。这套匹配的种子应能表述目标集成电路和模块元素库之间的候选匹配。如果在种子和模块元素库之间有匹配,下一个过程就是培养种子。

[0079] 参考图 6。等到目标集成电路和模块元素库之间的匹配种子找到之后,就可以将大门加进种子,或者将临近匹配的种子加以合并和重叠,梯次培养匹配的种子。这里说明的培养种子的过程和步骤可以继续进行,直到一个元素库组成部分完成,或者直到再也找不着元素库的匹配。

[0080] 以图 6 为例。种子 A_0 是个初始种子,它表述一个小型电路小片,在目标集成电路中它出现次数为 N10、N20 和 N50 等。 A_0 同元素库的初始匹配是模块是 L1、L2、L3、L4 和 L5。下一步, A_0 种入种子 A_1 ,包括了额外的大门或匹配的种子。其结果是,种子 A_1 代表了一个不同的、更大的电路小片,在目标集成电路中它出现的次数为 N10、N50 和 N55 等。相应地,种子 A_1 同模块 L1、L3 和 L5 匹配。种子的生长继续,直到种子 A_n 的元素库中再也找不到匹配。

[0081] 对于种子匹配的花费而言,选择合适的起始种子和培养方法是重要的考虑。同样,在本发明的一个实施方法中,我们可以将选种过程加以适应,使之可以识别优化的起始种子并从中进行操作,并且采用高效的培养策略。例如,在一个培养策略中,可以使用分割暗示去缩减培养时间。还有,根据大种子优于小种子以及出现次数少的种子优于出现次数频繁的种子这个工程原理,优化培养法可以应用到选择候选种子。换句话说,有较多至高点和边缘的种子将被选择,用来生长取代较少至高点或边缘的种子;在集成电路出现不太频繁的种子将被选择,用来生长取代在集成电路中出现较为频繁的种子。

[0082] 正如能被对集成电路工艺具有一般技能的人所领会的那样,上述各点比起蛮力培养种子的方法有几个有点。这些优点包括,但不限于,只培养匹配的种子就减少了目标中的查寻空间;有效地减少了候选元素库匹配的数量;能够预先计算元素库,以便减少计算时间。

[0083] 根据本发明的各种实施方法,在选种时需要考虑几种可能性,其中包括单门、两个相连的有些独特的门、直接相邻(例如锥形)和布尔功能等非结构性的模块。选择优化种

子可以基于不同的考虑。例如，在选择最优化种子时，可以根据所选种子的共性（例如太通常的种子将不足以限制候选模块）和独特性（例如，越独特越好），需做几种考虑。

[0084] 然而，另一个方法是从网络表中甄选一小部分，考虑到模块的几个输入和输出，或者时钟及其重组信号等，该网络表很可能就是一个功能块。通过计算布尔真理表或两进制精确图（BDD）并将杂乱脉冲干扰应用到布尔真理表或两进制精确图中，甄选的种子将会受到逻辑功能特性的制约。然后我们就可以使用布尔功能去取代组成布尔功能网络表的那些部分。那个布尔功能具有单一的表述性并具有布尔功能的特性标签。因此这个甄选种子的方法就能促进网络表格小片的识别，这些小片具有相同的逻辑但有不同的大门级实施状况。图 7 这个示意图显示用于布尔功能 3302 的 BDD3301。

[0085] 根据上述的选种策略，并根据本发明的一个实施方法，如 16 比特加法器、16 比特 20MUX 和 16 比特解码器等行为型种子，可以被认为是优化种子。在一个例子中，许多不同的种子从目标集成电路中得到计算。这些种子在单个电路家族内具有高出现频率，在其它家族中具有较低出现频率，它们被选来形成行为型种子。这些行为型种子可以储存在单独的元素库内，用于以后同现有的模块元素库进行匹配。

[0086] 在一个实施方法中，我们可以使这个方法和系统适应，去识别非元素库的元素和 / 或不够精确的匹配。例如，非精确匹配的定义仍然要使用元素库获得。举一个例子，我们可以试图用多种定义 / 试探法来做非精确的匹配并查寻整个元素库；好的定义可以匹配不同模块的不同版本，而不好的定义一下子可以匹配不同的模块。在一个例子中，元素库中多个例证同形为种子匹配，用以识别功能和一些输入输出。探索法也可以用来寻找同类元素，分割暗示可以用来制约新的模块。

[0087] 根据本发明的一个方面，我们可以将各种结构数据开采技术应用到上述描述的各种选种过程，也可以应用到目标集成电路和各种模块元素库中去。为了促进目标开采过程，图 9 提供了一般性排列结构的示意图，其中每个种子和目标集成电路中一组电路小片的出现相连，也同模块元素库中一组模块的出现相连。

[0088] 在本发明的一个实施方法中，种子可以进一步或者有选择性地得到杂乱脉冲干扰的特性。元素库和目标可以由提取的种子打上索引，也可以用储存在数据结构中它们的杂乱脉冲干扰来打索引。在这样的实施方法中，通过杂乱脉冲干扰手段或其它合适的方法，在排列的数据结构中寻找匹配的种子，这样就可以产生初始的匹配。

