



(12) 发明专利申请

(10) 申请公布号 CN 115510783 A

(43) 申请公布日 2022. 12. 23

(21) 申请号 202211073686.8

(22) 申请日 2022.09.02

(71) 申请人 芯华章科技股份有限公司

地址 211800 江苏省南京市江北新区华创路共享空间01栋18层

(72) 发明人 张玉晨 闫亚飞

(74) 专利代理机构 北京风雅颂专利代理有限公司 11403

专利代理师 李莎

(51) Int. Cl.

G06F 30/3308 (2020.01)

权利要求书2页 说明书8页 附图4页

(54) 发明名称

实现时序逻辑用户定义原语的方法、设备及存储介质

(57) 摘要

本公开提供一种实现时序逻辑用户定义原语的方法、设备及存储介质。该方法包括：接收所述用户定义原语的描述，所述描述至少包括第一输入、第二输入以及与所述第一输入和所述第二输入关联的输出；根据所述描述生成与所述第一和第二输入分别对应的第一和第二真值表，其中，所述第一真值表包括所述第一、第二输入和所述输出在上一时刻的第一、第二在先输入值和在先输出值，以及所述第一输入在当前时刻的第一当前输入值；确定所述第一真值表中与所述第一、第二在先输入值、所述在先输出值和所述第一当前输入值对应的第一元素值；以及基于所述第一元素值，在所述第一真值表中确定所述用户定义原语的所述输出在当前时刻的当前输出值。

元素位置	I1	I2	q	元素值			
				I1+=0	I1+=1	I1+=x	I1+=z
0	x	x	x	39	75	3	3
4	0	0	0				
8	0	0	1				
12	0	0	x				
16	0	1	0				
20	0	1	1				
24	0	1	x				
28	0	x	0				
32	0	x	1				
36	0	x	x				
...							
104	x	x	1				

1. 一种实现时序逻辑用户定义原语 (User-Defined Primitive, UDP) 的方法, 包括:
 - 接收所述用户定义原语的描述, 所述描述至少包括第一输入、第二输入以及与所述第一输入和所述第二输入关联的输出;
 - 根据所述用户定义原语的所述描述生成与所述第一输入和所述第二输入分别对应的第一真值表和第二真值表, 其中, 所述第一真值表包括所述第一输入、所述第二输入和所述输出在上一时刻的第一在先输入值、第二在先输入值和在先输出值, 以及所述第一输入在当前时刻的第一当前输入值;
 - 确定所述第一真值表中与所述第一在先输入值、所述第二在先输入值、所述在先输出值和所述第一当前输入值对应的第一元素值; 以及
 - 基于所述第一元素值, 在所述第一真值表中确定所述用户定义原语的所述输出在当前时刻的当前输出值。
2. 根据权利要求1所述的方法, 其中, 所述第一真值表还包括元素位置, 确定所述第一真值表中与所述第一在先输入值、所述第二在先输入值、所述在先输出值和所述第一当前输入值对应的第一元素值进一步包括:
 - 根据所述第一在先输入值、所述第一当前输入值、所述第二在先输入值和所述在先输出值从所述用户定义原语的所述描述中确定所述当前输出值;
 - 基于所述第一当前输入值、所述第二在先输入值和所述当前输出值在所述第一真值表中确定所述元素位置; 以及
 - 根据所述元素位置和所述当前输出值计算所述第一元素值。
3. 根据权利要求2所述的方法, 其中, 所述元素位置配置为所述第一真值表的位置索引, 并且所述元素位置的值由所述第一当前输入值的状态数确定。
4. 根据权利要求3所述的方法, 其中, 根据所述元素位置和所述当前输出值计算所述第一元素值进一步包括:
 - 使用所述元素位置的值与所述当前输出值的和作为所述第一元素值。
5. 根据权利要求4所述的方法, 其中, 基于所述第一元素值, 在所述第一真值表中确定所述用户定义原语的所述输出在当前时刻的当前输出值进一步包括:
 - 获取所述第一元素值的低两位数据作为所述当前输出值。
6. 根据权利要求4所述的方法, 其中, 基于所述第一元素值, 在所述第一真值表中确定所述用户定义原语的所述输出在当前时刻的当前输出值进一步包括:
 - 获取所述第一元素值的低两位数据之外的高位数据, 所述高位数据指示所述当前输出值在所述第一真值表中的元素位置;
 - 根据所述高位数据在所述第一真值表中确定所述当前输出值所在的行; 以及
 - 从所述行中确定所述当前输出值。
7. 根据权利要求4所述的方法, 其中, 基于所述第一元素值, 在所述第一真值表中确定所述用户定义原语的所述输出在当前时刻的当前输出值进一步包括:
 - 基于所述第一元素值和所述第一当前输入值的状态数进行取模运算; 以及
 - 将所述取模运算的结果作为所述当前输出值。
8. 根据权利要求1至7任一项所述的方法, 其中, 所述第二真值表包括所述第一输入、所述第二输入和所述输出在上一时刻的第一在先输入值、第二在先输入值和在先输出值, 以

