



(12) 发明专利

(10) 授权公告号 CN 109299530 B

(45) 授权公告日 2023. 07. 25

(21) 申请号 201811062104.X

(22) 申请日 2018.09.12

(65) 同一申请的已公布的文献号
申请公布号 CN 109299530 A

(43) 申请公布日 2019.02.01

(73) 专利权人 成都奥卡思微电科技有限公司
地址 610000 四川省成都市高新区天府大道中段1366号天府软件园E6-1-16\17

(72) 发明人 袁军 曹皖林

(74) 专利代理机构 成都华风专利事务所(普通合伙) 51223
专利代理师 徐丰 张巨箭

(51) Int. Cl.
G06F 30/398 (2020.01)
G06F 111/04 (2020.01)

(56) 对比文件

US 2001010091 A1, 2001.07.26

US 2013117722 A1, 2013.05.09

US 2014282321 A1, 2014.09.18

US 5572437 A, 1996.11.05

US 7130783 B1, 2006.10.31

US 2015331786 A1, 2015.11.19

审查员 王海容

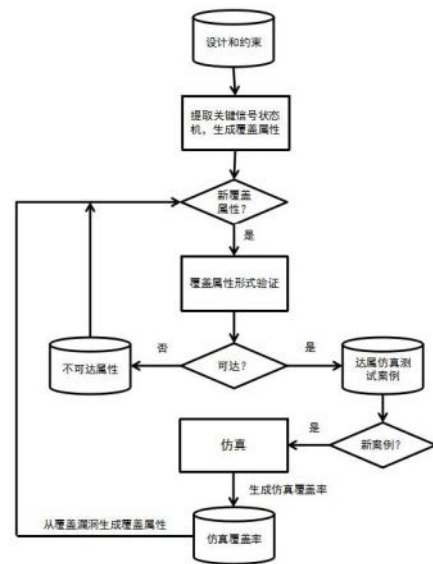
权利要求书2页 说明书6页 附图3页

(54) 发明名称

一种仿真测试案例生成方法、系统、存储介质和终端

(57) 摘要

本发明公开了一种仿真测试案例生成方法、系统、存储介质和终端,方法包括以下步骤:设置基于设计输入端规范的约束;读入将要被验证的设计;提取所述设计的关键信号及相应的状态机;生成所述状态机的相应的可达性属性;对所述设计、所述约束和所述可达性属性进行形式验证;其中,可达的可达性属性用于生成测试案例;对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例;对最小化操作后的测试案例进行仿真并计算仿真覆盖率;对仿真出的漏洞和不可达属性采用形式验证进行分析。本发明早期运用形式验证方法自动生成测试案例,尽可能的在人工介入之前就达到一个较高的覆盖率。