[0089] 根据本发明的一个方面，图 10 和图 11 这两个示意图，表述在目标集成电路中寻找电路结构的目标开采例证。图 10 显示我们怎样使用选种过程识别潜在的新模块或者在目标集成电路中寻找参考功能块。在种子匹配和生长过程的末尾，相应的种子 A_n 在模块元素库中没有了匹配，这时一个潜在的新模块就被识别出来。

[0090] 参考图 11 并根据本发明的一个实施方法，我们在匹配和培养的过程中使用 16 比特的 ALU 行为种子。在这种情况下，选种过程已经在种子 A_1 下的元素库中识别到了所有现存的 16 比特 ALU 模块。在这个过程的末尾，一个没有匹配的种子意味着一个新的 16 比特的 ALU 已被识别出来。

[0091] 如上所述，本文透露的结构数据开采过程涉及到通过杂乱脉冲干扰手段说明带标签图形的特性。根据本发明的一个方面，图 12 从特性的角度显示了前面说明的选种过程。换句话说，下面的过程针对目标的图形表述或需要开采的元素库，说明了选种的过程。这个

过程始于从目标图像模型中选择初始的子图，并计算子图的杂乱脉冲干扰值 3501。下一个过程涉及确定元素库 3502 中是否存在杂乱脉冲干扰值。如果元素库中能找到这个杂乱脉冲干扰值，那么这个过程就培养出了所选的子图 3503。我们接着将会对此作说明。然而，如果在元素库中找不到这个杂乱脉冲干扰值，那么我们就可以摈弃这个所选的子图，因为它无助于识别网络表中的功能块。选择初始子图以及检查初始的杂乱脉冲干扰这些步骤可以重复几次，可以按顺序重复，也可以平行重复，以便获得能够代表目标集成电路和模块元素库之间匹配子图的一套匹配的杂乱脉冲干扰值。

[0092] 参阅图 13。等到在模块元素库中找到了子图的杂乱脉冲干扰值后，我们可将另一个子图 3402 或 3401 加入到所选的子图中，或者同临近带有匹配杂乱脉冲干扰的子图合并或重叠，形成一个更大的相连子图 3403。通过这个办法，我们就可以梯次培养子图 3401 和 3402。只要还有子图需要培养，本文说明的寻找特性过程及其所有的步骤可以不断进行。图 12 显示的这一过程的 3504 步骤，元素库中相关的信息同所找到的杂乱脉冲干扰值相结合，可以用来识别目标集成电路中网络表的功能块。

[0093] 参阅图 14，并根据本发明的一个实施方法，在子图生长的过程中，可以使用一种替代性的杂乱脉冲干扰算法说明带标签图形的特性。在 $M \log(d)$ 的运算中，替代性的杂乱脉冲干扰算法能够说明带 M 边缘和直径 d 。子图的特性被其所有至高点的杂乱脉冲干扰说明。这个过程涉及重复子图中的每个至高点，并说明相邻至高点的特性。在这个过程的开始，一个至高点被其标签的杂乱脉冲干扰说明特性，紧接着这个至高点被自身现有的杂乱脉冲干扰、关联边缘的标签和临近至高点现有的杂乱脉冲干扰等说明了特性。这个替代性杂乱脉冲干扰的效果是，在每个重复循环之后，有效相邻的直径就增加 2 倍。

[0094] 在本发明的一个实施方法中，匹配的标准包括编辑图中的距离、计算子图中的杂乱脉冲干扰、包括杂乱脉冲干扰出现的数据次数、计算小型布尔功能的杂乱脉冲干扰以及它们的数据在出现次数，等等。

[0095] 在子图的生长过程中，跨越功能块界限的子图通常会使不匹配杂乱脉冲干扰数量大幅度增加。要限制一个子图扩展成为另一个功能块，必须给某些连接较低的优先权，例如动力的连接或排错信号等，有好处的，除非这些信号很小并有可能就处在本身模块。另外，应该给“模块内”（如小型反馈圈）的信号较高的优先权。

[0096] 此外，作为一种替补手段，可以将分割探索法用于指导计算杂乱脉冲和匹配的过程。探索法包括循环探测、作为界限的储存器、时钟树状电路的分布、测试电路、动力分布和重要的 / 至关重要的信号（即少量依靠或影响许多方面的信号，例如“yaggle”工具）。

[0097] 在一个实施方法中，我们将数据列加以组织，将每个图形的杂乱脉冲干扰组成图表，列出各自的出现次数。

[0098] 在本发明的一个实施方法中，在结构数据开采的过程中，我们计算了一个图形的杂乱脉冲干扰（用于精确的匹配），如果杂乱脉冲干扰相匹配，就宣布它们为同类，无需进行大门与大门之间的匹配。

[0099] 对于不精确的匹配，杂乱脉冲干扰（即具体的子图）可以组织成为一个表格式的数据结构，即 Sergei Brin 所说的数据树。

[0100] 在本发明的一个实施方法中，除了布局的匹配外，我们还对网络表进行了匹配。例如，对照布局领域的相互关系，可以对这个系统进行配置去识别好的匹配，也就是为匹配进