及所述第二输入在当前时刻的第二当前输入值；

所述方法进一步包括：

确定所述第二真值表中与所述第一在先输入值、所述第二在先输入值、所述在先输出值和所述第二当前输入值对应的第二元素值；

基于所述第一元素值，在所述第一真值表中确定所述用户定义原语的所述在先输出值；以及

基于所述第二元素值，在所述第二真值表中确定所述用户定义原语的所述当前输出值。

9. 一种实现时序逻辑用户定义原语的电子设备，包括：

存储器，用于存储一组指令；以及

至少一个处理器，配置为执行所述一组指令以使得所述电子设备执行如权利要求1至8任一项所述的方法。

10. 一种非暂态计算机可读存储介质，所述非暂态计算机可读存储介质存储电子设备的一组指令，所述一组指令在被执行时使得所述电子设备执行权利要求1至8任一项所述方法。

实现时序逻辑用户定义原语的方法、设备及存储介质

技术领域

[0001] 本公开涉及计算机软件领域,尤其涉及一种实现时序逻辑用户定义原语的方法、设备及存储介质。

背景技术

[0002] Verilog语言中内置了多种基础电路元件,例如,AND、OR、XOR等。但是在实际使用时,用户有时候需要构建特定功能的模块,这些模块如果仅用内置的这些基础元件去描述可能会很复杂。所以,Verilog还为用户提供了自己编写原语的能力,这种原语就是用户定义原语(User-Defined Primitive,简称UDP)。

[0003] UDP包括组合逻辑UDP和时序逻辑UDP。其中,时序逻辑UDP的当前输出值不但取决于当前输入值,还取决于当前的内部状态(即,上一个输出值)。

[0004] UDP的实现通常是通过查找真值表的方式来获取输出。以边沿触发的时序逻辑UDP为例,其输入的可能状态包括9种(00,01,0x,10,11,1x,x0,x1,xx,即,从0到0,从0到1,从0到x等)。此时,对于一个输入为N的边沿触发的时序逻辑UDP,它的真值表中元素的数量是 $9^N * 3$,其中,3表示上一个输出值可能有三种状态(0,1,x)。传统的时序逻辑UDP的真值表方案会造成存储空间的巨大开销。

发明内容

[0005] 有鉴于此,本公开提出了一种实现时序逻辑用户定义原语的方法、设备及存储介质。

[0006] 本公开第一方面,提供了一种实现时序逻辑用户定义原语的方法,包括:接收所述用户定义原语的描述,所述描述至少包括第一输入、第二输入以及与所述第一输入和所述第二输入关联的输出;根据所述用户定义原语的所述描述生成与所述第一输入和所述第二输入分别对应的第一真值表和第二真值表,其中,所述第一真值表包括所述第一输入、所述第二输入和所述输出在上一时刻的第一在先输入值、第二在先输入值和在先输出值,以及所述第一输入在当前时刻的第一当前输入值;确定所述第一真值表中与所述第一在先输入值、所述第二在先输入值、所述在先输出值和所述第一当前输入值对应的第一元素值;以及基于所述第一元素值,在所述第一真值表中确定所述用户定义原语的所述输出在当前时刻的当前输出值。