1. 一种仿真测试案例生成方法,其特征在于:包括以下步骤:
设置基于设计输入端规范的约束;
读入将要被验证的设计;
提取所述设计的关键信号及相应的状态机;
生成所述状态机的相应的可达性属性;
对所述设计、所述约束和所述可达性属性进行形式验证;其中,可达的可达性属性用于生成测试案例,不可达的可达性属性用于纠正覆盖率标准;
对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例;经过本步骤优化后的每一个测试案例由于可能有未被赋值输入可以被随机赋值而成为多个测试,这样一个案例因为设计输入的随机选取得以被多次应用;
对最小化操作后的测试案例进行仿真并计算仿真覆盖率;对仿真出的漏洞和不可达属性重新生成覆盖属性后,采用形式验证进行分析。
2. 根据权利要求1所述的一种仿真测试案例生成方法,其特征在于:所述的关键信号包括:本身是一个设计中状态机的状态信号、先入先出的控制信号、指令指针、状态存储器和宽度超过一定值的总线中的一种或者多种。
3. 根据权利要求1所述的一种仿真测试案例生成方法,其特征在于:提取所述设计的关键信号后,相应的状态机通过网表的结构分析和/或逻辑分析继而被提取。
4. 根据权利要求1所述的一种仿真测试案例生成方法,其特征在于:在所述的提取所述设计的关键信号及相应的状态机的步骤之后,还包括:
删除部分关键信号及相应的状态机和/或增加更多的关键信号及相应的状态机。
5. 一种仿真测试案例生成系统,其特征在于:包括:
约束设置模块:用于设置基于设计输入端规范的约束;
设计读入模块:用于读入将要被验证的设计;
提取模块:用于提取所述设计的关键信号及相应的状态机;
可达性属性生成模块:用于生成所述状态机的相应的可达性属性;
测试案例生成模块:用于对所述设计、所述约束和所述可达性属性进行形式验证;其中,可达的可达性属性用于生成测试案例,不可达的可达性属性用于纠正覆盖率标准;
测试案例优化模块:对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例;经过优化后的每一个测试案例由于可能有未被赋值输入可以被随机赋值而成为多个测试,这样一个案例因为设计输入的随机选取得以被多次应用;
仿真模块:对最小化操作后的测试案例进行仿真并计算仿真覆盖率;
覆盖属性生成与形式验证模块:对仿真出的漏洞和不可达属性重新生成覆盖属性后,采用形式验证进行分析。
6. 根据权利要求5所述的一种仿真测试案例生成系统,其特征在于:所述的关键信号包括:本身是一个设计中状态机的状态信号、先入先出的控制信号、指令指针、状态存储器和宽度超过一定值的总线中的一种或者多种。
7. 根据权利要求5所述的一种仿真测试案例生成系统,其特征在于:提取所述设计的关

键信号后,相应的状态机通过网表的结构分析和/或逻辑分析继而提取。

8.根据权利要求5所述的一种仿真测试案例生成系统,其特征在于:所述的系统还包括:

信号与状态机选择模块:用于对提取模块提取到的关键信号及相应的状态机进行处理,包括删除部分关键信号及相应的状态机和/或增加更多的关键信号及相应的状态机。

9.一种存储介质,其上存储有计算机指令,其特征在于:所述计算机指令运行时执行权利要求1至4中任一项所述的一种仿真测试案例生成方法的步骤。

10.一种终端,包括存储器和处理器,所述存储器上存储有可在所述处理器上运行的计算机指令,其特征在于,所述处理器运行所述计算机指令时执行权利要求1至4中任一项所述的一种仿真测试案例生成方法的步骤。

一种仿真测试案例生成方法、系统、存储介质和终端

技术领域

[0001] 本发明涉及集成电路功能验证领域,尤其一种仿真测试案例生成方法、系统、存储介质和终端。

背景技术

[0002] 仿真验证是芯片设计功能验证的重要手段。仿真验证的第一个步骤就是测试案例的生成。传统的测试案例生成方法是根据设计规范手册去人工生成针对每一条规范的测试案例,且测试案例要满足一定的覆盖率要求,比如一定比率的设计代码必须被在测试中被运行。人工生成所有的测试案例即费时又容易出错,还往往达不到规定的覆盖率要求。

[0003] 现代的仿真测试中基于约束的测试案例生成比较常用,比如SystemVerilog中的UVM方法。好处是测试案例不需要逐条生成。但是事先必须建立起符合设计规范的约束,也是一项费时的工作。覆盖率的提升又必须通过后期对约束的调试来实现。

[0004] 由于功能验证难度理论上随设计规模成指数级增长,其耗用的时间已占整个芯片设计过程的70%。作为功能验证重要手段的仿真,现阶段其覆盖率无论是前期用人工方案或基于约束的方案尽量去提升,还是后期通过形式验证技术去弥补,都是一个比较费时和非自动化的过程。自动仿真覆盖测试的生成将极大的提高设计验证乃至整个设计过程的效率。

发明内容

[0005] 本发明的目的在于克服现有技术的不足,提供一种仿真测试案例生成方法、系统、存储介质和终端,提高设计验证乃至整个设计过程的效率。

[0006] 本发明的目的是通过以下技术方案来实现的:一种仿真测试案例生成方法,包括以下步骤:

[0007] 设置基于设计输入端规范的约束;

[0008] 读入将要被验证的设计;

[0009] 提取所述设计的关键信号及相应的状态机;

[0010] 生成所述状态机的相应的可达性属性;

[0011] 对所述设计、所述约束和所述可达性属性进行形式验证;其中,可达的可达性属性用于生成测试案例,不可达的可达性属性用于纠正覆盖率标准;

[0012] 对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例;

[0013] 对最小化操作后的测试案例进行仿真并计算仿真覆盖率;对仿真出的漏洞和不可达属性重新生成覆盖属性后,采用形式验证进行分析。

[0014] 进一步地,所述的关键信号包括:本身是一个设计中状态机的状态信号、先入先出的控制信号、指令指针、状态存储器和宽度超过一定值的总线中的一种或者多种。

[0015] 进一步地,提取所述设计的关键信号后,相应的状态机通过网表的结构分析和/或

逻辑分析继而提取。

[0016] 进一步地,在所述的提取所述设计的关键信号及相应的状态机的步骤之后,还包括:

[0017] 删除部分关键信号及相应的状态机和/或增加更多的关键信号及相应的状态机。

[0018] 本发明还提供一种仿真测试案例生成系统,包括:

[0019] 约束设置模块:用于设置基于设计输入端规范的约束;

[0020] 设计读入模块:用于读入将要被验证的设计;

[0021] 提取模块:用于提取所述设计的关键信号及相应的状态机;

[0022] 可达性属性生成模块:用于生成所述状态机的相应的可达性属性;

[0023] 测试案例生成模块:用于对所述设计、所述约束和所述可达性属性进行形式验证;

其中,可达的可达性属性用于生成测试案例,不可达的可达性属性用于纠正覆盖率标准;

[0024] 测试案例优化模块:对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例;

[0025] 仿真模块:对最小化操作后的测试案例进行仿真并计算仿真覆盖率;

[0026] 覆盖属性生成与形式验证模块:对仿真出的漏洞和不可达属性重新生成覆盖属性后,采用形式验证进行分析。

[0027] 进一步地,所述的关键信号包括:本身是一个设计中状态机的状态信号、先入先出的控制信号、指令指针、状态存储器和宽度超过一定值的总线中的一种或者多种。

[0028] 进一步地,提取所述设计的关键信号后,相应的状态机通过网表的结构分析和/或逻辑分析继而提取。

[0029] 进一步地,所述的系统还包括:

[0030] 信号与状态机选择模块:用于对提取模块提取到的关键信号及相应的状态机进行处理,包括删除部分关键信号及相应的状态机和/或增加更多的关键信号及相应的状态机。

[0031] 本发明还提供一种存储介质,其上存储有计算机指令,所述计算机指令运行时执行所述的一种仿真测试案例生成方法的步骤。

[0032] 本发明还提供一种终端,包括存储器和处理器,所述存储器上存储有可在所述处理器上运行的计算机指令,所述处理器运行所述计算机指令时执行所述的一种仿真测试案例生成方法的步骤。

[0033] 本发明的有益效果是:

[0034] (1) 本发明主要的优点是早期运用形式验证方法自动生成测试案例,尽可能的在人工介入之前就达到一个较高的覆盖率。

[0035] 虽然本发明也用到约束,但是这些约束直接来源于对设计输入端的规范,不同于基于约束的测试生成中的约束,后者的约束往往是针对某一类的测试,不同类别的测试需要不同的约束。

[0036] 本发明的测试案例优化使得测试最小化,这样一个案例因为设计输入的随机选取得以被多次应用,以达到更大的覆盖率。

[0037] 另外在最后一步我们也用到现有基于形式验证方法的漏洞和不可达,这样我们把基于约束、随机测试及漏洞和不可达分析有机的结合到基于自动机自动提取的测试生成流程中,最大效率的发挥了四者的优点。