行有效的组合查寻算法,以此对照在布局领域中方便地寻找位置。

[0101] 此外,作为一种替补方法,我们可以将本文说明的系统和方法加以配置,产生元素库与元素库之间的匹配,即具有模块和功能块(即参考网络表)特点的数据矿形式。举几个例子,这类匹配可以用来更好地界定匹配的标准,调节(即调准)算法和探索法,评估元素库的可消耗性,并且提供两次没有偏见的测试。

[0102] 此外,作为一种替补方法,我们可以将本文说明的系统和方法加以配置,进行元素库和目标网络表的匹配,也就是说,识别存在于元素库中已知功能的模块。

[0103] 此外,作为一种替补方法,我们可以将本文说明的系统和方法加以配置,进行一个目标网络表与另一个目标网络表的匹配,也就是说,识别存在于元素库中已知功能的模块。

[0104] 此外,作为一种替补方法,我们可以将本文说明的系统和方法加以配置,实施布尔模块的匹配。例如,两进制的精确图可以作核对之用。它们的制作和比较相对高效。虽然它们通常纠错能力较差(例如,由于人工制品的缘故,在逆转工程中出错难以避免),但对于较小的模块它们特别适用。在另一个实施方法中,我们通过一个组合查寻空间,提供了一个预先识别的输入和输出,用来识别模块的边界,从而起动或促进布尔模块的匹配。图7是布尔模块匹配的示意图。

[0105] 如上所述,在本发明各种示意实施方法中,标准单元级网络表可以提取,或者作为起点使用,或者至少与一些分割或分割暗示一起使用。因此我们可以提供不同的方法和系统,获得更高级的网络表,以便让大部分,或者至少一组重要的模块(即算术算法和储存器等)可以组织起来,并且获得它们的行为说明。

[0106] 很显然,本发明前面的实施方法只是一些例子,可以用不同的方法实施。我们不应认为这类目前和未来的实施方法变化有违本发明的精神和范围。这样的变化对于熟知集成电路的人们来说是显而易见的,它们应该包括在下面的权利要求的范围之内。