[0007] 本公开第二方面,提供了一种实现时序逻辑用户定义原语的电子设备,包括:存储器,用于存储一组指令;以及至少一个处理器,配置为执行该组指令以使得所述电子设备执行如第一方面所述的方法。

[0008] 本公开第三方面,提供了一种非暂态计算机可读存储介质,所述非暂态计算机可读存储介质存储电子设备的一组指令,该组指令用于使所述电子设备执行如第一方面所述的方法。

[0009] 本公开提供的一种实现时序逻辑用户定义原语的方法、设备及存储介质,通过对

UDP的每一个输入分别生成对应的真值表,有效减少了真值表中元素的数量,进而降低了真值表对存储空间的占用。在仿真运行过程中,仿真工具可以更快地读取存储的真值表,提高UDP的实现效率。

附图说明

[0010] 为了更清楚地说明本公开或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本公开,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0011] 图1A示出了根据本公开实施例的示例性电子设备的结构示意图。

[0012] 图1B示出了根据本公开实施例的示例性仿真工具的示意图。

[0013] 图2A示出了根据本公开实施例的示例性时序逻辑用户定义原语的描述的示意图。

[0014] 图2B示出了根据本公开实施例的示例性传统真值表的示意图。

[0015] 图2C示出了根据本公开实施例的与第一输入对应的示例性第一真值表的示意图。

[0016] 图3示出了根据本公开实施例的示例性实现时序逻辑用户定义原语的方法的流程图。

具体实施方式

[0017] 为使本公开的目的、技术方案和优点更加清楚明白,以下结合具体实施例,并参照附图,对本公开进一步详细说明。

[0018] 需要说明的是,除非另外定义,本公开使用的技术术语或者科学术语应当为本公开所属领域内具有一般技能的人士所理解的通常意义。本公开中使用的“第一”、“第二”以及类似的词语并不表示任何顺序、数量或者重要性,而只是用来区分不同的组成部分。“包括”或者“包含”等类似的词语意指出现该词前面的元件或者物件涵盖出现在该词后面列举的元件或者物件及其等同,而不排除其他元件或者物件。“连接”或者“相连”等类似的词语并非限定于物理的或者机械的连接,而是可以包括电性的连接,不管是直接的还是间接的。

[0019] 如上所述,传统的时序逻辑用户定义原语(UDP)的真值表中元素的数量达到 $9^N \times 3$,元素数量越多,真值表占据的存储空间也就越大。因此,传统的时序逻辑UDP的真值表方案会造成存储空间的巨大开销。

[0020] 有鉴于上述问题,本公开提供一种实现时序逻辑用户定义原语的方法、设备及存储介质,通过对每一个输入分别生成对应的真值表,有效减少了真值表中元素的数量,进而降低了真值表对存储空间的占用。

[0021] 图1A示出了根据本公开实施例的示例性电子设备100的结构示意图。

[0022] 电子设备100例如可以是主机计算机。电子设备100可以包括:处理器102、存储器104、网络接口106、外围接口108和总线110。其中处理器102、存储器104、网络接口106和外围接口108可以通过总线110实现彼此之间在电子设备100内部的通信连接。

[0023] 处理器102可以是中央处理器(Central Processing Unit,CPU)、图像处理器、神经网络处理器、微控制器、可编程逻辑器件、数字信号处理器、应用专用集成电路(Application Specific Integrated Circuit,ASIC)、或一个或多个集成电路。处理器102

可以用于进行与本公开描述的技术相关的功能。在一些实施例中,处理器102还可以包括集成为单一逻辑组件的多个处理器。如图1A所示,处理器102可以包括多个处理器102a、102b和102c。