[0038] (2)本发明的方法步骤通过自动的方式实现,无需外围人员进行干涉即可实现。本发明提供一种仿真测试案例生成系统、存储介质和终端也解决了相应的技术问题。

附图说明

[0039] 图1为本发明方法流程图;

[0040] 图2为本发明状态机属性示意图;

[0041] 图3为本发明系统框图;

[0042] 图4为本发明终端连接示意图。

具体实施方式

[0043] 下面结合附图对本发明的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0044] 此外,下面所描述的本发明不同实施方式中所涉及的技术特征只要彼此之间未构成冲突就可以相互结合。

[0045] 实施例1

[0046] 实施例1提供一种仿真测试案例生成方法,该方法用于实现仿真测试案例的自动生成、覆盖率仿真检测及后续分析。具体地,如图1所示,所述方法包括以下步骤:

[0047] S1:设置基于设计输入端规范的约束。

[0048] 其中,相较于在测试案例生成过程中的约束,本申请采用基于设计输入端规范的约束。在测试案例生成过程中的约束往往是针对某一类的测试,而针对不同类别的测试需要不同的约束。

[0049] S2:读入将要被验证的设计。

[0050] 通常情况下,所述的设计包括包括任何芯片设计内容,比如CPU、GPU、嵌入控制器等等。而通常情况下,芯片设计一般用硬件设计语言描述,比如SystemVerilog,VHDL。

[0051] 而在本实施例中,对于被验证的设计,是由在EDA工具对设计进行验证。

[0052] S3:对所述设计进行分析,提取所述设计的关键信号及相应的状态机。

[0053] 其中,在本实施例的优选方案中,所述的关键信号的默认条件包括以下条件中的一个或者多个:本身是一个设计中状态机的状态信号、先入先出的控制信号、指令指针、状态存储器、宽度超过一定值的总线。

[0054] 另外,用户可以指定任何其他感兴趣的信号,并改设总线的宽带的阈值。

[0055] 而在提取关键信号后,状态机可以通过网表的结构分析或逻辑分析继而被提取。

[0056] 在本实施例的优选方案中,本步骤之后,还包括:

[0057] S3':用户可以删除部分关键信号及相应的状态机和/或增加更多的关键信号及相应的状态机。

[0058] S4:生成所述状态机的相应的可达性属性;

[0059] 在该步骤中状态机属性随后被自动生成,如图2,状态机属性分为三类:一是状态的可达性,一是状态转换的可达性,最后是无状态死锁。

[0060] 图2是一个仲裁器的例子,四个状态代表了仲裁器里两个关键信号Ack0和Robin的

可能的组合。比如状态10代表Ack0为1,Robin为0。标志为d的箭头代表状态10到状态00的转换,即在状态10和输入为d时,状态机进入00状态。

[0061] 也就是说,在该实例中,图2里的每一个状态都对应一个状态可达和无死锁属性,而每一个转换(一个箭头)都对应一个转换可达属性。

[0062] 另外,如图1所示,在该步骤之后,优选包括一个新覆盖属性判断步骤。

[0063] 具体地,由于步骤S4中的自动生成是一个循环过程,初始的覆盖属性由关键信号状态机生成。可达性属性会生成仿真案例,不可达属性被积累在一个不可达属性库里。用步骤S7生成的仿真案例驱动仿真并收集仿真覆盖率后,覆盖漏洞会被用来再次生成覆盖属性,这些属性首先和不可达属性库对比,如果已经包括在内,则不是新覆盖属性。而新覆盖属性经过形式验证,会进行第二次循环(即步骤S7中的内容)。

[0064] S5:对所述设计、所述约束和所述可达性属性进行形式验证;其中,可达的可达性属性用于生成测试案例,不可达的可达性属性用于纠正覆盖率标准。

[0065] 具体地,在形式验证方法中,当一个可达性属性被证明时(可达性成立)和一个无死锁(一种特殊的可达性)属性被证伪时,验证算法会自动生成一组输入信号序列,这个序列按顺序驱动所述将要被验证设计时,会把该设计从其初始状态带入到可达性属性描述的状态(或者对于无死锁属性,带入到一个状态循环,形成死锁)。这就是一个仿真过程,这组信号也被称为仿真测试案例。