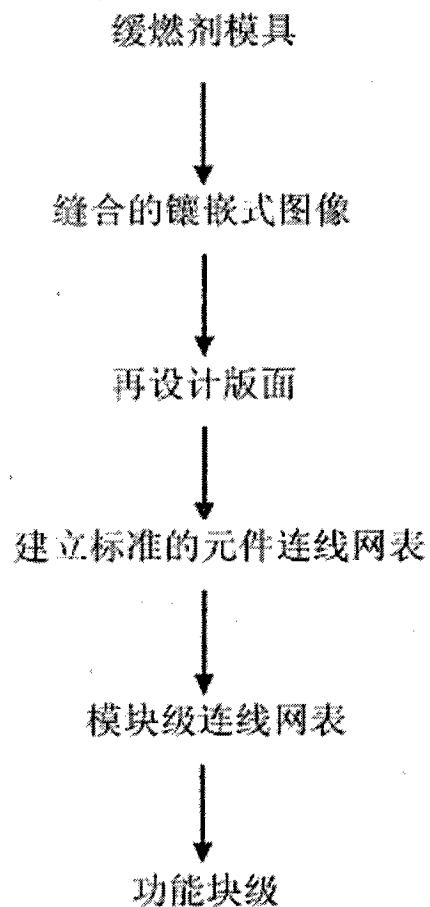


图 1

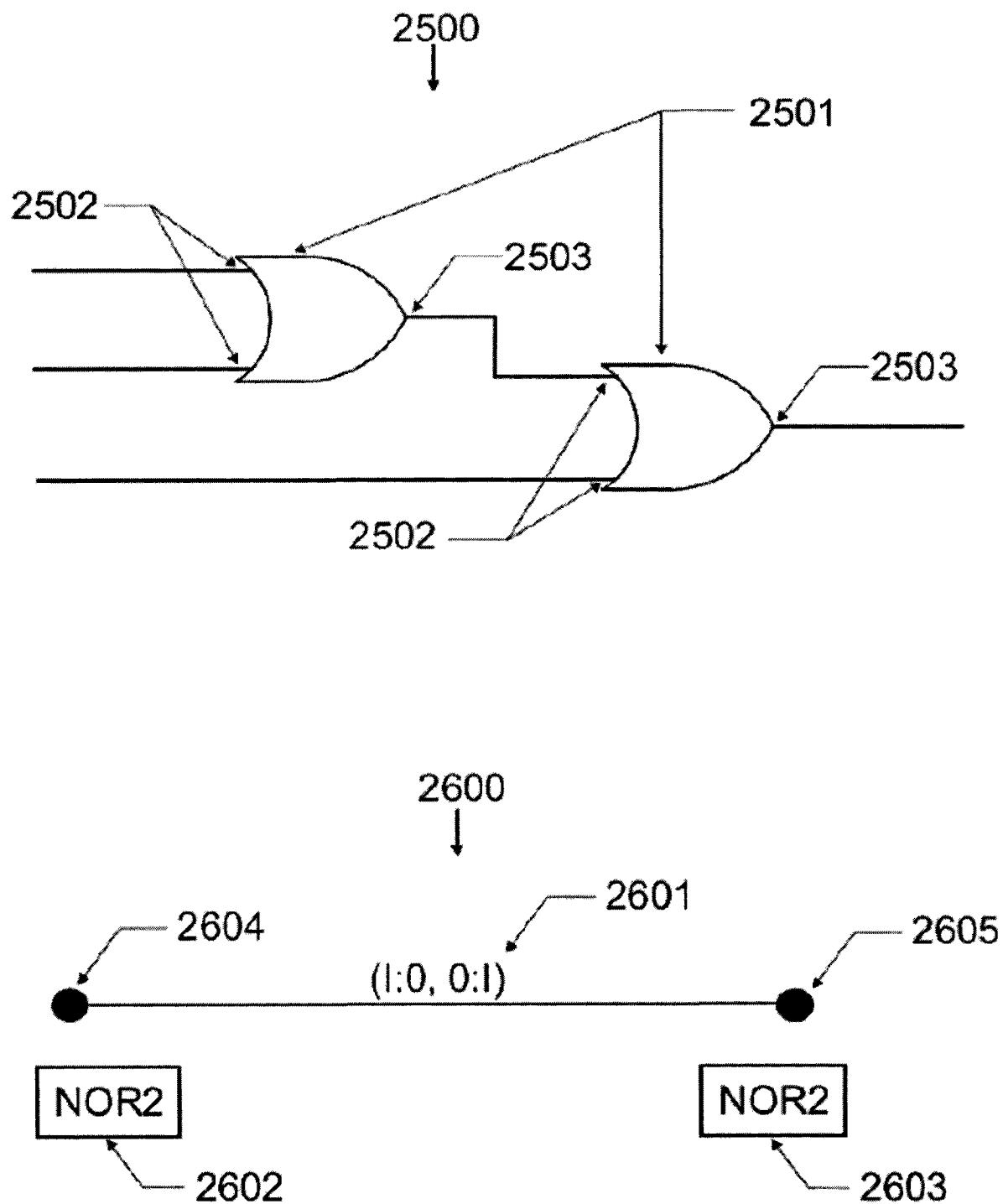


图 2

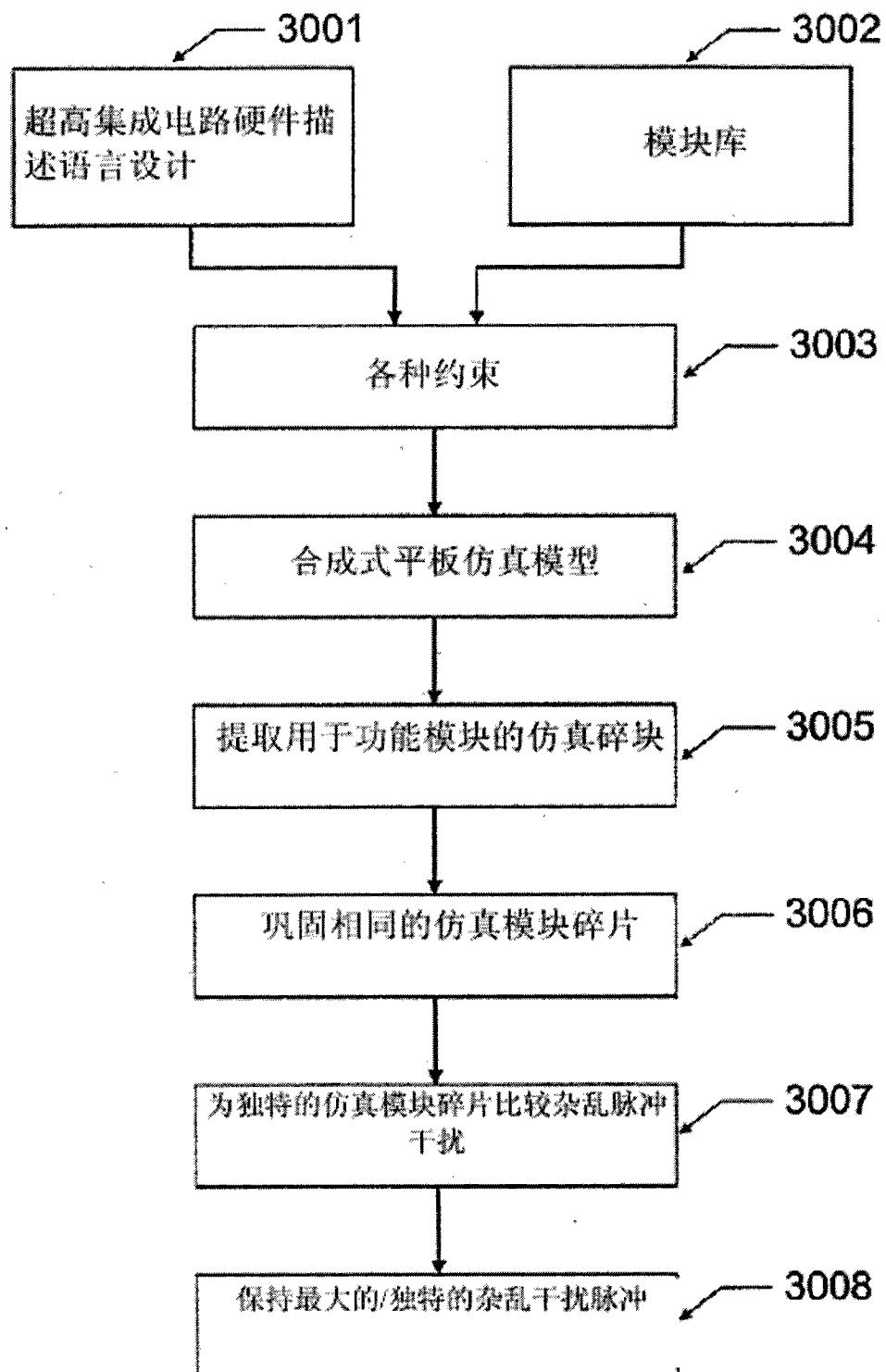


图 3

3202				