[0024] 存储器104可以配置为存储数据(例如,指令集、计算机代码、中间数据、真值表等)。例如,如图1A所示,存储的数据可以包括程序指令(例如,用于实现本公开的技术的程序指令)以及真值表。处理器102也可以访问存储的程序指令和真值表,并且执行程序指令以生成或读取真值表。存储器104可以包括非暂态计算机可读存储介质,例如易失性存储装置或非易失性存储装置。在一些实施例中,存储器104可以包括随机访问存储器(RAM)、只读存储器(ROM)、光盘、磁盘、硬盘、固态硬盘(SSD)、闪存、存储棒等。

[0025] 网络接口106可以配置为使电子设备100能够经由网络与一个或多个其他外部装置进行通信。该网络可以是能够传输和/或接收数据的任何有线或无线的网络。例如,该网络可以是有线网络、本地无线网络(例如,蓝牙、WiFi、近场通信(NFC)等)、蜂窝网络、因特网、或上述的组合。可以理解的是,网络的类型不限于上述具体示例。在一些实施例中,网络接口106可以包括任意数量的网络接口控制器(NIC)、射频模块、接收发射器、调制解调器、路由器、网关、适配器、蜂窝网络芯片等的任意组合。

[0026] 外围接口108可以配置为将电子设备100与一个或多个外围装置连接,以实现信息输入及输出。例如,外围装置可以包括键盘、鼠标、触摸板、触摸屏、麦克风、各类传感器等输入装置以及显示器、扬声器、振动器、指示灯等输出装置。

[0027] 总线110可以被配置为在电子设备100的各个组件(例如,处理器102、存储器104、网络接口106和外围接口108)之间传输信息,并且例如可以是内部总线(例如,处理器-存储器总线)、外部总线(USB端口、PCI-E总线)等。

[0028] 在一些实施例中,除了图1A所示出的以及上述所描述的处理器102、存储器104、网络接口106、外围接口108和总线110之外,电子设备100还可以包括实现正常运行所需的一个或多个其他组件和/或实现本公开的实施例的解决方案所必需的一个或多个其他组件。在一些实施例中,电子设备100可能不包含图1A中所示的一个或多个组件。

[0029] 需要说明的是,尽管上述电子设备100的构成架构仅示出了处理器102、存储器104、网络接口106、外围接口108和总线110,但是在具体实施过程中,该电子设备100的构成架构还可以包括实现正常运行所必需的其他组件。此外,本领域的技术人员可以理解的是,上述电子设备100的构成架构中也可以仅包含实现本公开实施例方案所必需的组件,而不必包含图中所示的全部组件。

[0030] 图1B示出了根据本公开实施例的示例性仿真工具120的示意图。仿真工具120可以是运行在电子设备100上的计算机程序。

[0031] 在芯片设计领域,通常可以利用仿真工具对一个设计进行仿真。仿真工具例如可以是芯华章科技股份有限公司出品的GalaxSim仿真工具。图1B示出的示例性仿真工具120可以包括编译器122和仿真器124。编译器122可以接收源代码121(例如,VHDL、Verilog、SystemVerilog等硬件描述语言),并编译为执行代码123(例如,机器码、汇编代码、软件代码、真值表等)。仿真器124可以根据执行代码123进行仿真,并将仿真结果125输出。例如,仿真工具120可以将仿真结果125经由图1A的外围接口108输出到输出设备上(例如,显示在显示器上)。

[0032] 在一些实施例中,当需要在Verilog语言所编写的源代码121中构建特定功能的模块时,用户可以自行编写UDP以满足功能需要。

[0033] 为便于描述,本公开实施例以一个二输入的时序逻辑UDP作为示例进行描述。

[0034] 图2A示出了根据本公开实施例的示例性时序逻辑用户定义原语的描述200的示意图。描述200可以包括UDP状态表。UDP状态表可以由用户定义,代表根据输入状态和UDP的当前状态得到的UDP输出值。