[0066] 也就是说,对于可达性被证明的情况和无状态死锁被证伪的情况都会触发用于见证这些证明的测试案例的自动生成(即包含步骤S4中提及到的状态机属性的三类的任意一种)。

[0067] S6:对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例。

[0068] 其中,经过本步骤优化后的每一个测试案例由于可能有未被赋值输入可以被随机赋值而成为多个测试,这样一个案例因为设计输入的随机选取得以被多次应用,以达到更大的覆盖率。

[0069] S7:对最小化操作后的测试案例进行仿真并计算仿真覆盖率;对仿真出的漏洞和不可达属性重新生成覆盖属性后,采用形式验证进行分析(返回步骤S4)。

[0070] 漏洞和不可达性的逆反即可达性属性,分析即用形式验证方法来证明这些属性。和前面可达性属性验证结果一样,一二类可达性被证明和第三类可达性被证伪,形式验证工具都可以生成测试案例,以达到对这些漏洞的覆盖。

[0071] 实施例2

[0072] 本实施例提供一种仿真测试案例生成系统,该系统的发明构思与实施例1相同,用于实现仿真测试案例的自动生成、覆盖率仿真检测及后续分析。具体地,如图3所示,该系统包括:

[0073] 约束设置模块:用于设置基于设计输入端规范的约束;

[0074] 设计读入模块:用于读入将要被验证的设计;

[0075] 提取模块:用于提取所述设计的关键信号及相应的状态机;

[0076] 可达性属性生成模块:用于生成所述状态机的相应的可达性属性;

[0077] 测试案例生成模块:用于对所述设计、所述约束和所述可达性属性进行形式验证;

其中,可达的可达性属性用于生成测试案例,不可达的可达性属性用于纠正覆盖率标准;

[0078] 测试案例优化模块:对生成测试案例进行最小化操作,包括去掉所述测试案例中与所述可达性属性和所述约束都无关的输入信号和/或去掉重复的测试案例;

[0079] 仿真模块:对最小化操作后的测试案例进行仿真并计算仿真覆盖率;

[0080] 覆盖属性生成与形式验证模块:对仿真出的漏洞和不可达属性重新生成覆盖属性后,采用形式验证进行分析。

[0081] 具体地,如图3所示,图中的设计约束包括的是约束设置模块,图中的形式验证与自动覆盖包括的是设计读入模块、提取模块、可达性属性生成模块、测试案例生成模块和测试案例优化模块,图中的仿真包括的是仿真模块,图中的仿真覆盖包括覆盖属性生成与形式验证模块的对仿真出的漏洞和不可达属性重新生成覆盖属性的部分,而覆盖属性生成与形式验证模块的采用形式验证进行分析属于图中的形式验证与自动覆盖。

[0082] 更优地,在本实施例中,所述的关键信号包括:本身是一个设计中状态机的状态信号、先入先出的控制信号、指令指针、状态存储器和宽度超过一定值的总线中的一种或者多种。

[0083] 更优地,在本实施例中,提取所述设计的关键信号后,相应的状态机通过网表的结构分析和/或逻辑分析继而提取。

[0084] 更优地,在本实施例中,所述的系统还包括:

[0085] 信号与状态机选择模块:用于对提取模块提取到的关键信号及相应的状态机进行处理,包括删除部分关键信号及相应的状态机和/或增加更多的关键信号及相应的状态机。

[0086] 本发明实施例提供的一种仿真测试案例生成系统中相关部分的说明请参见本发明实施例1提供的一种仿真测试案例生成方法中对应部分的详细说明,在此不再赘述。另外,本发明实施例提供的上述技术方案中与现有技术中对应的技术方案实现原理一致的部分并未详细说明,以免过多赘述。