杂乱干扰脉冲值 64 比特	地点	数量	前身	模块种类
257	1. 储存器转换级 (I17), 1. 储存器转换级 (I18) ...	75		加法器 - 75
258	1. 储存器转换级 (I40), 1. 储存器转换级 (I41) ...	81		加法器 - 75
:				
493	1. 储存器转换级 (I17, I40), 1. 储存器转换级 (I41, I18)	70	257, 258	加法器 - 75
903	1. 储存器转换级 (I17, I40, I44)	2	493	

3201 3203

图 4

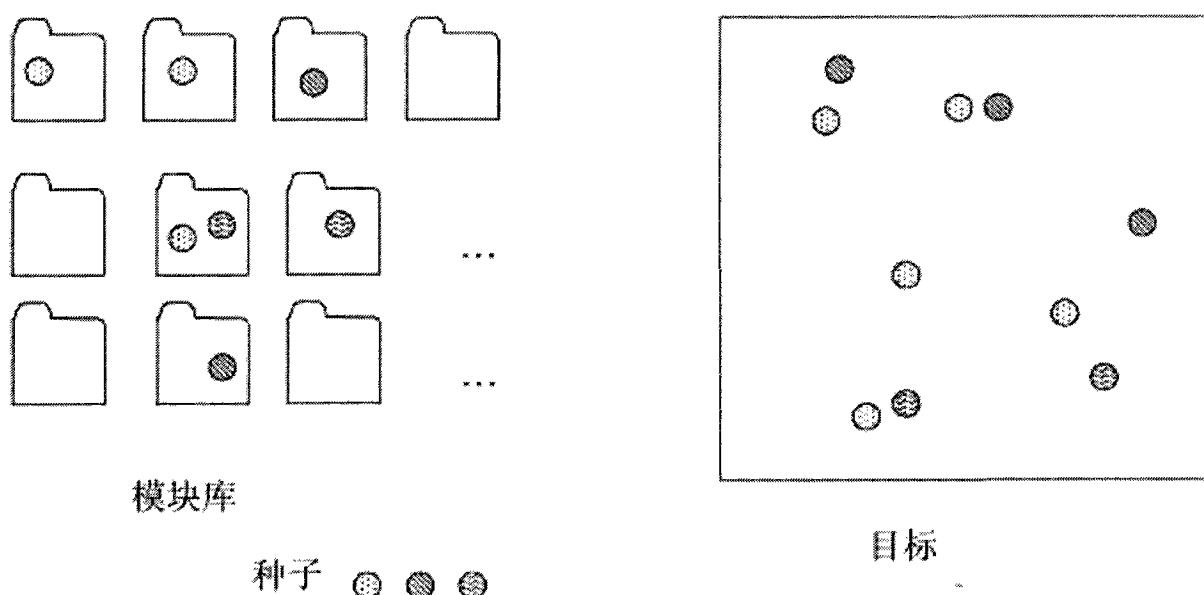
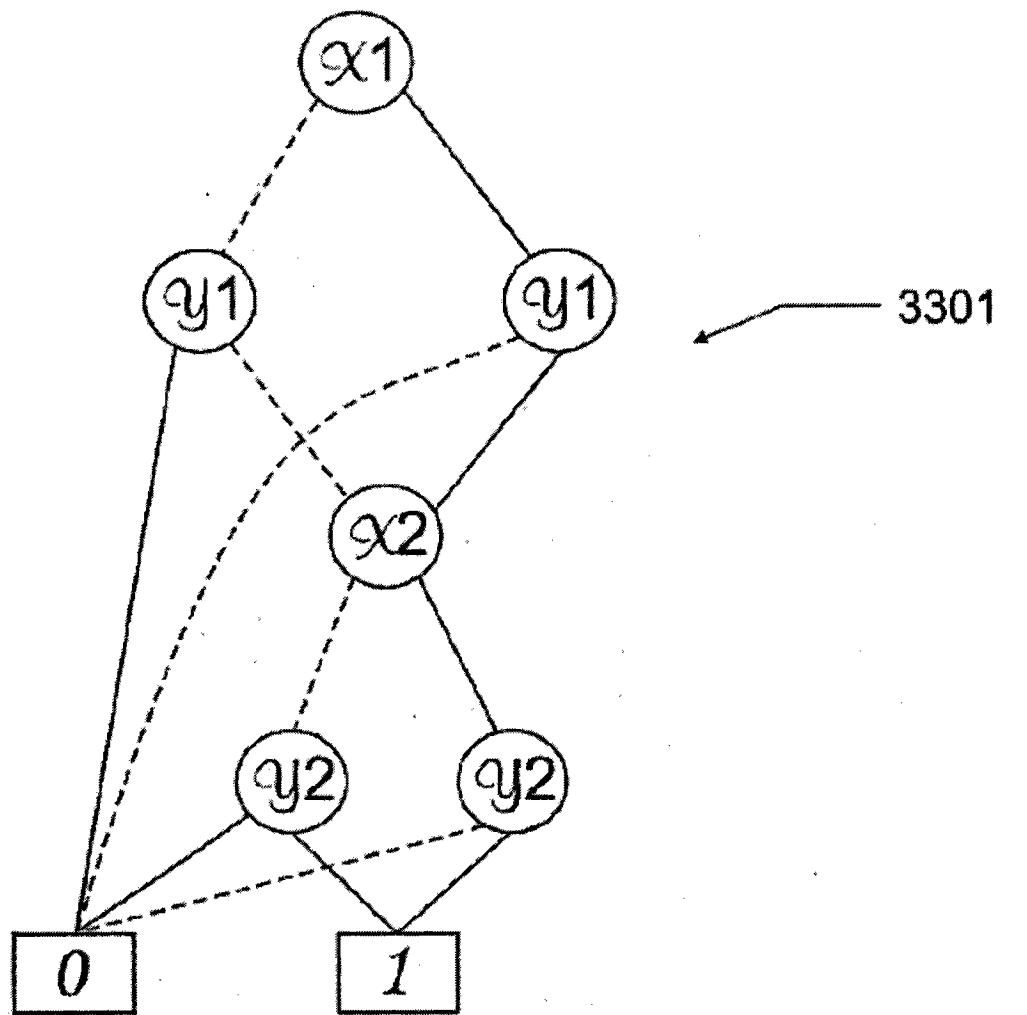