[0035] 如图2A所示,描述200至少可以包括UDP的第一输入 i_1 、第二输入 i_2 、上一个时刻的输出值 q 、以及与第一输入 i_1 、第二输入 i_2 、上一时刻输出值 q 相关联的当前时刻的输出 $q+$ 。可以理解的是,描述200可以包括多于2个的输入。从描述200中可以得到,当第一输入 i_1 是从上一时刻的0跳变到当前时刻的1(即,第一输入 i_1 是上升沿),第二输入 i_2 是0,且上一时刻的输出值 q 是1时,UDP当前时刻的输出 $q+$ 是0;当第一输入 i_1 是上升沿,第二输入 i_2 是1时,不管上一时刻的输出值 q 是多少,UDP当前时刻的输出 $q+$ 是1。

[0036] UDP的实现通常是通过查找真值表的方式来获得输出的。

[0037] 图2B示出了根据本公开实施例的示例性传统真值表210的示意图。对于一个二输入的时序逻辑UDP来说(即输入 $N=2$),传统的真值表方案中元素的数量是243。

[0038] 在一些实施例中,仿真工具120可以基于如图2A所示的描述200分别生成与第一输入 i_1 对应的第一真值表和与第二输入 i_2 对应的第二真值表。

[0039] 图2C示出了根据本公开实施例的与第一输入 i_1 对应的示例性第一真值表220的示意图。

[0040] 如图2C所示,第一真值表220中可以包括UDP的第一输入 i_1 在上一时刻的在先输入值 I_1 ,第二输入 i_2 在上一时刻的在先输入值 I_2 ,UDP在上一时刻的在先输出值 q ,以及第一输入 i_1 在当前时刻的当前输入值 I_1+ 。其中,在先输入值 I_1 、 I_2 和在先输出值 q 均可以为(0,1,x),所以第一真值表220有 $3^3=27$ 种可能的状态,即,第一真值表220可以是27行。UDP的当前输入值 I_1+ 可以为(0,1,x,z)。仿真工具120可以根据图2A所示的描述200得到第一真值表220中与在先输入值 I_1 、 I_2 、在先输出 q 、当前输入 I_1+ 对应的元素值。

[0041] 以 $I_1=x$, $I_2=x$, $q=x$, $I_1+=0$ 为例,此时,第一输入 i_1 为(x0),第二输入 i_2 为x,在先输出 q 为x。回到如图2A所示的描述200,这种状态未被定义,所以当前输出 $q+$ 为x(即,未知)。用户可以定义输出x的值为3。因此,此时UDP的当前输出值 $q+$ 为 $x=3$ 。

[0042] 为了用户更方便的查找输入变化后UDP的输出值,第一真值表220还可以包括元素位置(如第一真值表220的第一列所示)。元素位置从0开始标识。在一些实施例中,由于第一输入 i_1 在当前时刻的当前输入值 I_1+ 可以为(0,1,x,z)这四种状态,所以每一行的元素位置的值都在上一行元素位置的值的基础上加4。第一真值表220中元素位置的值可以为0,4,8,⋯,104,共计27个值。

[0043] 仿真工具120可以使用 I_1+ 替换 I_1 , $q+$ 替换 q ,在第一真值表220中找到 $I_1=0$, $I_2=x$, $q=x$ 处于元素位置为36的行。基于该元素位置的值36和前述得到的当前输出值3,仿真工具120可以计算得到与 $I_1=x$, $I_2=x$, $q=x$, $I_1+=0$ 对应的元素值。在一些实施例中,仿真工具120可以使用元素位置的值36与当前输出值3的和作为该元素值。如图2C所示,与 $I_1=x$, $I_2=x$, $q=x$, $I_1+=0$ 对应的元素值为 $36+3=39$ 。