[0087] 实施例3

[0088] 基于实施例1的实现,实施例还提供一种存储介质,其上存储有计算机指令,所述计算机指令运行时执行实施例1中所述的一种仿真测试案例生成方法的步骤。

[0089] 基于这样的理解,本实施例的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(Read-OnlyMemory,ROM)、随机存取存储器(RandomAccessMemory,RAM)、磁碟或者光盘等各种可以存储程序代码的介质。

[0090] 实施例4

[0091] 基于实施例1的实现,本实施例还提供一种终端,包括存储器和处理器,所述存储器上存储有可在所述处理器上运行的计算机指令,所述处理器运行所述计算机指令时执行实施例1所述的一种仿真测试案例生成方法的步骤。

[0092] 在本发明提供的实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。

[0093] 如图4所示,本实施例的其中一种方式为形式验证部分和仿真部分分别在不同处

理器上运行。而在本实施例的另外一种方式为所有部分均在统一处理器上运行。

[0094] 在本发明所提供的所有实施例中,应该理解到,所揭露装置和方法,可以通过其它的方式实现。以上所描述的装置实施例仅仅是示意性的,例如,所述单元/模块的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,又例如,多个单元或模块可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些通信接口,装置或单元的间接耦合或通信连接,可以是电性,机械或其它的形式。

[0095] 显然,上述实施例仅仅是为清楚地说明所作的举例,而并非对实施方式的限定,对于所属领域的普通技术人员来说,在上述说明的基础上还可以做出其他不同形式的变化或变动。这里无需也无法对所有的实施方式予以穷举。而由此所引申出的显而易见的变化或变动仍处于本发明创造的保护范围之内。

