



(12) 发明专利申请

(10) 申请公布号 CN 104112033 A

(43) 申请公布日 2014. 10. 22

(21) 申请号 201410155402. 9

(22) 申请日 2014. 04. 17

(30) 优先权数据

13/866, 624 2013. 04. 19 US

(71) 申请人 奥特拉有限公司

地址 美国加利福尼亚

(72) 发明人 A·泰特利 D·S·高曼

(74) 专利代理机构 北京三友知识产权代理有限公司 11127

代理人 吕俊刚 刘久亮

(51) Int. Cl.

G06F 17/50(2006. 01)

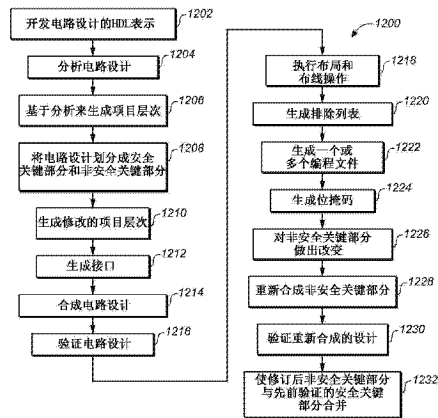
权利要求书3页 说明书20页 附图11页

(54) 发明名称

促进认证的划分设计

(57) 摘要

本发明涉及促进认证的划分设计。本发明一般地涉及现场可编程门阵列(FPGA)。一些实施方式涉及用于划分FPGA电路设计以促进认证的方法和系统。在一个方面,一种方法包括生成电路设计的硬件描述语言(HDL)实施方式。该方法另外地包括将设计划分成第一部分和第二部分。在一些实施方式中,第二部分对应于设计的安全关键部分然而第一部分对应于非安全关键部分。该方法另外地包括为第一部分生成第一配置设定以及为第二部分生成第二配置设定。该方法另外地包括验证用于第一部分的第一配置设定和用于第二部分的第二配置设定或者将其提供给第三方认证机构以用于验证。该方法进一步包括提供用于第二部分的配置设定以用于编程到PLD中。



1. 一种方法,该方法包括以下步骤:

生成要被实现在可编程逻辑器件 PLD 上的电路设计的硬件描述语言 HDL 实施方式;

将所述电路设计划分成第一部分和第二部分,所述第一部分包括用于耦接所述第一部分和所述第二部分的接口,所述第二部分是包括要求由独立认证机构认证为满足功能安全要求的功能的安全关键部分;

基于所述 HDL 实施方式生成用于所述第一部分的第一配置设定;

基于所述 HDL 实施方式生成用于第二部分的第二配置设定;

至少验证用于第二部分的所述第二配置设定或者至少将用于第二部分的所述第二配置设定提供到第三方认证机构以用于验证;以及

提供用于第二部分的经验证的第二配置设定以用于编程到 PLD 中。

2. 根据权利要求 1 所述的方法,所述方法进一步包括以下步骤:

对所述第一部分进行修订;

生成用于修订后的第一部分的第一配置设定;以及

将用于所述修订后的第一部分的第一配置设定与用于第二部分的所述经验证的第二配置设定合并,以及

其中,提供用于第二部分的经验证的第二配置设定以用于编程到所述 PLD 中的步骤包括为所述修订后的第一部分和所述第二部分两者提供经合并的配置设定以用于编程到所述 PLD 中。

3. 根据权利要求 2 所述的方法,其中,用于第二部分的所述配置设定被以编程位的形式生成,并且其中,用于第二部分的所述编程位作为第一编程文件的一部分被包括。

4. 根据权利要求 2 或 3 所述的方法,其中,用于所述修订后的第一部分的第一配置设定被以编程位的形式生成,并且其中,用于所述修订后的第一部分的第一配置设定作为第二编程文件的一部分被包括。

5. 根据权利要求 4 所述的方法,所述方法进一步包括生成位掩码,所述位掩码包括其中每一个位都具有第一逻辑值的第一组位和其中每一个位都具有第二逻辑值的第二组位,所述第一组位中的每一个位都处于所述位掩码的如下位置中,即,该位置与所述第一编程文件或所述第二编程文件中的被第二部分使用的编程位相对应,其中,将用于所述修订后的第一部分的第一配置设定与用于第二部分的所述经验证的第二配置设定合并的步骤包括使用所述位掩码来将所述第一编程文件与所述第二编程文件合并以生成合并的编程文件。

6. 根据权利要求 5 所述的方法,所述方法进一步包括使用所述位掩码来验证用于第二部分的所述编程位在经合并的编程文件中尚未被变更。

7. 一种由如下的过程形成的可编程逻辑器件 PLD,所述过程包括以下步骤:

生成要被实现在可编程逻辑器件 PLD 上的电路设计的硬件描述语言 HDL 实施方式;

将所述电路设计划分成第一部分和第二部分,所述第一部分包括用于耦接所述第一部分和所述第二部分的接口,所述第二部分是包括要求由独立认证机构认证为满足功能安全要求的功能的安全关键部分;

基于所述 HDL 实施方式生成用于第一部分的第一配置设定;

基于所述 HDL 实施方式生成用于所述第二部分的第二配置设定；

至少验证用于所述第二部分的所述第二配置设定或者至少将用于所述第二部分的所述第二配置设定提供到第三方认证机构以用于验证；以及

提供用于所述第二部分的经验证的第二配置设定以用于编程到 PLD 中。

8. 根据权利要求 7 所述的 PLD, 其中, 所述过程进一步包括以下步骤:

对所述第一部分进行修订;

生成用于修订后的第一部分的第一配置设定; 以及

将用于所述修订后的第一部分的第一配置设定与用于所述第二部分的所述经验证的第二配置设定合并, 以及

其中, 提供用于所述第二部分的经验证的第二配置设定以用于编程到所述 PLD 中的步骤包括为所述修订后的第一部分和所述第二部分两者提供经合并的配置设定以用于编程到所述 PLD 中。

9. 根据权利要求 7 或 8 所述的 PLD, 其中, 用于所述第二部分的所述配置设定被以编程位的形式生成, 并且其中, 用于所述第二部分的所述编程位作为第一编程文件的一部分被包括。

10. 根据权利要求 9 所述的 PLD, 其中, 用于所述修订后的第一部分的第一配置设定被以编程位的形式生成, 并且其中, 用于所述修订后的第一部分的第一配置设定作为第二编程文件的一部分被包括。

11. 根据权利要求 10 所述的 PLD, 其中, 所述过程进一步包括生成位掩码, 所述位掩码包括其中每一个位都具有第一逻辑值的第一组位和其中每一个位都具有第二逻辑值的第二组位, 所述第一组位中的每一个位都处于所述位掩码的如下位置中, 即, 该位置与所述第一编程文件或所述第二编程文件中的被所述第二部分使用的编程位相对应, 并且其中, 将用于所述修订后的第一部分的第一配置设定与用于所述第二部分的所述经验证的第二配置设定合并的步骤包括使用所述位掩码来将所述第一编程文件与所述第二编程文件合并以生成合并的编程文件。

12. 根据权利要求 11 所述的 PLD, 其中, 所述过程进一步包括使用所述位掩码来验证用于所述第二部分的所述编程位在经合并的编程文件中尚未被变更。

13. 一种方法, 该方法包括以下步骤:

生成要被实现在可编程逻辑器件 PLD 上的电路设计的硬件描述语言 HDL 实施方式;

将所述电路设计划分成第一部分和第二部分, 所述第一部分包括用于耦接所述第一部分和所述第二部分的接口;

生成所述第二部分的网表表示;

验证所述第二部分;

在验证所述第二部分之后, 对所述第一部分进行修订;

生成修订后的第一部分的网表表示; 以及

将经先前验证的第二部分的所述网表表示与所述修订后的第一部分的所述网表表示合并。

14. 根据权利要求 13 所述的方法, 其中, 所述第二部分是包括要求认证为满足功能安全要求的功能的安全关键部分。

15. 根据权利要求 13 或 14 所述的方法,所述方法进一步包括以下步骤:

生成包括用于所述修订后的第一部分和所述第二部分两者的配置设定的编程文件;

生成位掩码,所述位掩码包括其中每一个位都具有第一逻辑值的第一组位和其中每一个位都具有第二逻辑值的第二组位,所述第一组位中的每一个位都处于所述位掩码的如下位置中,即,该位置与所述编程文件中的被所述第二部分使用的编程位相对应;以及

使用所述位掩码来验证用于所述第二部分的所述编程位在所述编程文件中尚未被变更。

## 促进认证的划分设计

### 技术领域

[0001] 本公开一般地涉及现场可编程门阵列(FPGA),并且更具体地,涉及用于划分 FPGA 电路设计以促进认证的方法和系统。

### 背景技术

[0002] 在许多现代应用中,FPGA 电路开发者仅设计整个 FPGA 设计的一部分。例如,一个 FPGA 电路开发者能够设计处理器核,另一 FPGA 开发者能够设计存储器块,另一人能够设计存储器控制器,以及然而其它人能够设计 FPGA 设计的其它部分。各种电路开发者可以作为不同团队的一部分为单个实体或者为不同的实体(例如,为不同的商业企业中的不同的电路开发者)而工作。例如,因为设计和验证电路设计的困难和费用以及设计的“商用可能”,设计实体可能期望使电路设计保密,或者至少期望不公开,或者将电路设计的公开限制于诸如其它开发者、用户、设计者或客户的其它方。半导体知识产权核或简单地“IP 核”或“IP 块”能够指的是逻辑的单位、单元、芯片布局,或者另外指的是要被作为更大 FPGA 电路设计的一部分与其它电路相结合地使用的电路设计。IP 块是特定电路开发者的设计,并且典型地是特定电路开发者的知识产权。IP 开发者(或“设计者”)可以将 IP 块的使用出售或者授权给之后将 IP 块实现到用户自己的更大设计中的另一开发者或用户,该用户自己的更大设计它本身可以是更大设计的一部分。

[0003] 尽管 IP 块能够作为网表被发送到其它方,诸如 FPGA 的用户,但是 IP 块典型地在寄存器传送级(RTL)级被发送,诸如以可合成硬件描述语言(HDL)实现的设计的形式。因为设计被以 HDL 形式传递,所以用户能够在功能级修改设计。然而,如上所述,因为开发者可能因为验证所牵涉的时间和费用或由于其它原因而希望使设计保密(例如,作为商业秘密),所以开发者可能不想用户能够具有对电路设计的功能描述的这样的访问。例如,因为如果设计被修改了则电路开发者无法提供对 IP 块的授权或支持,所以电路开发者可能期望防止 FPGA 用户能够访问电路设计的功能描述。将电路设计作为软件网表提供给用户能够比以 HDL 形式提供设计提供对逆向工程的更好保护,但仍然能够实现逆向工程。IP 开发者还可以出于开发者自己的好意而简单地期望确保终端用户或客户收到已验证实施方式的利益。另外地,一些 IP 开发者被要求让他们的 IP 块被诸如认证机构(诸如总部设在德国科隆的莱茵 TÜV)的第三方独立地验证以确保集成该设计的产品将满足功能安全要求。例如,认证机构可以认证 IP 块满足电气/电子/可编程电子安全相关系统规范—IEC61508—行业中应用的规则的国际标准的功能安全。这可能是显著地代价高的且常常费时的过程。

### 发明内容

[0004] 本公开一般地涉及诸如现场可编程门阵列(FPGA)的可编程逻辑器件(PLD)。一些实施方式涉及用于划分 FPGA 电路设计以促进认证的方法和系统。在所公开的主题的一个方面,一种方法包括生成要被实现在可编程逻辑器件(PLD)器件上的电路设计的硬件描述语言(HDL)实施方式。所述方法另外地包括将电路设计划分成第一部分和第二部分,第一

部分包括用于耦接第一部分和第二部分的接口。所述方法另外地包括基于 HDL 实施方式为第一部分生成第一配置设定以及基于 HDL 实施方式为第二部分生成第二配置设定。所述方法另外地包括验证用于第一部分的第一配置设定和用于第二部分的第二配置设定或者将其提供给第三方认证机构以用于验证。所述方法进一步包括提供用于第二部分的配置设定以用于编程到 PLD 中。

[0005] 在一些实施方式中,所述方法进一步包括对第一部分进行修订、为修订后的第一部分生成修订后的第一配置设定、以及将用于修订后的第一部分的第一配置设定与用于第二部分的第二配置设定合并。在一些这样的实施方式中,提供用于第二部分的配置设定以用于编程到 PLD 中包括为修订后的第一部分和第二部分两者提供经合并的配置设定以用于编程到 PLD 中。在一些实施方式中,所述第二部分是包括要求认证为满足功能安全要求的功能的安全关键部分。

[0006] 在一些实施方式中,所述方法进一步包括分析 HDL 实施方式以生成电路设计的层次实施方式,其中将电路设计划分成第一部分和第二部分在层次表示级被执行。在一些实施方式中,所述方法进一步包括生成要为第二部分而保留的资源的排他列表。在一些实施方式中,用于第二部分的配置设定被以编程位的形式生成并且其中用于第二部分的编程位作为第一编程文件的一部分被包括。在一些实施方式中,用于修订后的第一部分的第一配置设定被以编程位的形式生成并且其中用于修订后的第一部分的编程位作为第二编程文件的一部分被包括。在一些实施方式中,所述方法进一步包括生成位掩码,所述位掩码包括其中每一个位都具有第一逻辑值的第一组位和其中每一个位都具有第二逻辑值的第二组位,所述第一组位中的每一个位都处于所述位掩码的如下位置中,即,该位置与所述第一编程文件或所述第二编程文件中的被所述第二部分使用的编程位相对应。在一些这样的实施方式中,将用于修订后的第一部分的第一配置设定与用于第二部分的第二配置设定合并包括使用位掩码来将第一编程文件与第二编程文件合并以生成合并的编程文件。例如,在一些实施方式中,当 CAD 工具用位掩码对第一编程文件操作时,工具被使得能够从第一编程文件中提取用于第二部分的编程位。在一些实施方式中,所述方法进一步包括使用位掩码来验证用于第二部分的编程位在经合并的编程文件中尚未被变更。

[0007] 在另一方面,公开了一种可编程逻辑器件(PLD),其由包括生成要被实现在可编程逻辑器件(PLD)器件上的电路设计的硬件描述语言(HDL)实施方式的过程来形成。所述过程另外地包括将电路设计划分成第一部分和第二部分,第一部分包括用于耦接第一部分和第二部分的接口。所述过程另外地包括基于 HDL 实施方式为第一部分生成第一配置设定以及基于 HDL 实施方式为第二部分生成第二配置设定。所述过程另外地包括验证用于第一部分的第一配置设定和用于第二部分的第二配置设定或者将其提供给第三方认证机构以用于验证。所述过程进一步包括提供用于第二部分的配置设定以用于编程到 PLD 中。

[0008] 在一些实施方式中,所述过程进一步包括对第一部分进行修订、为修订后的第一部分生成修订后的第一配置设定、以及将用于修订后的第一部分的第一配置设定与用于第二部分的第二配置设定合并。在一些这样的实施方式中,提供用于第二部分的配置设定以用于编程到 PLD 中包括为修订后的第一部分和第二部分两者提供经合并的配置设定以用于编程到 PLD 中。在一些这样的实施方式中,所述第二部分是包括要求认证为满足功能安全要求的功能的安全关键部分。