[0044] 类似地,对于 $I_1=x$, $I_2=x$, $q=x$, $I_1+=1$,仿真工具120可以重复前述步骤,从描述

200中得到当前输出 $q+$ 为 x (即,3);在第一真值表220中找到 $I1=1, I2=x, q=x$ 处于元素位置为72的行。所以,与 $I1=x, I2=x, q=x, I1+=1$ 对应的元素值为 $72+3=75$ 。依此类推,直至仿真工具120得到第一真值表220中所有的元素值。

[0045] 同样地,仿真工具120可以使用生成第一真值表220的方法来生成与第二输入 $i2$ 对应的第二真值表,在此不再赘述。

[0046] 经过前述步骤,仿真工具120已经分别生成了与第一输入 $i1$ 对应的第一真值表220和与第二输入 $i2$ 对应的第二真值表。在仿真过程中,当UDP接收到新的第一输入或第二输入时,仿真工具120就可以直接通过查找已生成的第一真值表220或第二真值表来获得UDP的当前输出值。

[0047] 以UDP接收到新的第一输入 $i1$ 为例,例如, $I1=x, I2=x, q=x, I1+=0$,此时,第一输入 $i1$ 由在先输入 x 变为当前输入0。

[0048] 在一些实施例中,仿真工具120可以在第一真值表220中找到相应的元素值39。元素值39在计算机中以二进制数据(100111)进行存储。基于该元素值39,仿真工具120可以获取元素值39在计算机中存储的低两位数据(11),得到UDP的当前输出值3。

[0049] 在另一些实施例中,仿真工具120可以获取元素39在计算机中存储的低两位数据之外的高位数据(100100),该高位数据(100100)可以指示当前输出值 $q+$ 在第一真值表220中的位置36。仿真工具120根据该高位数据的值就可以在第一真值表220中确定当前输出在元素位置为36的行。仿真工具120可以从确定的行中读取输出(即, q 列)对应的值 x (即,3)为当前输出值。

[0050] 在又一些实施例中,仿真工具120可以使用元素值39对第一输入 $i1$ 的当前输入值 $I1+$ 的状态数4执行取模运算。取模运算所得到的结果3就是此时UDP的当前输出值。

[0051] 在一些实施例中,UDP还可以同时接收到新的第一输入 $i1$ 和新的第二输入 $i2$ 。此时,仿真工具120可以先在第一真值表220中处理新的第一输入 $i1$,将从第一真值表120中查找得到的输出作为在先输出 q ;再利用该新的第一输入 $i1$ 、在先输出 q 在第二真值表中处理新的第二输入 $i2$,从第二真值表中得到UDP的当前输出值 $q+$ 。这样,仿真工具120通过在两个真值表中一次分别处理变化的输入值,得到最终UDP的当前输出值。

[0052] 采用本公开实施例所使用的方法,每个真值表中元素值的数量为 $3^{N+1} * 4$ 。对于一个二输入的时序逻辑UDP来说(即输入 $N=2$),每个真值表元素值的数量为108,两个真值表的元素值的总数量为216。可以看到,元素值的总量小于如图2B所示的传统真值表中的元素数量243。

[0053] 在一些实施例中,UDP的输入 z 是和输入 x 作一样的处理的,因此当前输入值 $I1+=x$ 和 $I1+=z$ 可以合并为同一列处理。在这种情况下,第一真值表220的元素值的数量可以进一步减少为81。相应地,两个真值表的元素值的总数量为162,进一步减少了真值表中的元素值的数量。

[0054] 可以理解的是,上述实施例是以一个二输入的时序逻辑UDP为例进行说明的。在实际应用中,时序逻辑UDP至少可以支持9个输入。这种情况下,传统真值表方案中的元素数量高达 10^{10} 量级,而本公开实施例所使用的方法可以将元素值的数量降低到 10^6 量级。随着输入数量的增多,本公开实施例所提供的方法在存储空间方面的优势愈发明显。