图 5

	库存次数	目标次数
种子 A ₀	L1, L2, L3, L4, L5	N10, N20, N50, ...
种子 A ₁	L1, L3, L5	N10, N50, N55, ...
种子 A ₂	L1	N10, N50, ...
...		
种子 A _n	完成	N10, N50

图 6



A BDD for $(x_1 = y_1) \&\& (x_2 = y_2)$ ← 3302

图 7

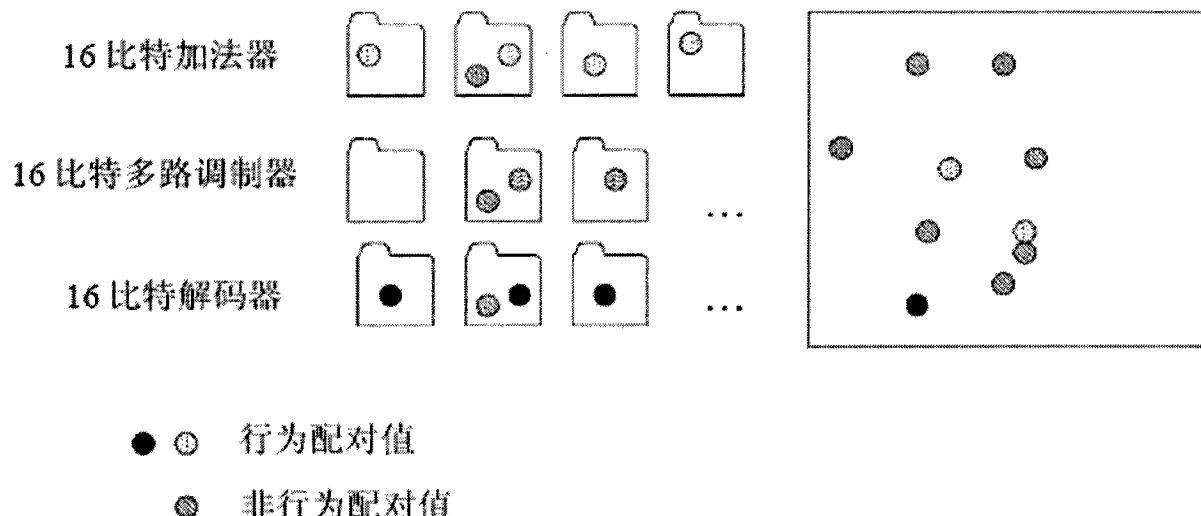


图 8

种子 1	目标中的表格次数	库存中的表格次数
种子 2	目标中的表格次数	库存中的表格次数
种子 3	目标中的表格次数	库存中的表格次数
种子 4	目标中的表格次数	库存中的表格次数
种子 N	目标中的表格次数	库存中的表格次数

图 9

	库存次数	目标次数
种子 A ₀	L1, L2, L3, L4, L5	N10, N20, N50, ...
种子 A ₁	L1, L3, L5	N10, N50, N55, ...
种子 A ₂	L1	N10, N50, ...
种子 A _n	零位	N10, N50, ...