[0009] 在一些实施方式中,所述过程进一步包括分析 HDL 实施方式以生成电路设计的层次实施方式,其中将电路设计划分成第一部分和第二部分在层次表示级被执行。在一些实施方式中,所述过程进一步包括生成要为第二部分而保留的资源的排他列表。在一些实施方式中,用于第二部分的配置设定被以编程位的形式生成并且其中用于第二部分的编程位作为第一编程文件的一部分被包括。在一些实施方式中,用于修订后的第一部分配置设定被以编程位的形式生成并且其中用于修订后的第一部分的编程位作为第二编程文件的一部分被包括。在一些实施方式中,所述过程进一步包括生成位掩码,所述位掩码包括其中每一个位都具有第一逻辑值的第一组位和其中每一个位都具有第二逻辑值的第二组位,所述第一组位中的每一个位都处于所述位掩码的如下位置中,即,该位置与所述第一编程文件或所述第二编程文件中的被所述第二部分使用的编程位相对应。在一些这样的实施方式中,将用于修订后的第一部分的修订后第一配置设定与用于第二部分的第二配置设定合并包括使用位掩码来将第一编程文件与第二编程文件合并以生成合并的编程文件。在一些实施方式中,所述过程进一步包括使用位掩码来验证用于第二部分的编程位在经合并的编程文件中尚未被变更。

[0010] 在另一方面,公开了一种方法,其包括生成要被实现在可编程逻辑器件(PLD)上的电路设计的硬件描述语言(HDL)实施方式。所述方法包括将电路设计划分成第一部分和第二部分,第一部分包括用于耦接第一部分和第二部分的接口。所述方法另外地包括生成第二部分的网表表示。所述方法进一步包括验证第二部分。在验证第二部分之后,所述方法包括对第一部分进行修订。所述方法然后包括生成修订后的第一部分的网表表示。所述方法另外地包括将经先前验证的第二部分的网表表示与修订后的第一部分的网表表示合并。

[0011] 在一些实施方式中,所述第二部分是包括要求认证为满足功能安全要求的功能的安全关键部分。在一些实施方式中,所述方法进一步包括分析 HDL 实施方式以生成电路设计的层次实施方式,其中将电路设计划分成第一部分和第二部分在层次表示级被执行。在一些实施方式中,所述方法进一步包括生成包括用于修订后的第一部分和第二部分两者的编程设定的编程文件。在一些这样的实施方式中,所述方法进一步包括生成位掩码,所述位掩码包括其中每一个位都具有第一逻辑值的第一组位和其中每一个位都具有第二逻辑值的第二组位,所述第一组位中的每一个位都处于所述位掩码的如下位置中,即,该位置与所述编程文件中的被所述第二部分使用的编程位相对应。在一些这样的实施方式中,所述方法另外地包括使用位掩码来验证用于第二部分的编程位在所述编程文件中尚未被变更。

[0012] 参考附图在下面进一步对这些和其它方面进行描述。

#### 附图说明

[0013] 图 1 示出了 FPGA 架构的示例实施方式。

[0014] 图 2 示出了逻辑单元的示例实施方式。

[0015] 图 3 示出了示例 4 输入 LUT。

[0016] 图 4 示出了图示用于设计 IP 块的示例过程的流程图。

[0017] 图 5 示出了图示用于为 IP 块的秘密部分生成配置设定的示例过程的流程图。

[0018] 图 6A 示出了示例位掩码。

[0019] 图 6B 示出了由图 6A 的位掩码到示例 IP 开发者的编程文件的逐位应用产生的位

流。

[0020] 图 7 示出了图示用于将 IP 块集成到 IP 用户的设计中的示例过程的流程图。

[0021] 图 8A 示出了其中将被秘密部分使用的位标识有默认值的 IP 用户的编程文件的示例部分。

[0022] 图 8B 示出了由图 6B 的结果与图 8A 的 IP 用户的编程文件的部分的逐位应用产生的合并的编程文件(或其一部分)。

[0023] 图 9 示出了用于将用于 IP 块的秘密部分的配置设定合并到 IP 用户的配置设定中的示例过程的流程图。

[0024] 图 10 示出了用于将用于 IP 块的秘密部分的配置设定合并到 IP 用户的配置设定中的示例过程的流程图。

[0025] 图 11A 示出了 IP 开发者的编程文件的一部分的示例位阵列。

[0026] 图 11B 示出了在用于秘密部分的编程位的易位之后图 11A 的阵列。

[0027] 图 12 示出了图示用于生成包括安全关键部分以及非安全关键部分两者的电路设计(诸如用于 FPGA 的 IP 块)的示例过程的流程图。

[0028] 图 13 示出了图示用于将修订后的非安全关键部分与先前验证的安全关键部分合并的示例过程的流程图。

[0029] 图 14 示出了图示用于将修订后的非安全关键部分与先前验证的安全关键部分合并的另一示例过程的流程图。

[0030] 图 15 示出了一般地示出能够被用来实现上面所描述的 CAD 和其它工具的功能性中的一些或全部的计算机系统的图解框图表示。

### 具体实施方式

[0031] 在以下描述中,许多特定细节被阐述以便提供对所呈现的实施方式的彻底理解。可以在没有这些特定细节中的一些或全部的情况下实践所公开的实施方式。在其它实例中,尚未对众所周知的过程操作进行详细的描述以便不必要地使所公开的实施方式混淆。虽然将与特定实施方式相结合地描述所公开的实施方式,但是将理解的是,它不旨在限制所公开的实施方式。

[0032] 本公开一般地涉及诸如现场可编程门阵列(FPGA)的可编程逻辑器件(PLD),并且更具体地,涉及用于划分、传送以及集成 FPGA 电路设计的方法和系统。例如,一些实施方式涉及用于划分、传送一个或多个知识产权(IP)块以及将其与另一用户的设计的一个或多个其它 IP 块相集成的方法和系统。特定实施方式使得 IP 块开发者能够将 IP 块发送或者以其它方式提供给另一 IP 设计者、IP 用户或客户(在下文中共同地称为“用户”),而不用公开 IP 块的秘密部分的功能描述以及附加地确保秘密部分被以相同的方式实现在 PLD 上。在一些实施方式中,术语“用户”可以指的是正将一个或多个 IP 块或电路设计集成到要被配置或编程到物理 FPGA 器件中的最后的电路设计中的一方。在一些实施方式中,IP 开发者设计 IP 块以包括秘密部分和公共部分两者。在一些实施方式中,IP 开发者在寄存器传送级(RTL)级将公共部分作为硬件描述语言(HDL)实现的设计或者作为可合成网表提供给 IP 用户。在一些实施方式中,用户然后能够接收例如 IP 块的公共部分的 HDL 实现的表示,并且将公共部分与用户自己的设计相集成。用户然后能够基于包含配置设定或“编程位”的



已集成 HDL 设计来生成编程文件以将用户的设计(包括 IP 块的公共部分)实现在物理 FPGA 器件中。

[0033] 在一些实施方式中,IP 开发者以编程位的形式将 IP 块的秘密部分提供给用户,而不用提供秘密部分的 HDL、RTL 或网表实施方式。能够以可以为整个编程文件的一部分的位流的形式将用于秘密部分的编程位提供给 IP 用户。在其中用于秘密部分的编程位作为包括用于整个物理 FPGA 器件的位的整个编程文件被提供的一些实施方式中,用户使用由 IP 开发者所生成的并且同样提供给 IP 用户的位掩码来使来自 IP 开发者的编程文件的用于秘密部分的编程位与 IP 用户的编程文件合并。经合并的编程文件然后能够被加载到存储器中以用于编程到物理 FPGA 器件中。在一些其它实施方式中,诸如在对于部分可配置的 FPGA 器件有用的实施方式中,包含用于秘密部分的编程位的位流能够在 FPGA 设计的剩余部分已被编程到 FPGA 器件之后被编程到 FPGA 器件中。在一些这样的实施方式中,当包含用于秘密部分的编程位的位流被编程到 FPGA 器件中时已经编程到 FPGA 器件中的电路设计的剩余部分可以是运作的。

[0034] 附加地或替换地,在一些实施方式中,IP 块可以包括向认证机构要求功能安全认证的功能或设计部分以及不要求这样的认证的功能。如上所述,在一些情况下,在这样的设计被认证为与功能安全要求兼容之后,IP 设计者或用户可以寻求改变不要求功能安全认证的设计的一些功能或部分。然而,如上面同样描述的那样,常规技术要求 IP 块的所有部分被重新认证。然而,通过利用特定实施方式,IP 开发者能够设计 IP 块以包括安全关键部分(要求功能安全要求由认证机构认证的部分)以及非安全关键部分(不要求这样的认证的部分)两者。在一些实施方式中,开发者以 CRAM 编程位的形式至少将设计的安全关键部分提供给第三方认证机构(诸如莱茵 TÜV)。能够以位流的形式将用于安全关键部分的编程位提供给第三方认证机构以用于验证和认证,所述位流可以是整个编程文件的一部分。在安全关键部分与非安全关键部分的正确设计和划分的情况下,开发者或下游用户能够对设计的非安全关键部分做出改变同时确保用于安全关键部分的 CRAM 编程位保持不变—并且仍然有权通过认证机构认证满足认证机构的功能安全要求—而不管必须为非安全关键部分重新合成和重新生成编程位。

[0035] 随着数字电路的增加了的使用和复杂性,电路开发者已使用硬件描述语言在高级别上设计数字电路的数字逻辑描述和其它方面。一般地,硬件描述语言(HDL)是能够被用来描述电路的操作、其设计以及组织和测试以借助于模拟来验证其操作的计算机说明或建模语言。HDL 的示例包括 Verilog (IEEE1364)、VHDL (VHSIC 硬件描述语言)、Abel (高级布尔表达式语言)以及 Altera HDL (AHDL)。HDL 使得电路开发者能够在高级别上指定电路的功能性,而无需开发者指定特定电子技术和物理布局。典型地,开发者使用 HDL 来用定时信息指定电路的数据流模型。例如,开发者能够使用诸如电路或系统的空间和时间结构以及行为的基于 HDL 文本的表达式来指定他的设计。用 HDL 所实现的这种抽象级能够被称为寄存器传送级(RTL)。RTL 设计抽象根据例如硬件寄存器之间的数字信号(例如,数据)以及由组合逻辑对这些信号所执行的逻辑操作将电路设计建模为同步数字电路。

[0036] HDL 实现的设计能够使用计算机辅助设计(CAD)工具来创建。例如,还被称为电子 CAD (ECAD)的电子设计自动化(EDA)指的是用于设计包括诸如专用集成电路(ASIC)门阵列的集成电路以及诸如现场可编程门阵列(FPGA)的可编程逻辑器件(PLD)的电子系统

的软件工具(例如, CAD 工具)。使用这样的配置有 HDL 的 CAD 工具, 开发者能够写或者以其它方式指定电路设计。如上所述, 开发者能够使用 HDL 代码来指定设计中的一些或全部。开发者还能够使用高级合成语言来生成 HDL 实现的设计的一部分。例如, 这样的高级合成抽象语言包括来自 National Instruments 公司的 LabVIEW FPGA、来自 Annapolis Micro Systems, Inc. 的 CoreFire 设计套件、SystemVerilog、Handel-C 以及 SystemC 等等。开发者或用户能够导入已经在 HDL 指定的设计中的一些或全部, 并且使用 CAD 工具基于或者以其它方式修改所导入的设计。

[0037] 当开发者用 HDL 完成设计时, CAD 工具然后执行分析(例如, 代码评审或审查)为合成作准备。在分析期间, HDL 实现的设计经受各种错误检查操作以帮助在 HDL 实现的设计被合成之前解析错误。在一些实施方式中, CAD 工具能够基于分析来生成包括模块和模块之间的调用的设计的层次表示。使用 HDL 设计, 能够被结合到用来设计 HDL 的 CAD 工具中或者与其相集成的逻辑合成工具(并且在下文中还被一般地称为 CAD 工具) 然后根据例如逻辑电路和逻辑门将 HDL 实现的设计合成为设计实施方式。在一些实例中, 逻辑合成的结果是物理上可实现的网表。网表一般地是作为逻辑门或过程特定的标准单元所实现的电路设计的功能的布尔代数表示。和 HDL 对比, 网表语言一般地仅表达块或逻辑元件的层次之间的电路连通性。也就是说, 虽然 HDL 能够被用来表达包括时间特性的结构、行为或 RTL 架构, 但是网表语言不指定时间特性。网表然后能够被输出, 并且在一些实例中, 以例如“模拟”或“切换”网表的形式、或作为通用工业标准 EDIF (电子设计互换格式) 提供给另一方(例如, IP 用户)。

[0038] 用来设计 PLD 的 CAD 工具能够生成包括位流的二进制编程文件, 所述位流包含用于配置 PLD 的编程位。FPGA 是设计成在物理 FPGA 器件被制造之后被配置用于供用户使用的这种类型的 PLD。FPGA 包含可编程逻辑块和允许可编程逻辑块被以不同配置互连的可配置互连。图 1 示出了示例 FPGA 架构 100 (还被在本文中称为 FPGA 器件 100) 的代表性部分的示例实施方式。FPGA 架构 100 包括可配置逻辑块 (CLB) 102 (还被称为逻辑阵列块 (LAB)) 以及可配置输入 / 输出 (I/O) 垫 104、开关 106 以及布线架构或“结构 (fabric)” 108 的阵列。例如, 布线架构 108 能够包括连接逻辑块 102 和 I/O 垫 104 的多条垂直地定向的线路或通道 110 和多条水平的线路或通道 112。每个逻辑块 102 一般地包括一个或多个输入 114 和一个或多个输出 116。各种 FPGA 逻辑块 102 能够被配置成表现为简单的逻辑门(例如, 与门、与非门、或门、或非门以及异或门等等), 表现为逻辑电路(例如, 复用器和算术逻辑单元 (ALU) 等等), 表现为复杂组合功能, 表现为存储器元件(例如, 锁存器、触发器以及寄存器, 或甚至存储器的整个块)、存储器控制器, 以及表现为完整微处理器或处理器核。尽管在图 1 中出于说明的目的示出了六个逻辑块 102, 但是一般地, FPGA 100 或其它 PLD 能够包括布置在例如由布线架构 108 所互连的二维阵列中的几十个、几百个或更多逻辑块 102。本公开中所描述的特定实施方式不限于任何特定类型的 FPGA 或 FPGA 硬件布局(例如, 逻辑块、布线架构以及 I/O 垫的布局)。一些实施方式也可以适合于其它类型的 PLD。

[0039] 典型的 FPGA 逻辑块 102 包括一个或多个还被称为逻辑元件的逻辑单元 218。图 2 示出了逻辑单元 218 的示例实施方式。在一些实施方式中, 每个逻辑单元 218 都包括查找表 (LUT) 220。例如, 在一些 FPGA 器件中, 每个逻辑单元 218 包括 4 输入 LUT 220。在一些其它实施方式中, 每个 LUT 220 具有 7 个输入或另一数目的输入。在一些实施方式中, 每个