[0055] 本公开实施例所提供的方法通过对UDP的每一个输入分别生成对应的真值表,有

效减少了真值表中元素的数量,进而降低了真值表对存储空间的占用。在仿真运行过程中,仿真工具120可以更快地读取存储的真值表,提高UDP的实现效率。

[0056] 图3示出了根据本公开实施例的示例性实现时序逻辑用户定义原语(UDP)的方法300的流程图。方法300可以由例如图1A的电子设备100来执行,更具体地,可以由在电子设备100上运行的仿真工具120执行。方法300可以包括以下步骤。

[0057] 在步骤S302,仿真工具120可以接收用户定义原语(UDP)的描述(例如,图2A所示的描述200)。该描述至少包括第一输入(例如,图2A所示的第一输入*i*₁)、第二输入(例如,图2A所示的第二输入*i*₂)以及与该第一输入和所述第二输入关联的输出(例如,图2A所示的上一时刻输出值*q*和当前时刻的输出值*q*⁺)。

[0058] 在一些实施例中,该UDP的描述可以包括UDP状态表。

[0059] 在步骤S304,仿真工具120可以根据该UDP的描述生成与该第一输入和该第二输入分别对应的第一真值表(例如,图2C所示的第一真值表220)和第二真值表。其中,该第一真值表包括该第一输入、该第二输入和该输出在上一时刻的第一在先输入值(例如,图2C所示的在先输入值*I*₁)、第二在先输入值(例如,图2C所示的在先输入值*I*₂)和在先输出值(例如,图2C所示的在先输出值*q*),以及该第一输入在当前时刻的第一当前输入值(例如,图2C所示的当前输入值*I*₁⁺)。

[0060] 在步骤S306,仿真工具120可以确定该第一真值表(例如,图2C所示的第一真值表220)中与该第一在先输入值(例如,图2C所示的在先输入值*I*₁)、该第二在先输入值(例如,图2C所示的在先输入值*I*₂)、该在先输出值(例如,图2C所示的在先输出值*q*)和该第一当前输入值(例如,图2C所示的当前输入值*I*₁⁺)对应的第一元素值。

[0061] 在一些实施例中,该第一真值表还包括元素位置(例如,图2C所示的第一真值表220的第一列“元素位置”)。在一些实施例中,该元素位置可以配置为该第一真值表的位置索引,并且该元素位置的值由该第一当前输入值的状态数(例如,图2C所示的第一输入*i*₁的当前输入值*I*₁⁺的四种状态,状态数为4)确定。例如,第一真值表220中元素位置的值可以为0,4,8,⋯,104。

[0062] 在一些实施例中,仿真工具120可以根据该第一在先输入值(例如,*I*₁=*x*)、该第一当前输入值(例如,*I*₁⁺=0)、该第二在先输入值(例如,*I*₂=*x*)和该在先输出值(例如,*q*=*x*)从该UDP的描述(例如,图2A所示的描述200)中确定该UDP的当前输出值(例如,*q*⁺=*x*=3)。仿真工具120根据该第一当前输入值(例如,*I*₁⁺=0)、该第二在先输入值(例如,*I*₂=*x*)和该当前输出值(例如,*q*⁺=*x*)在该第一真值表中可以确定该元素位置(例如,元素位置36)。仿真工具120可以根据该元素位置和该当前输出值计算该第一元素值。

[0063] 在一些实施例中,仿真工具120可以使用该元素位置的值(例如:元素位置的值36)与该当前输出值(例如,*q*⁺=3)的和作为该第一元素值(例如,36+3=39)。

[0064] 在一些实施例中,仿真工具120可以使用生成第一真值表的方法来生成与第二输入对应的第二真值表,以确定该第二真值表中的第二元素值。

[0065] 在步骤S308,仿真工具120可以基于该第一元素值(例如,39),在该第一真值表中确定该UDP的输出在当前时刻的当前输出值。

[0066] 在一些实施例中,仿真工具120可以获取该第一元素值的低两位数据(例如,39的低两位数据11)作为该当前输出值(例如,3)。

[0067] 在另一些实施例中,仿真工具120可以获取该第一元素值的低两位数据之外的高位数据(例如,39的高位数据100100)。该高位数据可以指示该当前输出值在该第一真值表中的元素位置(例如,元素位置36)。仿真工具120可以根据该元素位置在该第一真值表中确定该当前输出值所在的行(例如,元素位置36所指示的行)。仿真工具120可以从该行中确定该当前输出值(例如,该行的q列的值 $x=3$)。