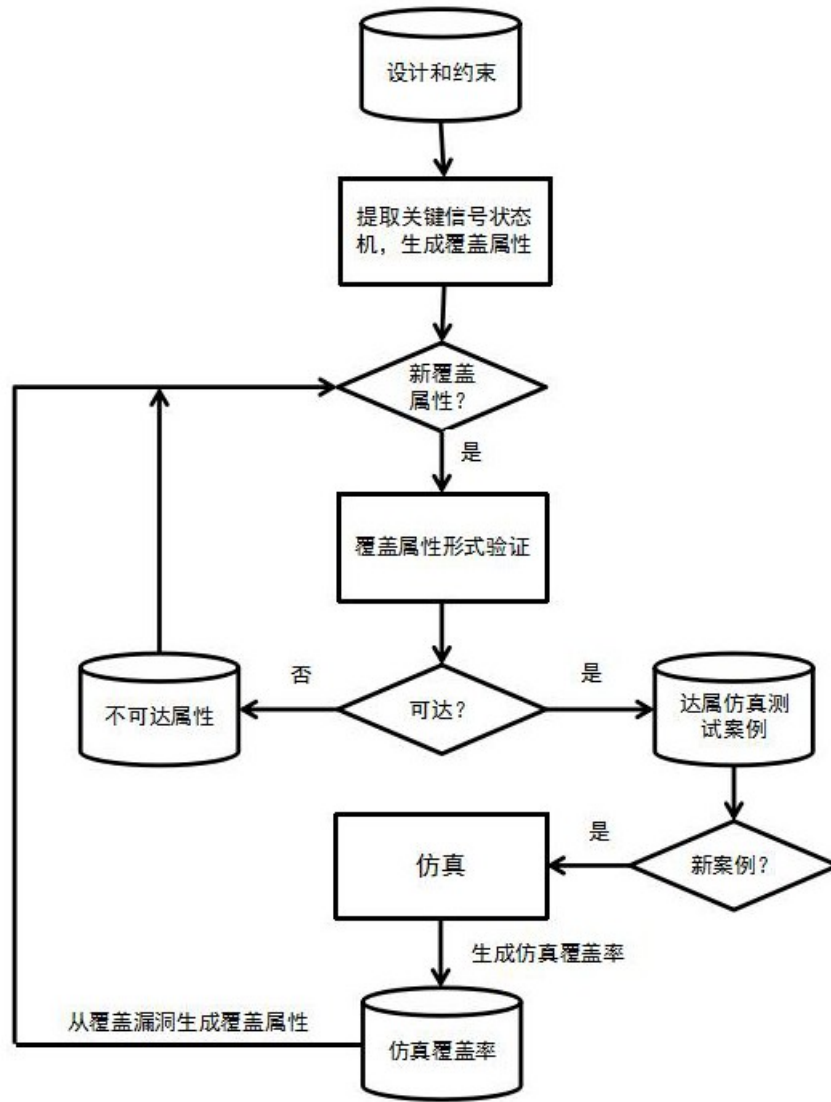


图1

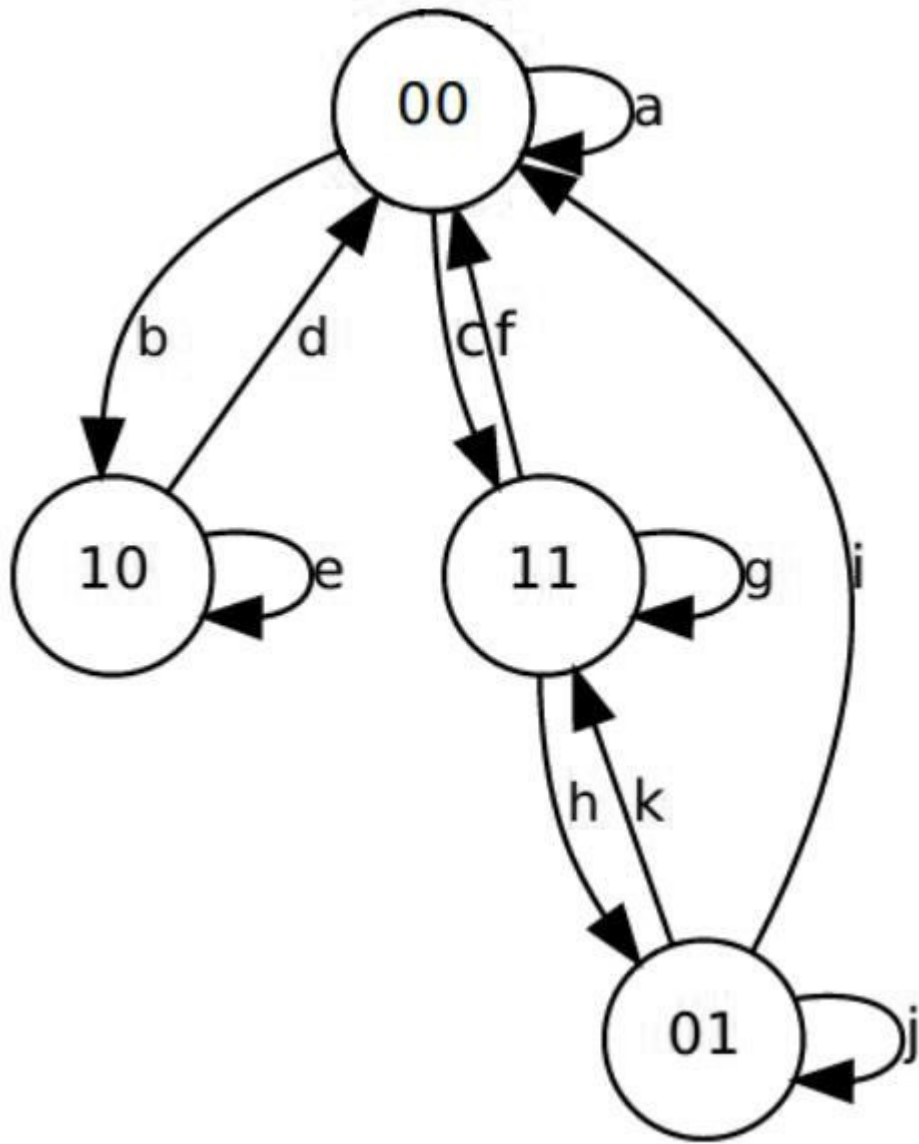


图2