LUT220 被实际上实现为两个或更多个的更少输入 LUT。例如,4 输入 LUT 能够被实现为两个互连的 3 输入 LUT。

[0040] 许多 FPGA 是基于静态随机存取存储器(SRAM)。典型地,  $2^n$  个 SRAM 单元和对应的位被用来实现  $n$  输入 LUT—将“ $n$ ”个输入或输入功能映射到单个输出的 LUT。因此,为了实现 4 输入 LUT,典型地能够需要 16 个 SRAM 编程位。SRAM 单元常常被称为配置 RAM 或“CRAM”。编程到 CRAM 中(例如,在加电时或在部分再配置期间)以实现诸如 IP 块的电路设计的编程位还被称为“配置设定”。在一些实施方式中,每个 LUT220 都能够被实现为  $2^n:1$  复用器(或“MUX”),其中每个 CRAM 将输入提供给复用器的对应输入。在一些这样的实施方式中,  $2^n:1$  复用器能够被实现为  $2^n-1$  个单独的 2:1 复用器。图 3 示出了示例 4 输入 LUT220。4 输入 LUT 包括每个都将输入提供给第一级的复用器 324a 的 2:1 复用器的 16 个 CRAM322。第一级的复用器 324a 的输出是到第二级的复用器 324b 的输入。第二级的复用器 324b 的输出是到第三级的复用器 324c 的输入。最后,第三级的复用器 324c 的输出是到第四级的复用器 324d 的输入,所述第四级的复用器 324d 具有单个输出 E。给定级的复用器中的每一个复用器都由到 LUT220 的输入中的对应一个来控制。例如,第一级复用器 324a 被第一输入 A 控制,第二级复用器 324b 被第二输入 B 控制,第三级复用器 324c 被第三输入 C 控制,以及第四级复用器 324d 被第四输入 D 控制。基于输入信号 A、B、C 以及 D,复用器组选择位来驱动输出 E。CRAM322 中的位规定哪一个逻辑操作或功能将被 LUT220 执行。每个 LUT220 能够一般地实现能够利用到 LUT 的输入的数目来指定的任何逻辑操作。因此,  $n$  输入 LUT 能够一般地实现能够利用  $n$  个或较少二进制输入指定的任何逻辑功能。

[0041] 其它 CRAM322 控制或者规定开关 106 和包括在逻辑块 102 之间的布线以及在各种其它输入与输出之间(包括在逻辑单元 208 之间)和在逻辑块 102 与 I/O 垫 104 之间的布线的布线架构 108 的配置。为了将一个或多个电路设计的功能性编程到 FPGA 器件 100 中,即,为了配置由电路设计所需的逻辑块 102、I/O 垫 104、开关 106 以及布线架构 108,用户的 CAD 工具典型地生成包括对于整个 FPGA 设计的 IP 块和其它电路设计中的全部所要求的编程位的编程文件。编程文件还能够包括不在所实现的 FPGA 设计中使用的在物理 FPGA 器件中的 CRAM 的默认值。编程文件还能够一般地包括一个或多个命令或指令。在一些实施方式或应用中, FPGA 器件 100 被连接到电路板。在一些过程流中,编程工具将编程文件传送到在与 FPGA100 相同的板上的非易失性存储器,诸如 Flash 存储器或 EEPROM。在加电时或者基于指令, FPGA100 进入编程或“配置”模式或状态(与关断状态或正常操作状态相反)。例如,在板加电时,在具有 FPGA100 的板上的一些其它逻辑(诸如控制器)能够使 FPGA100 进入配置模式。当在配置模式下时, FPGA100 预期其管脚中的一些(其编程管脚)将在 CRAM 编程操作中接收位。使 FPGA 进入配置模式的逻辑或板上的一些其它逻辑(诸如控制器或存储器控制器)然后从非易失性存储器中检索编程位并且将所检索到的编程位提供给 FPGA100。FPGA100 内的逻辑然后将位编程到适当的 CRAM322 中以配置形成互连逻辑块的结构逻辑块 102 (例如, LUT220)、I/O 垫 104、开关 106 以及布线架构 108 以及逻辑块内的逻辑单元 218 或 FPGA 设计的其它组件。

[0042] 如在图 2 的示例中所示,每个逻辑单元 218 还能够包括一个或多个其它元件,诸如例如进位逻辑(例如,全加法器(FA))226、复用器 228 以及寄存器(例如,诸如 D 触发器的触发器)230。寄存器 230 可以从 FPGA100 的全局时钟树(未示出)接收时钟信号。寄存器可以

接收 LUT220 的输出信号 E 并且基于时钟信号输出信号 E。在一些实施方式中, FPGA100 能够包括不同类型的逻辑单元 208。例如, 各种逻辑单元 208 能够包括具有各种数目的输入的 LUT220。其它逻辑单元 208 还能够包括诸如复用器、解复用器以及寄存器的其它硬连线组件。在一些实施方式中, FPGA100 还能够包括嵌入式处理器和一个或多个存储器块。如上所述, 许多 FPGA 是基于静态 RAM (SRAM) 的。基于 SRAM 的 FPGA 是易失性的以及因此 SRAM 单元一般地在每次加电时需要被配置。也就是说, 每当 FPGA 被加电, 一些逻辑从同样在板上的存储器(例如, FLASH 存储器)中检索编程位并且将编程位提供给 FPGA。FPGA 内的逻辑然后将位编程到适当的 CRAM322 中。其它类型的 FPGA 包括带集成 Flash 的基于 SRAM 的 FPGA、基于 Flash 的 FPGA 以及基于反熔丝的 FPGA。一些 FPGA 被配置为芯片上系统(SOC)。例如, 由 Altera 公司所生产的一些 SOC FPGA 集成嵌入式处理器。例如, 一些 Altera SOC FPGA 包括了集成的硬处理器系统(HPS)。这样的 HPS 能够包括使用高带宽互连骨干的处理器、外围设备以及存储器接口。

[0043] 在许多现代应用中, FPGA 电路开发者仅设计整个 FPGA 设计的一部分。例如, 一个 FPGA 电路开发者能够设计处理器核, 另一 FPGA 开发者能够设计存储器块, 以及然而其它人能够设计 FPGA 设计的其它部分。各种电路开发者可以为作为不同团队的一部分的单个实体或者为不同的实体(例如, 为不同的商业企业中的不同的电路开发者)而工作。至少部分地由于设计和验证电路设计的困难和费用以及设计的商用可能, 设计实体可能期望使电路设计保密, 或者至少期望不公开, 或者将电路设计的公开限制于诸如其它开发者、用户、设计者或客户的其它方。如本文中所述的半导体知识产品核或简单地“IP 核”或“IP 块”一般地指的是用于诸如 FPGA 的 PLD 的电路设计。一般地, IP 块将仅构成更大 FPGA 设计的一部分。在各种实施方式中, IP 块被设计成实现特定功能或功能性并且可以被设计成与其它用户的设计的另一 IP 块或诸块相集成以及因此与更大 FPGA 设计的一个或多个 IP 块或其它部分相结合地工作。每个 IP 块能够用一个或多个逻辑块 102 加以实现并且典型地包括多个逻辑单元 218。附加地, 在一些 FPGA 实施方式中, 每个逻辑块 102 能够实现一个或多个 IP 块中的一些或全部。IP 块是特定电路开发者的设计, 并且典型地是特定电路开发者的知识产权。开发者可以将 IP 块的使用出售或者授权给之后将 IP 块实现到用户自己的更大设计中的另一开发者或用户, 所述用户自己的更大设计它本身可以是更大设计的一部分。

[0044] 尽管 IP 块能够被作为网表发送到其它方, 诸如 FPGA 的用户, 但是 IP 块典型地在寄存器传送级(RTL)级被发送, 诸如以可合成硬件描述语言(HDL)实现的设计的形式。因为设计被以 HDL 形式传送, 所以用户能够在功能级修改设计。然而, 如上所述, 因为开发者可能因为验证所牵涉的时间和费用或由于其它原因而希望使设计保密(例如, 作为商业秘密), 所以开发者可能不想用户能够具有对电路设计的功能描述的这样的访问。例如, 因为如果设计被修改了则电路开发者无法提供对 IP 块的授权或支持, 所以电路开发者可能期望防止 FPGA 用户能够访问电路设计的功能描述。将电路设计作为软件网表提供给用户能够比以 HDL 形式提供设计提供对逆向工程的更好保护, 但仍然能够实现逆向工程。IP 开发者还可以出于开发者自己的好意而简单地期望确保终端用户或客户收到已验证实施方式的利益或者期望满足功能安全要求。

[0045] 一些 IP 开发者被要求让他们的 IP 块被诸如认证机构(诸如总部设在德国科隆的莱茵 TÜV)的第三方独立地验证以确保集成该设计的产品将满足功能安全要求。例如, 认

证机构可以认证 IP 块满足电气 / 电子 / 可编程电子安全相关系统规范—IEC61508:2010—行业中应用的规则的国际化标准的功能安全。这可能是显著地代价高的过程。IP 开发者可以寻求防止第二 IP 用户修改 IP 块以便与功能安全要求兼容的验证不被要求两次。另外地, IP 开发者和 IP 用户中的任一个或两者都可以寻求确保验证的 IP 块设计随着它被合成和验证以及在功能安全市场中随着它被认证机构认证而被提供给 IP 用户。因此 IP 用户的 CAD 工具可以与 IP 开发者的合成工具或第三方验证者的工具不同地合成、布局 HDL 实现的设计或者给其定路线,所以不能够保证 IP 用户生成如 IP 开发者生成的在 CRAM 级上的确切的编程位。由于至少这种原因, IP 用户可以期望由 IP 开发者生成的确切的编程位(在 CRAM 位级上),因为如由 IP 开发者生成的编程位所实现的设计已经被认证了。另外地, IP 块可以包括向认证机构要求功能安全认证的功能以及不要求这样的认证的功能。在一些情况下,在这样的设计被认证为与功能安全要求兼容之后,IP 开发者或用户可以寻求改变不要求功能安全认证的设计的一些功能或部分。然而,常规技术要求 IP 块的所有部分被重新认证。

[0046] 在一些实施方式中,设计过程使得 IP 开发者能够为 IP 块生成秘密部分和公共部分两者。图 4 示出了图示用于设计 IP 块的示例过程 400 的流程图。在一些实施方式中,过程 400 在其中 IP 开发者开发 IP 块的 HDL 表示的 402 处开始。在一些实施方式中,如上所述,IP 开发者使用 CAD 工具来设计或者建立 IP 块的 HDL 实现的表示。例如,CAD 工具能够包括具有图形用户接口(GUI)的输入级。CAD GUI 可以包括使得能实现诸如用于 IP 块设计的参数和规范的信息的高效或方便的键入的各种“向导”。例如,HDL 实现的 IP 块能够具有 RTL 架构的形式。如上所述,IP 开发者能够使用硬编码逻辑或者使用预建功能、宏功能或其它 IP 块来建立设计。IP 开发者还能够使用之后能够被 CAD 工具用来生成 HDL 描述的高级合成抽象语言。

[0047] 具有这样的 HDL 生成能力的 CAD 工具的一个示例是可购自加州圣何塞的 Altera 公司的可编程芯片上系统(SOPC)生成器(Builder)。在一些实施方式中,除提供生成 HDL 设计的功能性之外,CAD 工具还能够包括用于执行分析、合成、验证、布局以及布线和用于配置设定的生成的功能性。也就是说,在一些实施方式中,这些功能性中的一个或多个中的每一个都可以是单独的程序或单个 CAD 工具内集成程序的一部分。在一些其它实施方式中,功能性中的一个或多个可以用结合到单个 CAD 工具中或集成在单个 CAD 工具内的不同硬件(以及关联的固件和软件)加以实现。在一些其它实施方式中,功能性可以用两个或更多个不同的不同工具加以实现。在其中刚才描述的多个功能性中的每一个都被实现为单个 CAD 工具内的单独程序的一些实施方式中,程序之间的接口可以是数据库文件、日志或简单地在程序之间传送的消息。例如,代替将文件写入存储器,输入级能够将消息直接地发送到生成或者集成 IP 块的 HDL 表示的 HDL 生成器程序。类似地,HDL 生成器程序能够将 HDL 表示直接地提供给分析和合成工具。

[0048] 在 404 处,CAD 工具然后分析 HDL 实现的 IP 块。在一些实施方式中,CAD 工具在 406 处基于分析来生成 IP 块的项目层次(或层次表示)。在一些实施方式中,项目层次包括 IP 块的所有模块(包括其中每一个都能够实例化的模块)的层次树,以及模块之间的其它连接或调用。在 408 处,开发者然后能够使用 CAD 或其它工具通过大体上分隔或者遮蔽开发者期望包括在 IP 块的秘密部分中的项目层次的部分来划分或者以其它方式定义 IP 块的秘密部分。在这样做时,IP 开发者还定义 IP 块的公共部分—开发者将以 HDL、网表或其它功

能表示形式提供的 IP 块的部分(若有的话)。

[0049] 在一些实施方式中,CAD 工具然后在 410 处生成包括 IP 块的公共部分但不将模块包括在 IP 块的秘密部分内的修改的项目层次。在一些实施方式中,秘密部分被用 HDL 表示为使实现该秘密部分的模型实例化的调用,所述秘密部分它本身能够包括若干子模块。在一些过程流中,项目层次的公共部分然后能够被传送给 IP 用户。例如,能够作为经修改的项目层次或以 HDL 的形式来传送公共部分。然而,再次,无论以层次(例如,模块)形式、HDL 形式还是其它功能表示形式都不在功能表示级提供设计的秘密部分。因为 IP 开发者能够在这个阶段将 IP 块的公共部分的层次表示提供给 IP 用户,所以用户然后能够开始从事将 IP 开发者的 IP 块结合或者集成到 IP 用户的设计中。例如,通过区别哪些模块形成 IP 块的秘密部分。在一些实施方式中,IP 开发者在下面所描述的 IP 开发者的秘密部分的合成或验证之前将经修改的公共部分的项目层次或 HDL 实现的设计提供给 IP 用户。例如,IP 开发者可以提供设计的这个早期阶段以促进包括由 IP 开发者所设计的那些部分以及由接收到 IP 开发者的 IP 块的 IP 用户所设计的那些部分的整个组合的 FPGA 设计过程的总体速度和总体效率。在一些其它实施方式中,IP 开发者可以在合成和验证之后提供 HDL 实现的设计。因为公共部分能够包括整个 IP 块的输入和输出中的全部的列表或规范,所以用户能够将 IP 块集成在用户自己的设计内而不用具有 IP 块的秘密部分的知识。在一些实施方式中,提供给 IP 用户的 HDL 文件包括允许 IP 用户使用 CAD 工具在用户自己的设计中使 IP 块的公共部分的实例实例化的(例如,对模块或子模块的)命令或调用。例如,用户的 CAD 工具在使公共模块实例化之后,能够将 IP 块的输入和输出连接到用户的 IP 块中的一个或多个或者连接到其它开发者的 IP 块。公共部分还能够包括到秘密部分的接口(在下面描述),以及诸如例如时钟缓冲器、锁相环(PLL)以及存储器的共享资源等其它共享资源。