新模块?

图 10

	库存次数	目标次数
种子 A ₀	L1, L2, L3, L4, L5	N10, N20, N50, ...
种子 A ₁	L1, L3, L5	N10, N50, N55, ...
种子 A ₂	L1	N10, N50, ...
种子 A _n	零位	N10, N50, ...

所有 16 比特算术逻辑部件运算器

所有 16 比特算术逻辑部件运算器

图 11

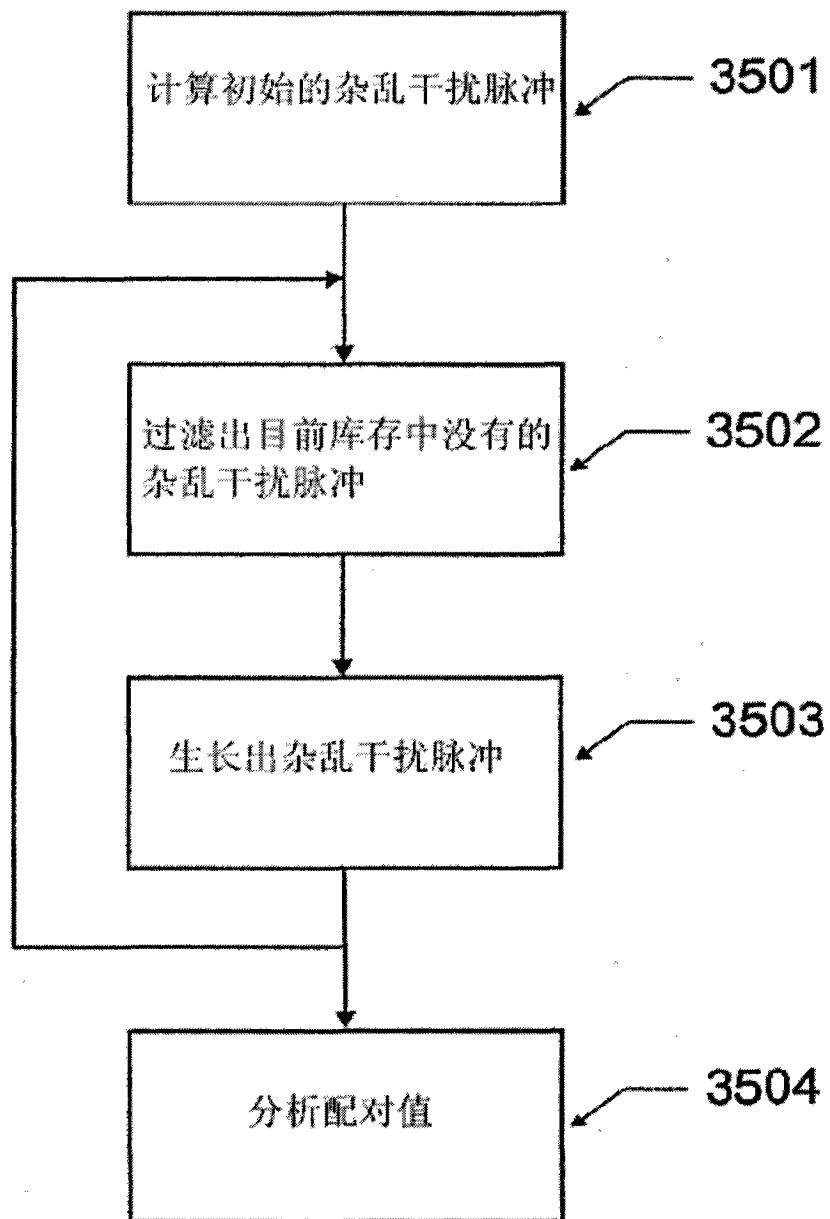


图 12

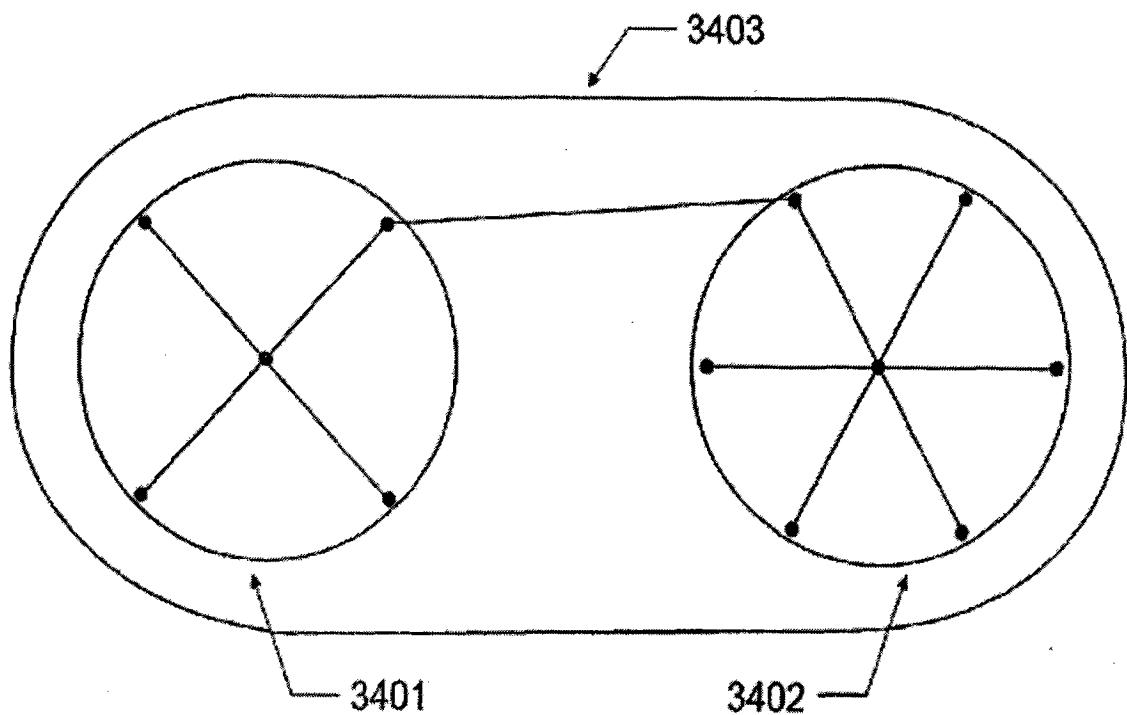


图 13

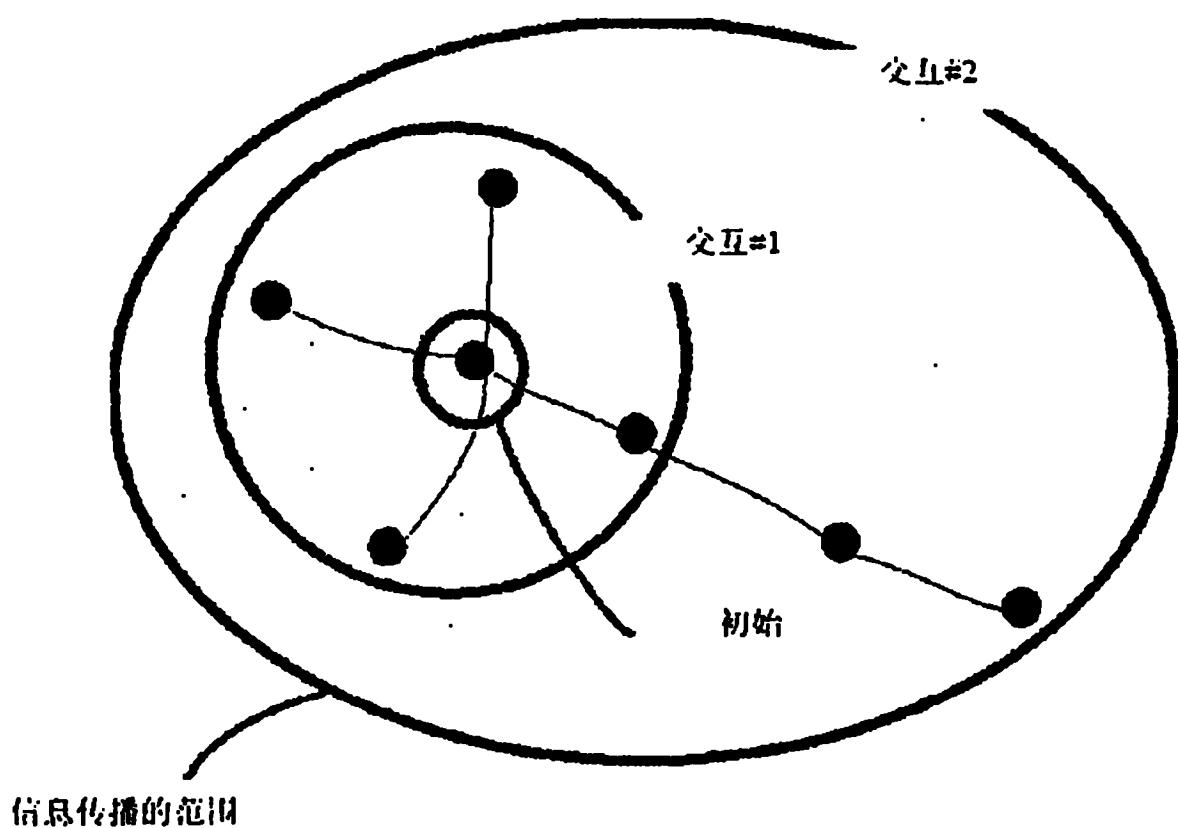


图 14