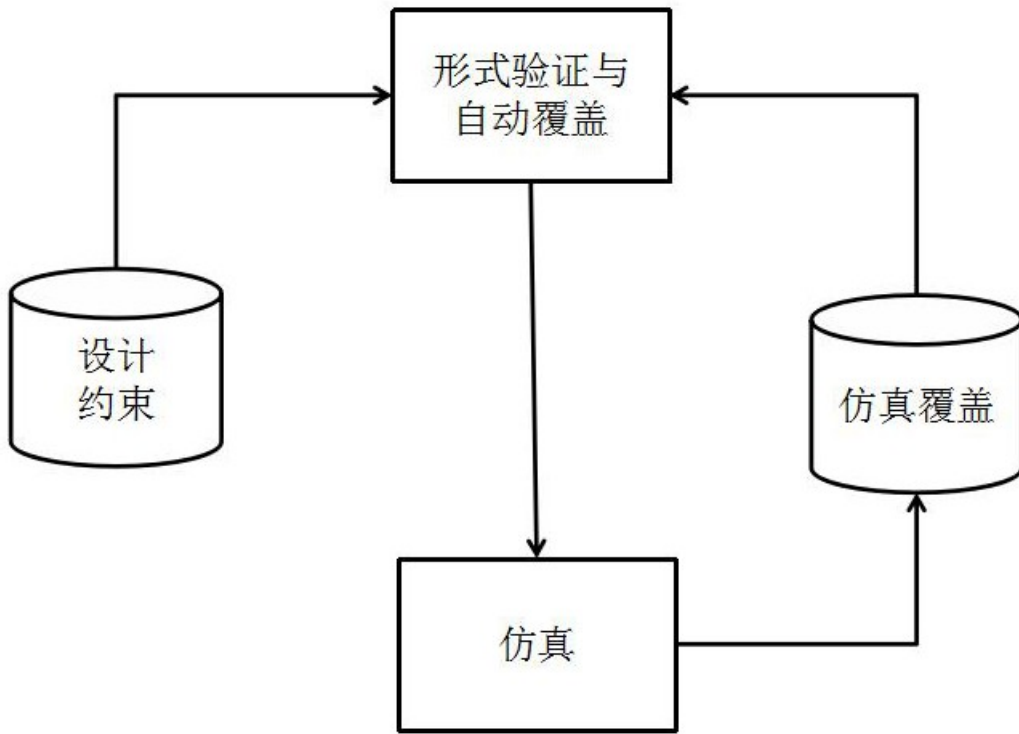


图3

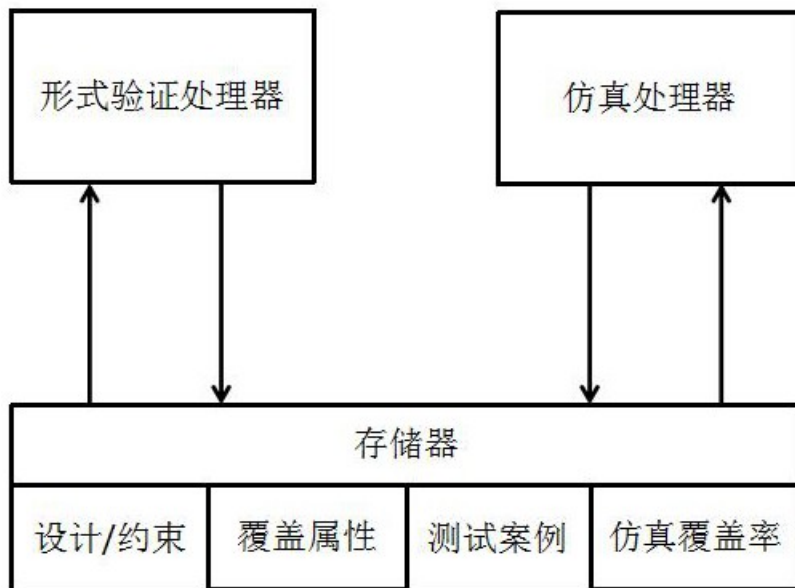


图4