[0050] 图 5 示出了图示用于为 IP 块的秘密部分生成配置设定的示例过程 500 的流程图。例如,能够按请求或者在根据过程 400 设计了 IP 块的 IP 开发者的指导下执行过程 500。在一些实施方式中,过程 500 在过程 400 之后不久开始。在一些其它实施方式中,过程 500 的一个或多个步骤或块与过程 400 的各种步骤或块并行地执行或者与过程 400 的各种步骤或块相互混合。在一些实施方式中,过程 500 在其中 CAD 工具基于在 410 处生成的经修改的项目层次、或者基于关于秘密部分与公共部分之间的交互的其它功能或模块级信息在 HDL 实现的设计中生成 IP 块的公共部分与 IP 块的秘密部分之间的接口(上面参考图 4 的过程 400 来描述)。在一些这样的实施方式中,能够用边界 LUT 实现接口。例如,IP 开发者能够设计 IP 块,或者 CAD 工具能够生成 IP 块,使得它在 IP 块的秘密部分与 IP 块的公共部分之间的边界处包括 LUT。边界 LUT 将从公共部分输入到边界 LUT 的信号和资源映射为秘密部分内的资源,并且将从秘密部分输出的信号和资源映射为公共部分内的资源。以这种方式,虽然 CAD 工具一般地需要具有它需要给到特定边界 LUT 的特定信号定路线的知识,但是 CAD 工具不需要在秘密部分内在输入 LUT 下游或在输出 LUT 上游执行的映射的知识。在一些这样的实施方式中,边界 LUT 能够被逻辑上定义为单个输入缓冲器,其输出被定路线到物理 FPGA 设计中的特定逻辑单元 218。在一些这样的实施方式中,边界 LUT 本身使用 FPGA 器件 100 的 LUT220 或其它物理 LUT 而被用 CRAM 编程位物理上实例化。在一些其它实施方式中,秘密部分与公共部分之间的接口的一个或多个部分能够实现有如用 CRAM 编程位物理上实例化的触发器、寄存器、线或其它逻辑或电路组件的功能性。

[0051] 在一些其它实施方式中,IP 开发者不将 IP 块的任何公共部分提供给 IP 用户。例如,IP 开发者可能期望使整个设计保密。在一些这样的实施方式中,在 502 处生成的边界接口是提供给 IP 用户的 IP 块的唯一网表、HDL 或其它功能级表示。然而不是作为秘密部分与公共部分之间的接口,而是该接口是秘密部分与外面(例如,集成秘密部分的 IP 用户的 FPGA 设计)之间的接口。

[0052] 在一些实施方式中,过程 500 在其中 IP 开发者的 CAD 工具至少对 IP 块的秘密部分的 HDL 实现的表示执行合成操作的 504 处继续。合成工具的一些示例是可购自俄勒冈州威尔逊维尔的 Mentor Graphics 公司的 Leonardo Spectrum 和可购自加利福尼亚州森尼韦尔的 Synplicity 公司的 Synplify。在一些实施方式中,CAD 工具然后在 506 处对已合成设计执行验证操作。一些示例验证工具包括可购自加利福尼亚州森尼韦尔的 Synopsys 公司的 Synopsys VCS、VSS 和 Scirocco 以及可购自加利福尼亚州圣何塞的 Cadence Design Systems 的 Cadence NC-Verilog 和 NC-VHDL。在一些实施方式中,在 506 处的验证之后,已合成设计能够被提供给 CAD 工具内的物理工具或程序,其中在 508 处布局和布线操作被执行。典型的布局和布线工具将逻辑单元定位于目标硬件器件的特定逻辑元件上并且依照实现设计所要求的逻辑和安全性来连接各种逻辑元件的输入与输出之间的线。

[0053] 在一些实施方式中,CAD 工具在 510 处基于布局和布线操作的输出来生成排他列表。该排他列表包括将被保留仅用于供用户最后的 FPGA 设计中的 IP 块的秘密部分使用的资源中的全部。因为接收到 IP 开发者的 IP 块的 IP 用户将没有 IP 块的秘密部分如何正在使用实际 FPGA 硬件的知识,所以 IP 用户确保 IP 用户的 CAD 工具在将 IP 开发者的 IP 块与 IP 用户的设计相集成时,不使用 IP 开发者已通过避免使用在排他列表中所指定的资源而为秘密部分保留的资源。在排他列表中所指定的资源能够包括例如将实现各种逻辑块 102、逻辑单元 218、LUT220、I/O 垫 104、开关 106 或将仅被 IP 块的秘密部分使用的布线架构 108 的其它部分的物理资源。

[0054] 可编程逻辑配置级能够采用布局和布线工具的输出,并且在 512 处为 FPGA 器件 100 生成配置设定。例如,能够在可购自加利福尼亚州圣何塞的 Alterra 公司的 Quartus 开发工具中提供布局和布线工具以及逻辑配置级。如由本领域的普通技术人员将了解的那样,能够在各种实施方式中使用各种合成、布局和布线以及可编程逻辑配置工具。在一些其它实施方式中,CAD 工具能够被配置成直接地根据 HDL 实现的设计来为 FPGA 生成配置设定。在一些实施方式中,在 512 处配置级的输出是包括形式为用于秘密部分的编程位的配置设定的编程文件。在一些实施方式中,虽然编程文件对于给定物理 FPGA 硬件架构一般地包括用于 CRAM322 中的全部的所有位,但是对于 IP 开发者的 IP 块未被生成的编程文件中的所有位分配了默认值。在一些实施方式中,针对 IP 块的公共部分所指定的位同样分配了默认值。在一些实施方式中,然后这时能够使用编程位再次验证 IP 块的定时、功率或其它要求。例如,用于秘密部分、公共部分的编程位以及秘密部分与公共部分之间的接口(其可以作为公共部分的一部分被包括)然后能够被编程到之后经受验证的物理 FPGA 器件中。验证的结果然后能够被提供给诸如认证机构(诸如莱茵 TÜV)的第三方。附加地或替换地,在一些实施方式中,第三方(诸如莱茵 TÜV)能够执行验证。

[0055] 在一些实施方式中,过程 500 在其中生成位掩码的 514 处继续。图 6A 示出了用于教导目的的示例位掩码 632。位掩码一般地标识被秘密部分使用的 IP 开发者的编程文件中

的位。尽管所图示的示例位掩码 632 仅是 64 位值的 8x8 阵列,但是应该了解的是,一些实施方式所要求的位掩码可以是更大的,例如,具有几百个、几千个或更多个位值。因为 IP 块的秘密部分所要求的编程位能够构成整个编程文件的仅一部分,所以位掩码被用来标识将被用来配置秘密部分的编程文件中的那些位。在其中 IP 块的秘密部分将被配置到仅单个逻辑块 102 或其它部分中的一些实施方式中,位掩码能够包括仅仅与该逻辑块或其它部分相关联的位。在一些其它实施方式中,位掩码能够包括用于整个编程文件的位。在所图示的示例中,具有逻辑值“1”的位值的所有位 634 指示编程文件中的对应位将在配置 IP 块的秘密部分时被使用。相比之下,具有逻辑值“0”的位值的所有位 636 指示编程文件中的对应位将不在配置 IP 块的秘密部分时被使用。

[0056] 在一些实施方式中,在 516 处,编程文件(包括用于 IP 块的秘密部分的配置设定/编程位)、位掩码、资源的排除列表、HDL 或网表实现的接口(或秘密和公共部分之间的接口的其它物理级表示)以及 HDL 实现的公共部分(或公共部分的公网表或其它功能级表示)被提供给 IP 用户。虽然被描述为单个步骤 516,但是在一些其它实施方式中,能够独立地或分别地做出步骤 516 的各种供给。例如,排除列表能够在为秘密部分提供编程位之前被提供给 IP 用户(例如,以总体上促进整个 FPGA 设计过程的速度和效率)。以无论什么形式提供给 IP 用户(例如,作为 HDL 或公网表的一部分)的接口(其可以是公共部分的一部分)向 IP 用户公开边界 LUT 或其它接口组件的位置。这允许 IP 用户的布局和布线操作将 IP 用户的设计与 IP 开发者的 IP 块连接。因此,通过向 IP 用户公开接口、公共部分的公网表或 HDL 实现的设计以及排除列表,IP 用户然后具有 IP 用户需要将 IP 块与 IP 用户自己的设计相集成的所有信息(例如,IP 块边界、输入以及输出和资源),而无需秘密部分的功能性的知识。

[0057] 在一些其它实施方式中,不是提供如刚才描述的排除列表,而是 IP 开发者替代地使用布图规划技术来选择并且向 IP 用户指示将为 IP 开发者的 IP 块以及具体地为 IP 块的秘密部分而保留的 FPGA 器件的部分或区。在一些这样的实施方式中,IP 开发者仅公开可以被 IP 块的秘密部分使用的资源的最大集,而不是通过 IP 开发者的布局和布线操作实际上生成的特定资源。布图规划的方法在其中多个 IP 开发者为一个用户提供 IP 块的情况下是特别有利的。如果平面布置图提前取得一致意见,并且每个 IP 开发者的布局和布线被约束成使用仅分配给 IP 开发者的布图规划区中的资源,则 IP 用户被确保当各种 IP 块被集成到一个设计中时没有资源将有冲突。

[0058] 在一些实施方式中,包括包含编程位的位流的编程文件中的一些或全部能够在传送给 IP 用户之前被加密。在一些实施方式中,至少用于秘密部分的编程位被加密。诸如 FPGA 器件 100 的 FPGA 器件还能够包括用于对加密的位流进行解密的解密逻辑。在这样的实施方式中,一般地是 FPGA 硬件的制造商或开发者确定加密方案并且然后将方法、密钥或其它加密技术提供给开发者或用户以对他们的编程位进行加密。在一些这样的实施方式中,FPGA 器件能够被配置成随着它们被提供给 FPGA 器件 100 并且在加载到 CRAM322 中之前及时对加密的编程位流进行解密。图 7 示出了图示用于将 IP 块集成到 IP 用户的设计中的示例过程的流程图。在一些实施方式中,过程 700 在其中 IP 用户生成 FPGA 设计或者将其导入到用户的 CAD 工具中的 702 处开始。用户的 FPGA 设计将一般地包括一个或多个 IP 块并且将与由另一 IP 开发者所生成的至少一个 IP 块接口对接。在一些实施方式中,IP



用户在 704 处接收 IP 开发者的 IP 块的公网表或 HDL 实现的公共部分, 诸如分别参考图 4 和 5 的过程 400 和 500 所描述的。在一些实施方式中, 公网表还包括 IP 用户的 CAD 工具所必须将芯片上的输入和输出信号线连接到的秘密 IP 块的接口输入和输出的位置。在一些实施方式中, 由 IP 开发者所提供的唯一公网表是秘密 IP 的边界输入和输出(例如边界 LUT) 的实施方式。IP 用户在 706 处还接收由 IP 开发者的 CAD 工具所生成的排除列表, 其指定 FPGA 设计中的哪些资源将被保留用于供 IP 块的秘密部分使用。在 708 处, IP 用户将公网表或 HDL 实现的公共部分和排除列表导入或者以其它方式提供到用户的 CAD 工具中。在 710 处, CAD 工具例如在 RTL、HDL 或网表级将公共部分与用户的设计的剩余部分相集成。

[0059] 在一些实施方式中, 过程 700 在其中 IP 用户的 CAD 工具对已集成设计执行分析和合成操作的 712 处继续。在一些实施方式中, 用户的 CAD 工具然后在 714 处对已合成设计执行验证操作。如上所述, 在一些实施方式中, 在 714 处的验证之后, 经验证的设计能够被提供给 CAD 工具内的物理工具或程序, 其中在 716 处布局和布线操作被执行。在布局和布线操作期间, IP 用户的 CAD 工具能够使用排除列表来保证它不使用由秘密 IP 所保留的资源, 并且秘密和公共部分(由 IP 开发者提供)之间的已合成接口用来给如由 IP 开发者所指示的 IP 块的秘密和公共部分之间的输入和输出定路线, 从而将由 IP 开发者所提供的 IP 块与 IP 用户的电路设计链接起来。可编程逻辑配置级然后能够采用布局和布线工具的输出, 并且在 718 处为要被编程到物理 FPGA 器件中的 IP 用户的设计生成配置设定。

[0060] 在一些实施方式中, 在 718 处配置级的输出是包括形式为用于包括 IP 开发者的 IP 块的公共部分的整个 FPGA 器件的编程位的配置设定的编程文件。如上所述(在 IP 开发者的编程文件的背景下), 虽然 IP 用户的编程文件对于给定物理 FPGA 硬件架构一般地包括 CRAM322 中的全部的位, 但是针对 IP 用户的设计未被生成的编程文件中的所有位(诸如由排除列表为 IP 开发者的设计的秘密部分而保留的那些)能够分配默认值。图 8A 示出了 IP 用户的编程文件的示例部分 840, 其中将被秘密部分使用的位 834 标识有默认值(图示为“d”)同时未被秘密部分使用的位 836 包括在 718 处针对它们所生成的值。因为 IP 用户的 CAD 工具提供有排除列表, 所以这确保在 716 处执行的布局和布线操作不使用在排除列表中所指定的任何资源, 以及因此, 这确保位 834 将被留在默认值。IP 用户的 CAD 工具能够通过将所生成的 CRAM840 与所提供的位掩码相比较来验证工具正确地遵守了排除列表—确保在位掩码 634 中包括有‘1’的位不分配除默认以外的值, 如在 834 中那样。在一些实施方式中, 在 720 处, IP 用户接收(如果他早已尚未这样做) IP 开发者的编程文件, 诸如参考图 5 的过程 500 上面所描述的编程文件。在 722 处, CAD 或其它工具然后使 IP 开发者的编程文件与 IP 用户的编程文件合并; 即, 为了将秘密部分的配置设定/编程位与由 IP 用户所生成的配置设定/编程位(包括用来实现 IP 开发者的 IP 块的公共部分和接口的位)合并。

[0061] 图 9 示出了用于将用于 IP 块的秘密部分的配置设定合并到 IP 用户的配置设定中的示例过程 900 的流程图。在一些实施方式中, 过程 900 是过程 700 的步骤 722 的示例实施方式。在一些实施方式中, 过程 900 在其中用户的 CAD 工具用同样由 IP 开发者所提供的位掩码对 IP 开发者的(先前接收到的)编程文件进行操作的 902 处开始。在一些实施方式中, 在 902 处执行逐位比较或逐位加法。例如, 图 6B 示出了由图 6A 的位掩码到示例 IP 开发者的编程文件的逐位应用产生的位流(示出为二维阵列)638。如图所示, 在位置上对应于

图 6A 的位掩码中的位 634 的图 6B 的阵列中的任何位 635 使值保持在 IP 开发者的编程文件中。在位置上对应于图 6A 的位掩码中的位 636 的图 6B 的阵列中的任何位 637 被设置为例如默认值。过程 900 然后能够在其中 IP 用户的 CAD 或其它工具执行在 902 处所获得的结果(如图 6B 中所示(为了教导目的))与 IP 用户的编程文件 840 (如图 8A 中所示(为了教导目的))之间的逐位比较的 904 中继续。例如,对于对应于秘密编程位中的一个的图 6B 中的每个位 635,该位 635 的值然后被添加或者以其它方式进入到 IP 用户的编程文件 840 中代替所对应的位位置 834 中的默认值。图 8B 示出了由图 6B 的结果 638 与图 8A 的 IP 用户的编程文件 840 的部分的逐位应用产生的合并的编程文件(或其一部分) 842。如在经合并的编程文件 842 中所示,位 835 现在具有从前在 IP 用户的原始编程文件 840 的位 834 中保持默认值的秘密编程位 635 的值。在一些实施方式中,在 906 处,经合并的编程文件然后被存储在例如与物理 FPGA 器件相同的板上的外部非易失性存储器(例如,Flash)中。如上所述,在给器件加电时,板上的逻辑被配置成使 FPGA 进入编程模式或配置模式,从非易失性存储器中检索编程位,以及将所检索到的编程位加载到 FPGA 器件中。FPGA 器件内的逻辑然后将编程位加载到适当的 CRAM 中。通过正像 IP 开发者已生成了它们那样加载用于秘密部分的编程位的值,IP 开发者能够被确保秘密部分的功能性像它被 IP 开发者的 CAD 工具验证的那样,同时不将秘密部分的实施方式公开给 IP 用户。

[0062] 图 10 示出了用于将用于 IP 块的秘密部分的配置设定合并到 IP 用户的配置设定中的示例过程 1000 的流程图。在一些实施方式中,过程 1000 是过程 700 的步骤 722 的示例实施方式。在一些实施方式中,过程 1000 在其中将 IP 用户的编程文件的编程位(例如,如在图 7 的过程 700 的 718 处获得的)加载到物理 FPGA 器件中的 1002 处开始。在这样的实施方式中,FPGA 器件有利地是部分可配置的 FPGA 器件。如上所述,典型地,配置或者再配置 FPGA 要求当外部控制器或其它逻辑将整个 FPGA 设计加载到 FPGA 器件中时 FPGA 处于配置模式。相比之下,部分再配置使得 FPGA 的特定(例如,关键的)部分在控制器将部分设计加载到部分可配置的 FPGA 器件上的可配置模块中时继续操作。一段时间以后,在 1004 处,IP 用户能够将仅包含为秘密部分所生成的那些编程位的位流加载到 FPGA 中。例如,不是在过程 700 的步骤 720 处接收整个 IP 开发者的编程文件,而是 IP 开发者能够选择仅发送包含用于秘密部分的编程位的位流。在其它实施方式中,IP 开发者能够发送如上所述的整个编程文件和位掩码以使得 IP 用户能够提取仅包含用于秘密部分的那些编程位的位流。附加地,通过发送位掩码,IP 用户还能够确保 IP 用户的编程文件不使用为秘密部分而保留的位中的任一个。如上所述,包含用于秘密部分的编程位的位流能够首先被存储在具有 FPGA 的板上的非易失性存储器中。具有 FPGA 的板上的控制器或其它逻辑随后能够检索位流并且将编程位加载到部分可配置的 FPGA 器件中进入到适当的 CRAM。在一些实施方式中,包含用于秘密部分的编程位的位流被以加密的形式从 IP 开发者接收,以便 IP 用户没有被用来实现 IP 开发者的 IP 块的秘密部分的位值的知识;相反地,IP 用户仅具有被以位掩码给出的位位置的知识。如上所述,一些 FPGA 器件还能够包括用于对加密的位流进行解密的解密逻辑。在这样的实施方式中,一般地是 FPGA 硬件的制造商或设计者确定加密方案并且然后将方法、密钥或其它加密技术提供给开发者或用户以对他们的编程位进行加密。在一些这样的实施方式中,FPGA 器件能够被配置成随着它们被提供给 FPGA 器件并且在加载到 CRAM 之前对经加密的编程位流进行解密。

[0063] FPGA 典型地具有将在 IP 块之间被共享的许多资源,所述 IP 块诸如例如全局布线网络或“树”以及其它可编程功率设定。在一些实施方式中,IP 开发者向 IP 用户指定全局布线树的拓扑。在一些这样的实施方式中,没有 IP 开发者的 IP 块的功能性在这个时候被公开;相反地,仅关于什么时钟信号或其它全局信号被使用以及其中他们使用了 FPGA 的哪些区的信息被公开。能够甚至在 IP 开发者将 IP 块的公共部分提供给 IP 用户之前将这种信息提供给 IP 用户。例如,在 716 处执行的布局和布线操作中,IP 用户然后具有将附加的扇出添加到这些信号以得到连接到时钟网络的 IP 用户的设计的部分。例如,在上面所描述的过程 900 或 1000 中的任一个中,当在步骤 514 处 IP 开发者生成位掩码时,工具能够附加地生成包含用于由 IP 开发者针对时钟网络所选择的线或布线资源中的全部的位的网络掩码、以及用于那些位的值 / 设定。在网络掩码中使用该组位,IP 用户能够验证 IP 用户在 718 处生成的编程文件对于在该网络掩码中所指定的位中的全部都具有相同的值。也就是说,要被实现在物理 FPGA 中的时钟拓扑保持相同。功率设定能够由 IP 用户以类似的方式预选择和传送。例如,IP 开发者能够以具有做出那些设定的位的掩码以及具有用于那些位的值的编程文件的形式将开发者已选择的功率设定的列表提供给用户。如上所述,IP 用户然后能够将那些功率设定集成到用户的设计中,并且以 IP 开发者实现它们的相同方式来验证设计的输出实现功率设定。

[0064] 附加地,在其中设计将被配置到其中的物理 FPGA 架构被一致地布置在芯片平面布置图中的实施方式中,能够有可能在芯片上任何地方使 IP 块实例化。以这种方式,IP 用户能够确定在 FPGA 中哪里实现 IP 开发者的 IP 块。例如,IP 用户的 CAD 工具能够如先前一样取得从 IP 开发者接收到的秘密部分的编程位。为了将 IP 块实现到不同的布图规划区或不同的逻辑块中,IP 用户的 CAD 工具能够改变 IP 块的物理实施方式的公共部分(如果被提供)的位置,并且还在 IP 开发者的编程文件中添加偏移从而向 CAD 工具指示将被保留用于秘密部分的新的资源集。以这种方式,IP 用户能够改变 IP 块的设计的位置。例如,图 11A 示出了 IP 开发者的编程文件的一部分的示例位阵列然而图 11B 示出了在用于秘密部分的编程位 1144 的改变位置之后图 11A 的阵列。为了利用图 11A 和 11B 中所图示的改变位置功能性,IP 用户还将一般地必须改变在 706 处接收的排除列表中所给出的资源的位置。

[0065] 如上所述,在一些实例中,整个电路设计的 IP 块或其它部分可以包括向认证机构要求功能安全认证的功能或设计部分以及不要求这样的认证的功能或设计部分。在一些情况下,在这样的设计被合成、验证以及认证为与功能安全要求兼容之后,开发者或下游用户可以寻求改变不要求功能安全认证的设计的一些功能或部分。然而,如上面同样描述的那样,常规技术要求设计的所有部分被重新认证。也就是说,因为每当工具被运行时(尤其是如果这些过程中的任一个在不同的工具上被执行)CAD 工具可以针对 HDL 实现设计不同地合成、布局、布线以及生成编程位,所以即使设计的安全关键部分未被改变,也不能够保证用于设计的安全关键部分的编程位在每一轮合成和生成之后是相同的。因此,按照惯例,尽管有意地仅对更大设计的非安全关键部分做出改变,但是因为不能够保证用于安全关键部分的编程位保持与由认证机构所认证的那些相同,所以认证将不依附重新合成的和重新生成的设计。为了获得已满足了功能要求的认证机构的认证,认证机构或下游用户将很可能要求重新认证。如上面所指出的那样,这样的认证可能是相当代价高的以及费时的。

[0066] 然而,通过利用特定实施方式,开发者能够划分设计以包括安全关键部分(要求功

能安全要求由认证机构认证的部分)以及非安全关键部分(不要求这样的认证的部分)两者。在一些实施方式中,开发者以 CRAM 编程位的形式至少将设计的安全关键部分提供给第三方认证机构(诸如莱茵 TÜV)。能够以位流的形式将用于安全关键部分的编程位提供给第三方认证机构以用于验证和认证,所述位流可以是整个编程文件的一部分。在安全关键部分与非安全关键部分的正确设计和划分的情况下,开发者或下游用户能够对设计的非安全关键部分做出改变同时确保用于安全关键部分的 CRAM 编程位保持不变—并且仍然有权通过认证机构认证满足认证机构的功能安全要求—而不管必须为非安全关键部分重新合成和重新生成编程位。认证机构或其它第三方可以要求安全关键部分保持不变并且有权认证的证据。为了确保这样的符合性,能够做出涉及安全关键部分的整个编程文件的位流部分的检查。例如,由安全关键部分的开发者所生成的位掩码能够被用来对最后的编程文件执行逐位比较以便确保与安全关键部分相对应的编程位保持不变并且有权认证。

[0067] 图 12 示出了图示用于生成包括安全关键部分以及非安全关键部分两者的电路设计(诸如用于 FPGA 的 IP 块)的示例过程 1200 的流程图。在许多点或方面上,过程 1200 的步骤与参考图 4 和 5 上面所描述的过程 400 和 500 的一个或多个步骤类似,其中安全关键部分类似于秘密部分并且非安全关键部分类似于公共部分。在一些实施方式中,过程 1200 在其中设计者开发电路设计的 HDL 表示的 1202 处开始。在一些实施方式中,如上所述,设计者使用 CAD 工具来开发或者建立设计的 HDL 实现的表示。在 1204 处,CAD 工具然后分析 HDL 实现的电路设计。在一些实施方式中,CAD 工具在 1206 处基于分析来生成电路设计的项目层次(或层次表示)。在一些实施方式中,项目层次包括电路设计的所有模块(包括其中每一个模块都能够实例化的模块)的层次树,以及模块之间的其它连接或调用。

[0068] 在 1208 处,开发者然后能够使用 CAD 或其它工具通过大体上分隔或者遮蔽开发者期望包括在电路设计的安全关键部分中的项目层次的部分来划分或者以其它方式定义电路设计的安全关键部分(开发者可能希望像上面所描述的那样保密)。在这样做时,开发者还定义电路设计的非安全关键部分。在一些其它实施方式中,电路设计的安全关键部分能够有意地位于整个 FPGA 或其它可编程逻辑设计的已知平面布置图区中。在一些实施方式中,CAD 工具然后在 1210 处生成包括电路设计的非安全关键部分但不包括在电路设计的安全关键部分内的模块的修改的项目层次。在一些实施方式中,安全关键部分被用 HDL 表示为使实现该安全关键部分的模块实例化的调用,所述安全关键部分它本身能够包括若干子模块。

[0069] 在一些实施方式中,CAD 工具然后在 1212 处基于在 1210 处生成的经修改的项目层次、或者基于关于安全关键部分与非安全关键部分之间的交互的其它功能或模块级信息在 HDL 实现的设计中生成电路设计的非安全关键部分与电路设计的安全关键部分之间的接口。在一些这样的实施方式中,能够用边界 LUT 实现该接口。例如,开发者能够设计电路设计,或者 CAD 工具能够生成电路设计,使得它在电路设计的安全关键部分与非安全关键部分之间的边界处包括 LUT。边界 LUT 将从非安全关键部分输入到边界 LUT 的信号和资源映射为安全关键部分内的资源并且将从安全关键部分输出的信号和资源映射为非安全关键部分内的资源。以这种方式,虽然 CAD 工具一般地需要具有它需要给到特定边界 LUT 的特定信号定路线的知识,但是 CAD 工具不需要在安全关键部分内在输入 LUT 下游或在输出 LUT 上游执行的映射的知识。

[0070] 在一些实施方式中,过程 1200 在其中 CAD 工具对电路设计的安全关键部分、非安全关键部分以及安全关键部分与非安全关键部分之间的接口的 HDL 实现的表示执行合成操作的 1214 处继续。在一些实施方式中,在 1214 处合成的结果是物理上可实现的门网表。在一些实施方式中,CAD 工具然后在 1216 处对已合成设计执行验证操作。在一些实施方式中,在 1216 处的验证之后,已合成设计能够被提供给 CAD 工具内的物理设计工具或程序,其中在 1218 处布局和布线操作被执行。在一些实施方式中,CAD 工具在 1220 处基于布局和布线操作的输出来生成排他列表。该排他列表包括将被保留仅用于供电路设计的安全关键部分使用的资源中的全部。

[0071] 可编程逻辑配置级能够采用布局和布线工具的输出,并且在 1222 处为 FPGA 或其它可编程逻辑器件生成配置设定。如上所述,能够在各种实施方式中使用各种合成、布局和布线以及可编程逻辑配置工具。在一些实施方式中,在 1222 处配置级的输出包括一个或多个编程文件。例如,配置级可以在 1222 处以用于安全关键部分的编程位的形式输出至少包括配置设定的第一编程文件。在一些这样的实施方式中,虽然第一编程文件对于给定物理 FPGA 硬件架构包括用于 CRAM 中的全部的所有位,但是对于开发者的设计的安全关键部分、非安全关键部分或接口未被生成的编程文件中的所有位分配了默认值。在一些实施方式中,配置级可以附加地输出第二编程文件,其对于安全关键部分仅包括编程位但对于包括针对非安全关键部分所指定的位的设计的其它位中的全部包括默认值。

[0072] 在一些实施方式中,然后这时能够使用编程位再次验证电路设计的定时、功率和/或其它功能安全要求。例如,用于安全关键部分、非安全关键部分以及安全关键部分与非安全关键部分之间的接口的编程位然后能够被编程到之后经受验证的物理 FPGA 器件中以确保与功能安全要求兼容。验证的结果然后能够被提供给诸如认证机构(诸如莱茵 TÜV)的第三方。附加地或替换地,在一些实施方式中,第三方(诸如莱茵 TÜV)能够执行验证。在一些实施方式中,包括用于安全关键部分和非安全关键部分两者以及接口的配置设计的第一编程文件被提供给认证机构以确保与功能安全要求兼容。

[0073] 在一些实施方式中,过程 1200 在其中生成与上面参考图 6A 示出和描述的位掩码类似的位掩码的 1224 处继续。位掩码一般地标识被安全关键部分(与如参考图 6A 所描述的秘密部分相反)使用的开发者的编程文件中的位。因为对于电路设计的安全关键部分所要求的编程位可以构成整个编程文件的仅一部分,所以位掩码被用来标识被用来配置安全关键部分的编程文件中的那些位。在其中电路设计的安全关键部分将被配置到仅单个逻辑块或其它部分中的一些实施方式中,位掩码能够仅包括与该逻辑块或其它部分相关联的位。在一些其它实施方式中,位掩码能够包括用于整个编程文件的位。例如,具有逻辑值“1”的位值的位掩码中的所有位能够指示编程文件中的对应位将在配置电路设计的安全关键部分时被使用。相比之下,具有逻辑值“0”的位值的掩码中的所有位都能够指示编程文件中的对应位将不在配置安全关键部分时被使用—这些位可以被用来例如对开发者的电路设计的非安全关键部分进行编程或者用来对最后的 FPGA 设计的其它部分进行编程。

[0074] 如上所述,情况常常或典型地是,开发者或用户可能希望对电路设计的非安全关键部分做出改变而不对安全关键部分做出改变。然而,如上所述,即使安全关键部分尚未被有意地改变,常规技术也一般地要求整个设计的重新认证。但采用本文中所描述的实施方式,开发者能够对非安全关键部分做出改变而无需重新认证。例如,开发者能够在 1226 处

对电路设计的非安全关键部分的 HDL 或其它抽象级表示做出改变,同时确保非安全关键部分的 HDL 表示或其它抽象级表示正确地连接到在 1212 处生成的接口。

[0075] 在一些实施方式中,在 1228 处 CAD 工具然后对非安全关键部分和接口执行另一合成操作。在一些实施方式中,在 1228 处合成的结果是与非安全关键部分相对应的物理上可实现的网表。在一些实施方式中,CAD 工具然后在 1230 处对在 1228 处被合成的设计执行验证操作。在 1232 处,CAD 或其它工具然后将在 1228 处生成的与修订后的非安全关键部分相对应的网表和与经先前认证的安全关键部分相对应的在 1214 处生成的网表的部分合并。

[0076] 图 13 示出了图示用于将修订后的非安全关键部分与先前验证的安全关键部分合并的示例过程 1300 的流程图。在一些实施方式中,过程 1300 是过程 1200 的步骤 1232 的示例实施方式。在一些实施方式中,过程 1300 在其中将在 1228 处合成的修订后的非安全关键部分的网表提供给 CAD 工具内的物理设计工具或程序的 1302 处开始。使用该工具,在 1228 处合成的网表然后在 1304 处与在 1214 处生成的网表合并。也就是说,使得在在 1226 处做出改变之前修订后的非安全关键部分的新近重新合成的网表代替对应于非安全关键部分的在 1214 处生成的网表的部分。在这样的实施方式中,对应于经先前验证的安全关键部分的在 1214 处生成的网表的部分未被变更。在 1306 处,然后对经合并的网表执行布局和布线操作。在布局和布线操作期间,CAD 工具能够使用早先生成的排除列表来保证它不使用由安全关键部分所保留的资源来给设计的安全关键部分与非安全关键部分之间的输入和输出定路线。因此,在这样的实施方式中,在 1306 处执行的布局和布线操作对于根据在 1218 处早先执行的布局和布线操作所确定的经先前验证的安全关键部分保持逻辑和布线的布局。

[0077] 可编程逻辑配置级能够采用布局和布线工具的输出,并且在 1308 处为经先前验证的安全关键部分、修订后的非安全关键部分以及接口生成配置设定。在一些实施方式中,在 1308 处配置级的输出是包括形式为用于经先前验证的安全关键部分、修订后的非安全关键部分以及接口的编程位的配置设定的编程文件。以这种方式,作为结果的合并的编程文件包含与当它们被先前认证时相同的用于安全关键部分的配置编程位以及用于修订后的非安全关键部分的配置编程位。每当开发者或用户期望改变非安全关键部分而无需重新认证时能够重复这个过程。

[0078] 图 14 示出了图示用于将修订后的非安全关键部分与经先前验证的安全关键部分合并的另一示例过程 1400 的流程图。在一些实施方式中,过程 1400 是过程 1200 的步骤 1232 的示例实施方式。在一些实施方式中,过程 1400 在其中将在 1228 处合成的修订后的非安全关键部分的网表提供给 CAD 工具内的物理设计工具或程序的 1402 处开始,其中在 1404 处,布局和布线操作然后被执行。在布局和布线操作期间,CAD 工具能够使用早先生成的排除列表来保证它不使用由安全关键部分所保留的资源来给设计的安全关键部分与修订后的非安全关键部分之间的输入和输出定路线。因此,在这样的实施方式中,在 1404 处执行的布局和布线操作对于根据在 1218 处早先执行的布局和布线操作所确定的经先前验证的安全关键部分保持逻辑和布线的布局。可编程逻辑配置级能够采用布局和布线工具的输出,并且在 1406 处为修订后的非安全关键部分和接口生成配置设定。在一些实施方式中,在 1406 处配置级的输出是包括形式为用于非安全关键部分和接口的编程位的配置设定的

第三编程文件。在一些实施方式中，CAD 工具能够通过将所生成的用于修订后的非安全关键部分的配置设定与在 1224 处生成的位掩码相比较来验证工具正确地遵守了排除列表。

[0079] 在一些实施方式中，在 1408 处，CAD 或其它工具然后将包括用于经先前验证的安全关键部分的编程位的在 1222 处生成的第二编程文件与包含用于修订后的非安全关键部分和接口的编程位的在 1406 处生成的第三编程文件合并。例如，像在参考图 9 所图示的过程 900 中类似地描述的那样，两个编程文件能够使用在 1224 处生成的位掩码而被合并，安全关键部分与秘密部分类似，非安全关键部分与公共部分类似。以这种方式，作为结果的合并的编程文件包含当它们被认证机构认证时的用于安全关键部分的原先生成的配置编程位以及用于修订后的非安全关键部分的配置编程位。每当开发者或用户期望改变非安全关键部分而无需重新认证时能够重复这个过程。

[0080] 认证机构或其它第三方可以要求安全关键部分保持不变并且有权认证的证据。为了确保这样的符合性，能够做出与安全关键部分相对应的整个编程文件的位流部分的检查。例如，在 1224 处生成的位掩码以及(取决于实施方式)在步骤 1308 处或在步骤 1408 处生成的最后的编程文件还能够被提供给认证机构。认证机构然后能够使用位掩码来执行最后的编程文件与被认证机构先前认证的在 1222 处生成的编程文件之间的逐位比较，以确保与安全关键部分相对应的编程位保持不变并且有权认证。

[0081] 此外，在一些实施方式中，出于安全性或跟踪目的，可能期望生成参考图 4 到 11 所描述的秘密部分、特别参考图 12 到 14 所描述的安全关键部分和 / 或上面所描述的位掩码中的任一个的散列。例如，用于秘密部分、安全关键部分或上面所描述的位掩码中的任一个的编程位能够通过诸如 MD5 消息摘要算法的密码散列函数来操作以生成加密的散列值。

[0082] 图 15 示出了一般地示出能够被用来实现上面所描述的 CAD 和其它工具的功能性中的一些或全部的计算机系统的图解框图表示。计算机系统 1500 包括被耦合到包括存储器 1504 (典型地是随机存取存储器或“RAM”)、存储器 1506 (典型地是只读存储器或“ROM”) 的装置的任何数量的处理器 1502 (还被称为中央处理单元, 或 CPU)。处理器 1502 能够被配置成生成电子设计。如本领域中众所周知的那样，存储器 1506 用来将数据和指令单向地传送到 CPU 并且存储器 1504 被典型地用来以双向方式传送数据和指令。

[0083] 这些存储器装置都可以包括上面所描述的任何适合类型的计算机可读介质。大容量存储装置 1508 还被双向地耦合到 CPU1502 并且提供附加的数据存储容量以及可以包括上面所描述的计算机可读介质中的任一个。大容量存储装置 1508 可以被用来存储程序、数据等，并且典型地是比存储器慢的诸如硬盘的辅助存储介质。大容量存储装置 1508 能够被用来保持预封装的逻辑或知识产权功能的库或数据库以及关于生成特定配置的信息。将理解的是，在适当的情况下，保持在大容量存储装置 1508 内的信息可以像虚拟存储器那样作为存储器 1504 的一部分被以标准方式结合。诸如 CD-ROM1514 的特定大容量存储装置还可以将数据单向地传递给 CPU。

[0084] CPU1502 还被耦合到包括一个或多个输入 / 输出装置的一个或多个接口 1510, 所述一个或多个输入 / 输出装置诸如视频监视器、轨迹球、鼠标、键盘、麦克风、触敏显示器、换能器卡阅读机、磁或纸带阅读机、平板、触针、语音或手写识别器, 或其它众所周知的输入设备, 当然也诸如其它计算机。CPU1502 可以是设计工具处理器。最后, CPU1502 可选地可以使用如一般地在 1512 处所示出的网络连接被耦合到计算机或电信网络。采用这样的网

络连接,设想了 CPU1502 在执行上面描述的过程步骤的过程中可能从网络接收信息,或者可能将信息输出到网络。应该注意的是,系统 1500 还可能与用于将完成的设计转移到可编程芯片上的装置相关联。上面描述的装置和材料将为计算机硬件和软件领域的技术人员所熟悉。

[0085] 尽管为了方便上面以单数方式描述了许多组件和过程,但是本领域的技术人员将了解的是,多个组件和重复的过程还能够被用来实践本公开的技术。虽然已经参考其特定实施方式特别示出并描述了特定实施方式,但是本领域的技术人员将理解的是,在不背离本发明的精神或范围的情况下可以做出所公开的实施方式的形式和细节上的改变。例如,本发明的实施方式可以与各种组件一起被采用,并且不应该被限制于上面所提到的实施方式。另外,以上过程中的一个或多个中的特定步骤可以被省略、执行多次、并行地执行以及以不同的次序执行。因此意图是,本发明被解释成包括落入本发明的真实精神和范围内的所有变化和等同物。

[0086] 本公开要求由 Titley 等人于 2013 年 4 月 19 日提交的标题为“PARTITIONING DESIGNS TO FACILITATE CERTIFICATION”的共同待决的美国专利申请序号 13/866,624 (代理人案号 ALTRP296X1/A03942CIP)的优先权,其通过引用整体地结合于此以用于所有目的。



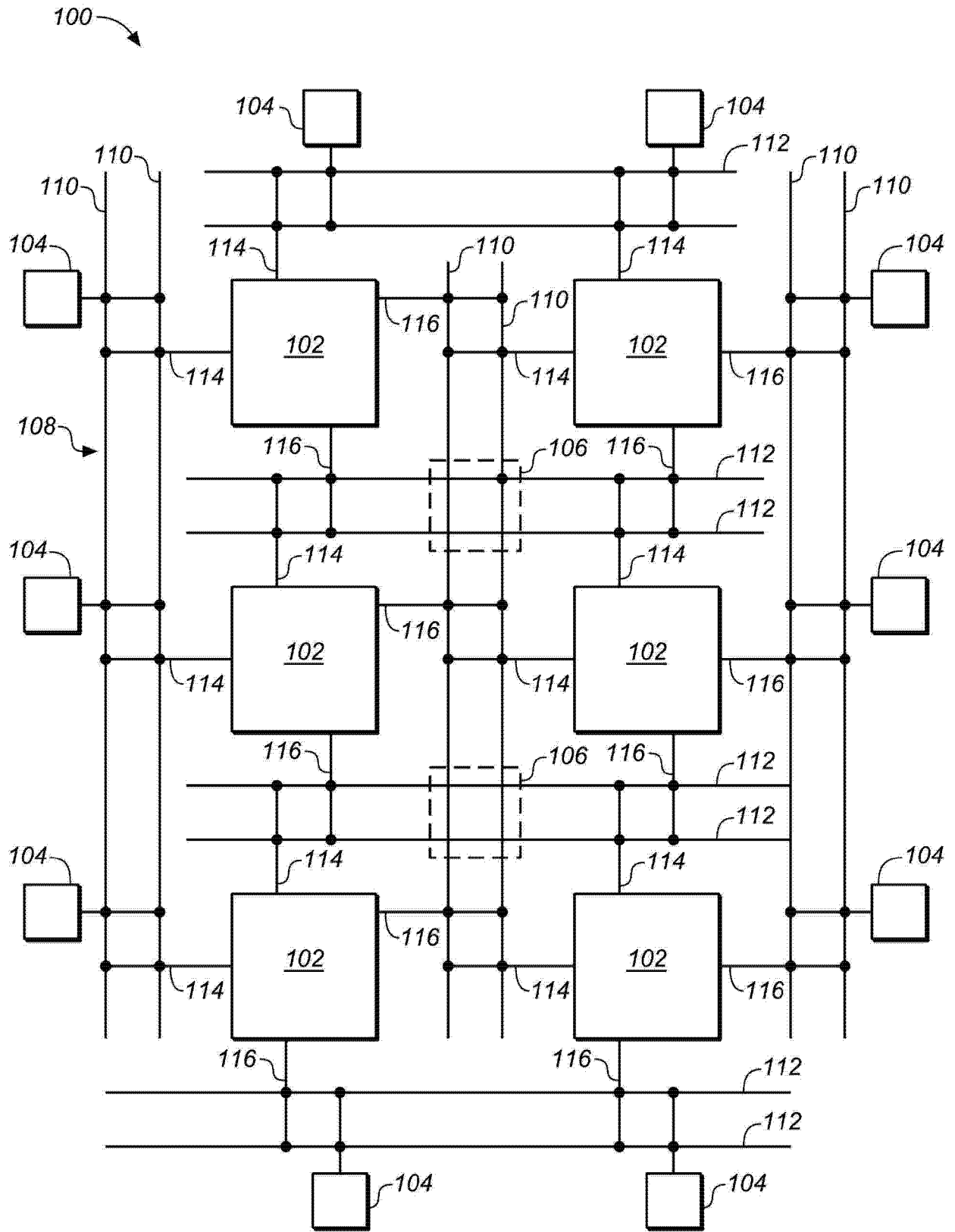


图 1

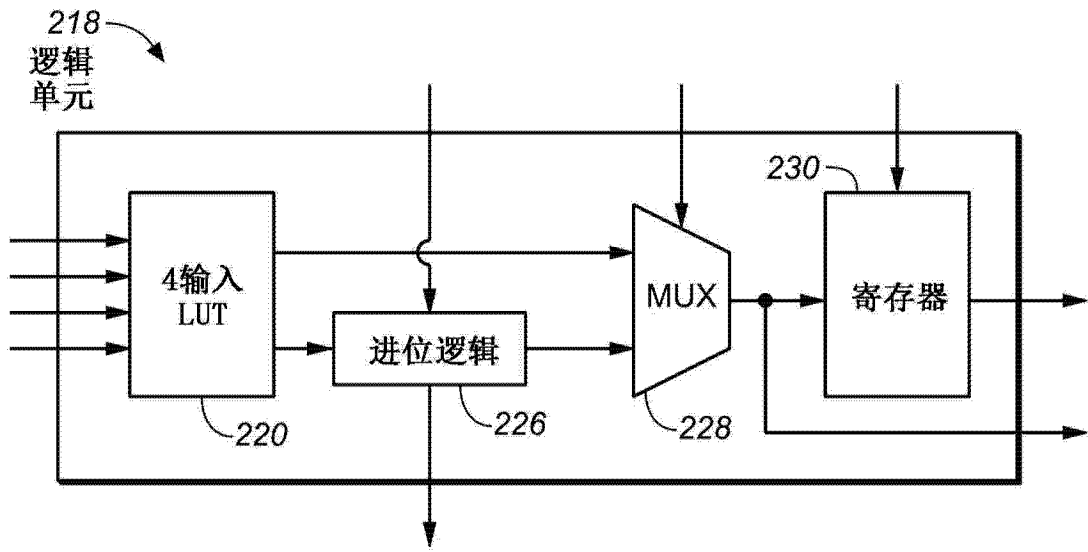


图 2

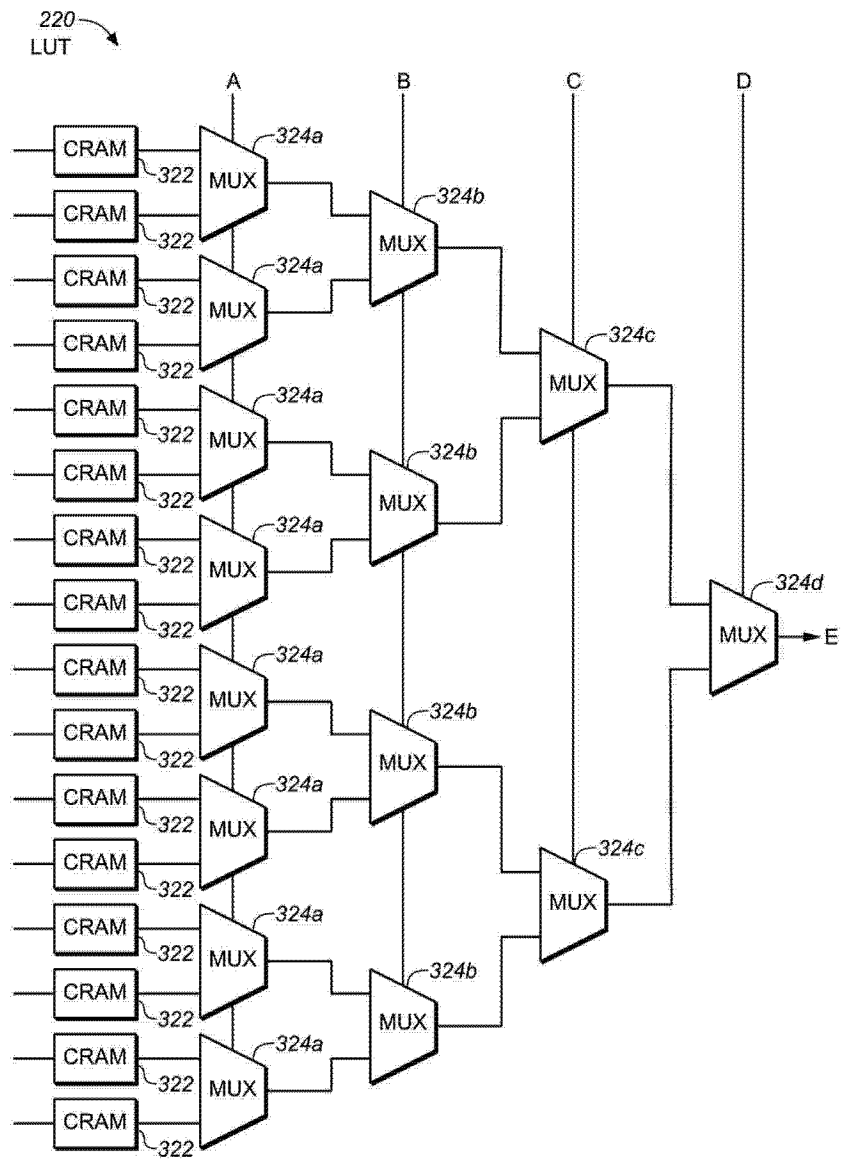


图 3

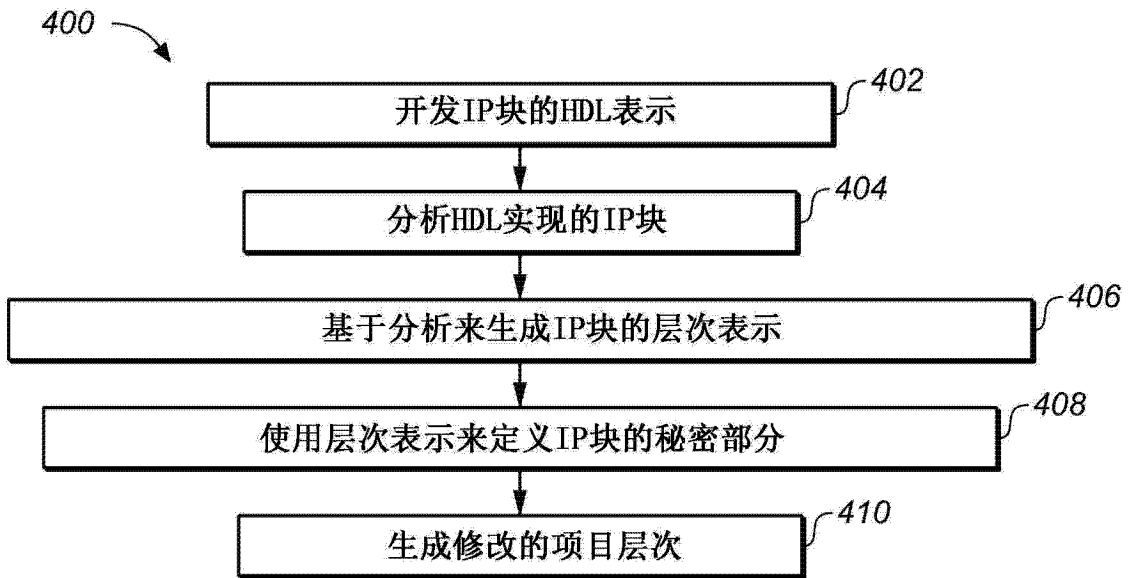


图 4

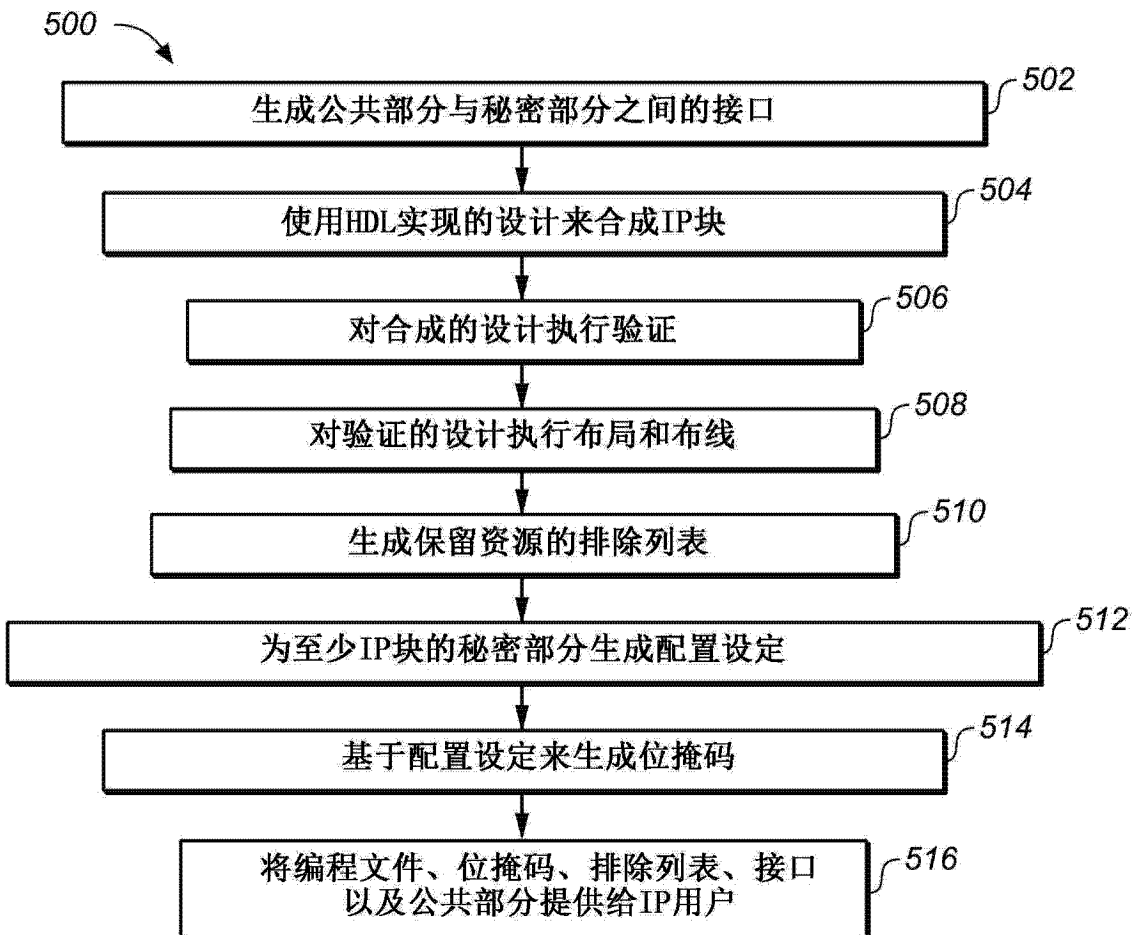


图 5

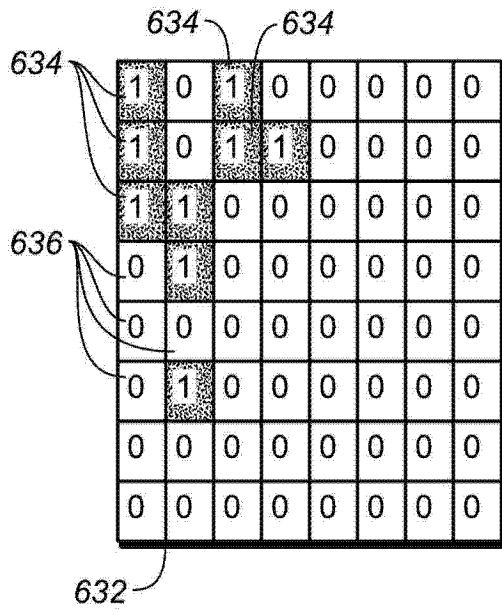


图 6A

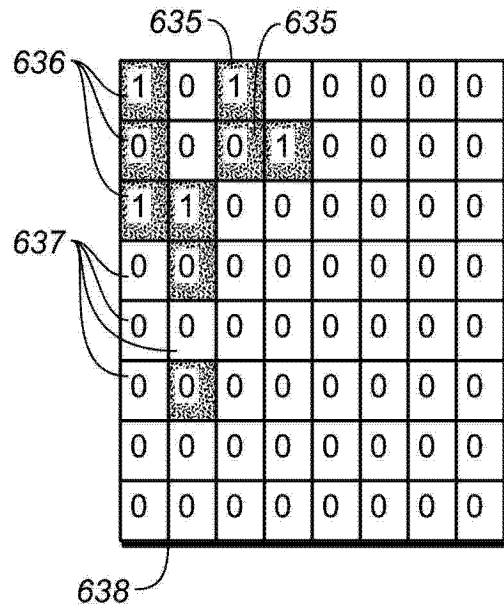


图 6B

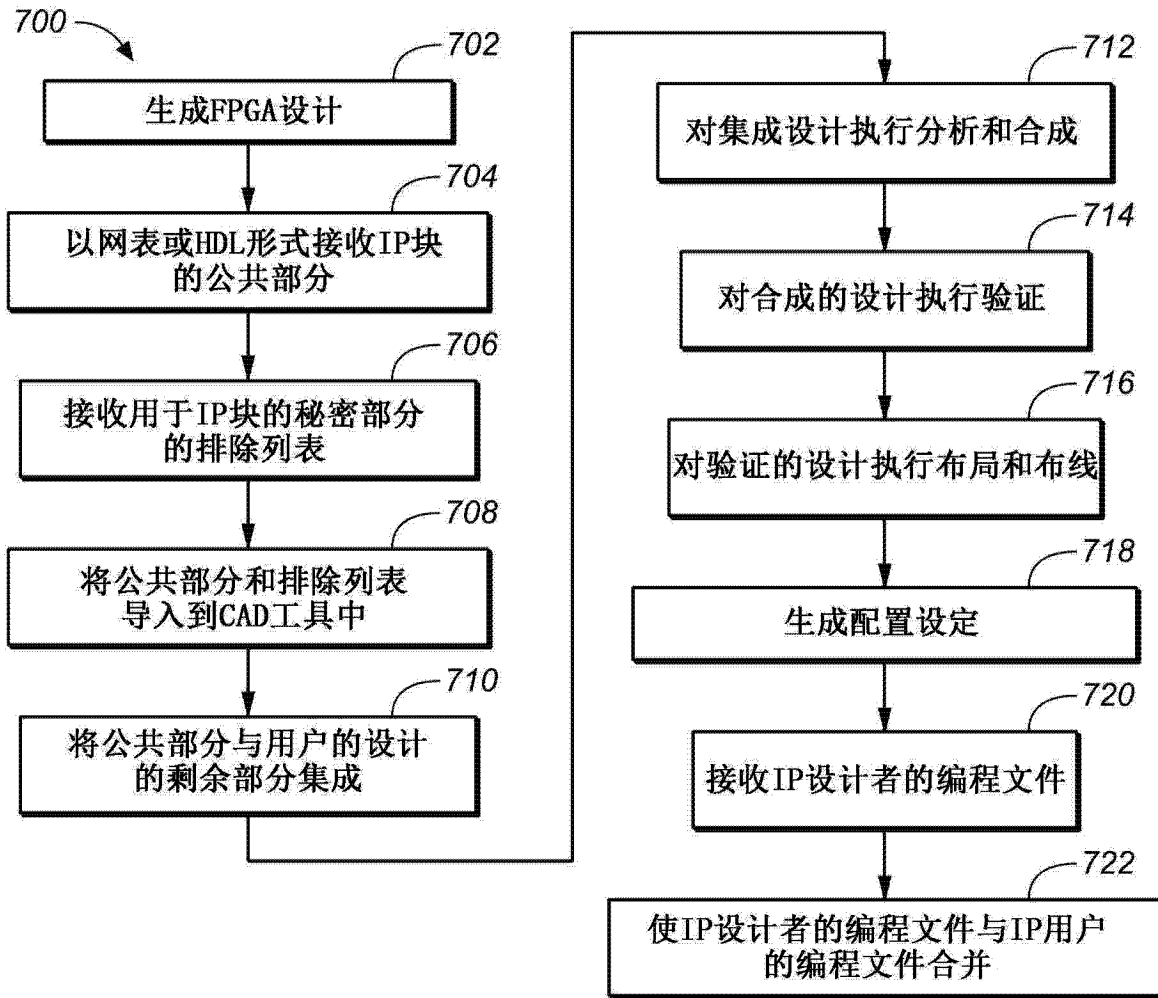


图 7

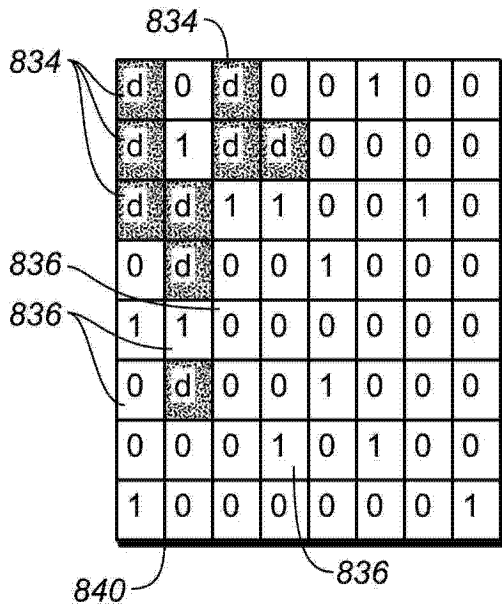


图 8A

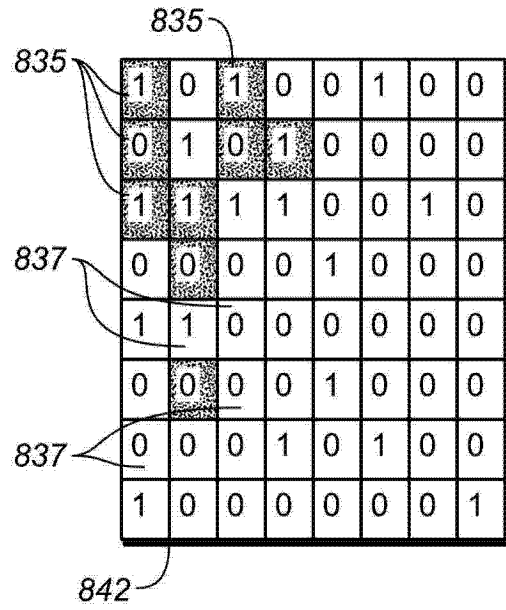


图 8B

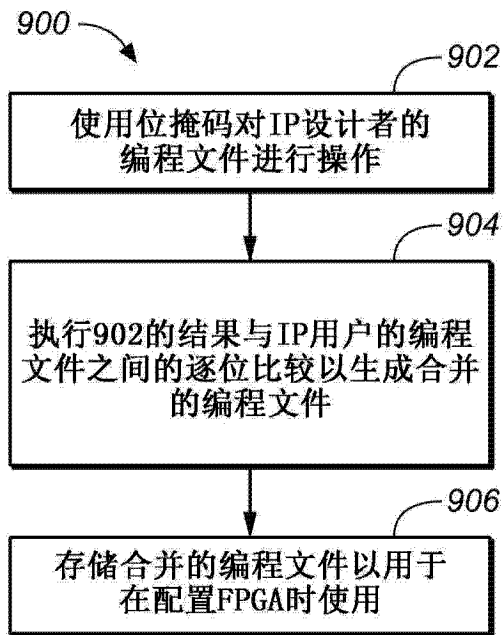


图 9

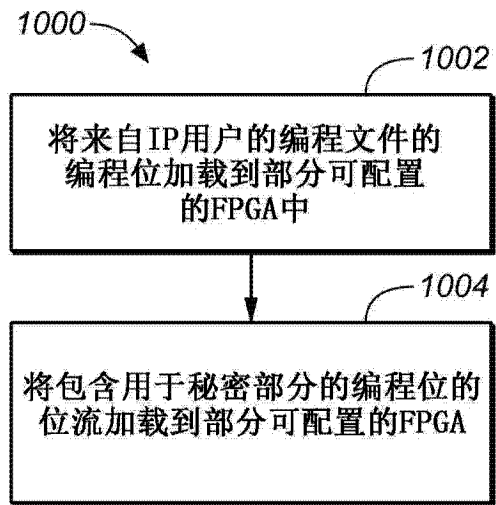


图 10

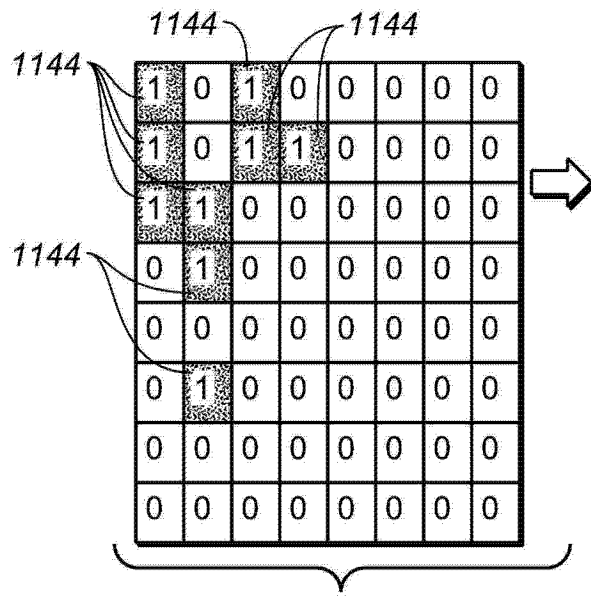


图 11A

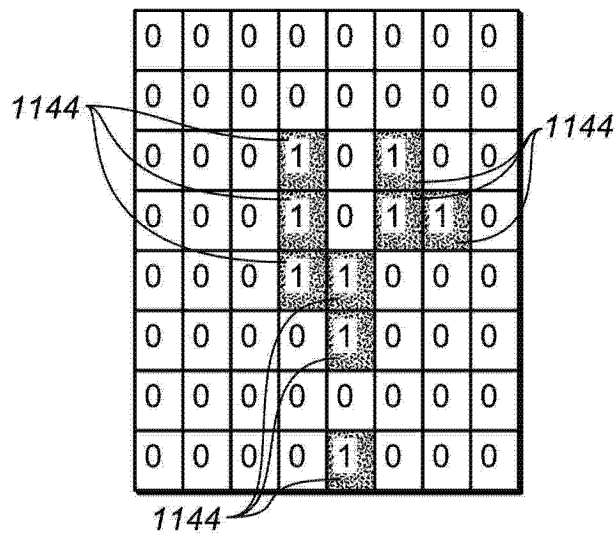


图 11B



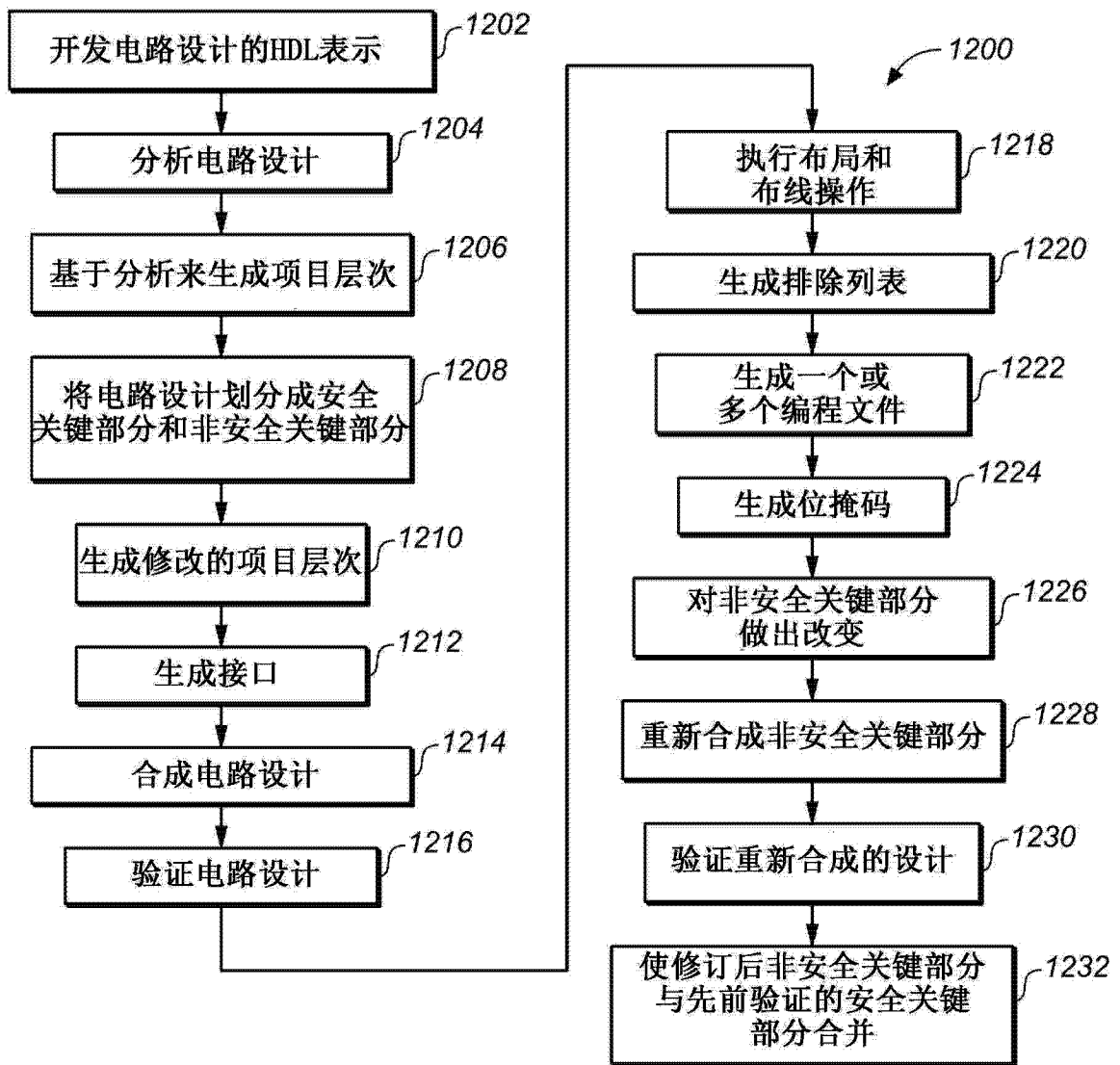


图 12

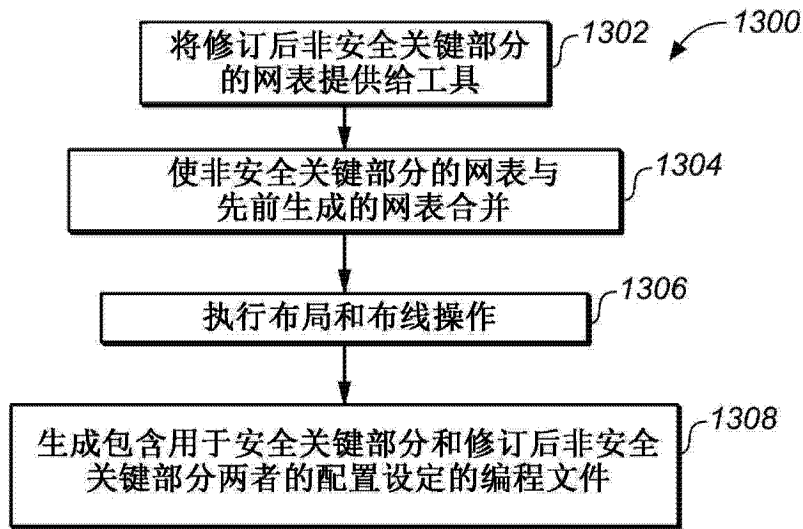


图 13

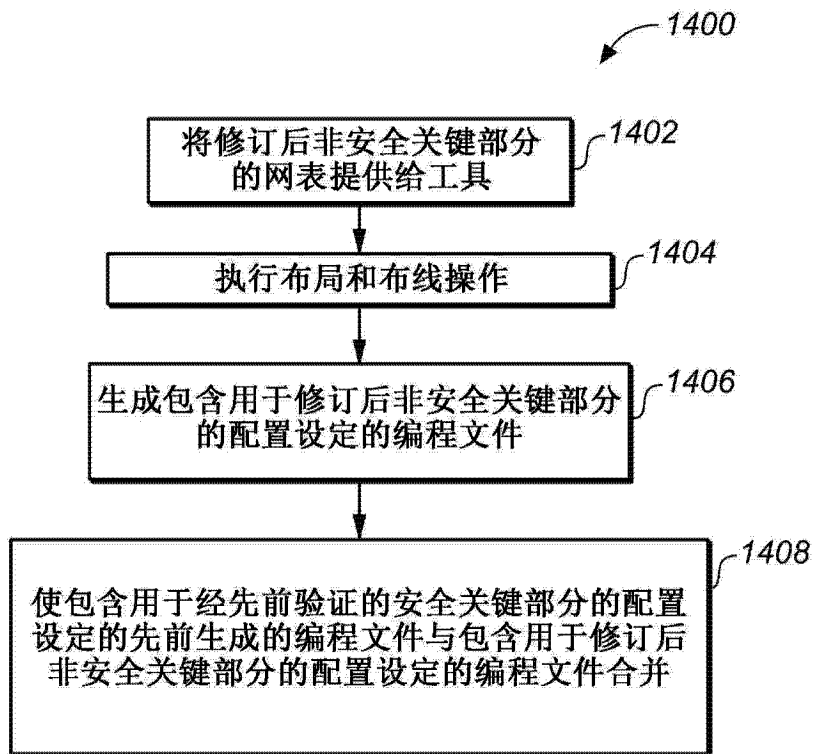


图 14

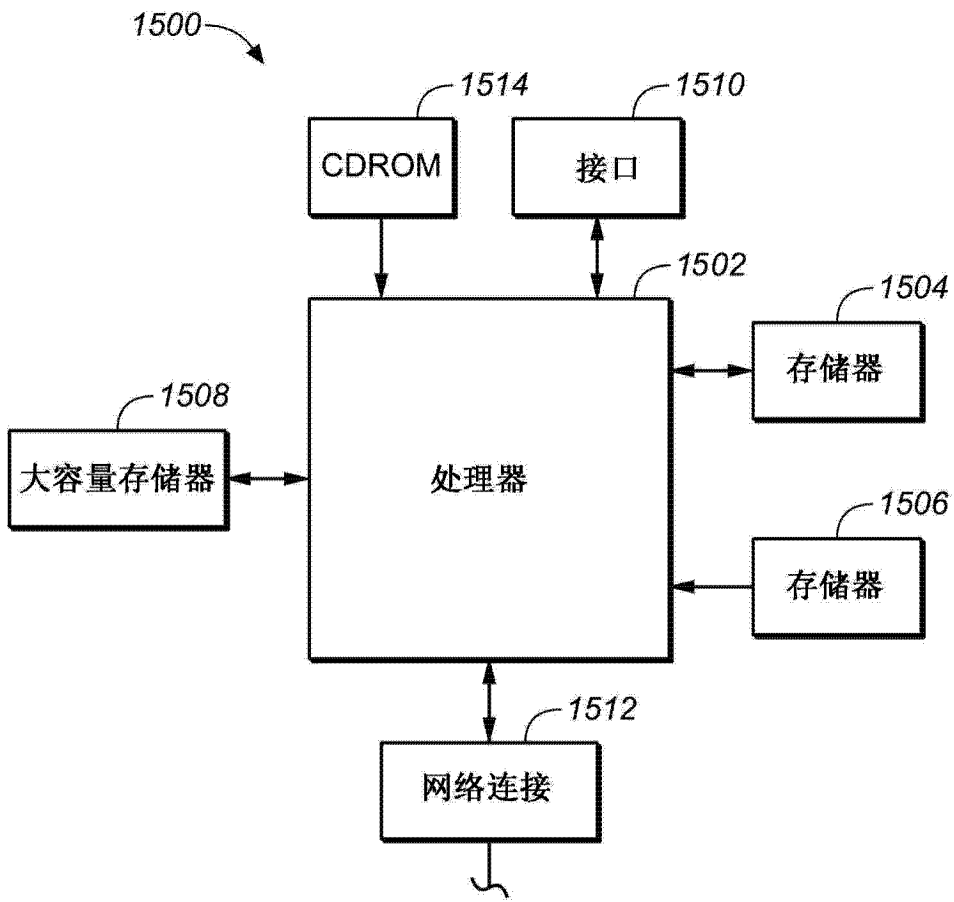


图 15