[0068] 在又一些实施例中,仿真工具120可以基于该第一元素值(例如,39)和该第一当前输入值的状态数(例如,状态数4)进行取模运算。仿真工具120可以将该取模运算的结果作为该当前输出值(例如,运算结果3)。

[0069] 在一些实施例中,UDP可以同时接收到新的第一输入和新的第二输入。仿真工具120可以基于第一元素值,在第一真值表中确定UDP的在先输出值。然后,仿真工具120可以继续基于第二元素值,在第二真值表中确定所UDP的当前输出值。

[0070] 本公开还提供了一种实现时序逻辑用户定义原语的电子设备。该电子设备可以是图1A所示的电子设备100。电子设备100可以被配置为执行存储在存储器104中的计算机程序以实现与本公开一致的实现时序逻辑用户定义原语的方法,诸如上述示例性方法(例如,图3中所示的方法300)。此处不再赘述。

[0071] 本公开还提供了一种非暂态计算机可读存储介质。非暂态计算机可读存储介质存储电子设备100的一组指令。当该组指令被执行时,使得电子设备100实现与本公开一致的实现时序逻辑用户定义原语的方法,诸如上述示例性方法(例如,图3中所示的方法300)。此处不再赘述。

[0072] 本实施例的计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质,可用于存储可以被计算设备访问的信息。

[0073] 上述对本公开特定实施例进行了描述。其他实施例在所附权利要求书的范围内。在一些情况下,在权利要求书中记载的动作或步骤可以按照不同于实施例中的顺序来执行并且仍然可以实现期望的结果。另外,在附图中描绘的过程不一定要求示出的特定顺序或者连续顺序才能实现期望的结果。在某些实施方式中,多任务处理和并行处理也是可以的或者可能是有利的。

[0074] 所属领域的普通技术人员应当理解:以上任何实施例的讨论仅为示例性的,并非旨在暗示本公开的范围(包括权利要求)被限于这些例子;在本公开的思路下,以上实施例或者不同实施例中的技术特征之间也可以进行组合,步骤可以以任意顺序实现,并存在如上所述的本公开的不同方面的许多其它变化,为了简明它们没有在细节中提供。

[0075] 另外,为简化说明和讨论,并且为了不会使本公开难以理解,在所提供的附图中可以示出或不示出与集成电路(IC)芯片和其它部件的公知的电源/接地连接。此外,可以以框图的形式示出装置,以便避免使本公开难以理解,并且这也考虑了以下事实,即关于这些框图装置的实施方式的细节是高度取决于将要实施本公开的平台(即,这些细节应当

完全处于本领域技术人员的理解范围内)。在阐述了具体细节(例如,电路)以描述本公开的示例性实施例的情况下,对本领域技术人员来说显而易见的是,可以在没有这些具体细节的情况下或者这些具体细节有变化的情况下实施本公开。因此,这些描述应被认为是说明性的而不是限制性的。

[0076] 尽管已经结合了本公开的具体实施例对本公开进行了描述,但是根据前面的描述,这些实施例的很多替换、修改和变型对本领域普通技术人员来说将是显而易见的。例如,其它存储器架构(例如,动态RAM(DRAM))可以使用所讨论的实施例。

[0077] 本公开旨在涵盖落入所附权利要求的宽泛范围之内的所有这样的替换、修改和变型。因此,凡在本公开的精神和原则之内,所做的任何省略、修改、等同替换、改进等,均应包含在本公开的保护范围之内。

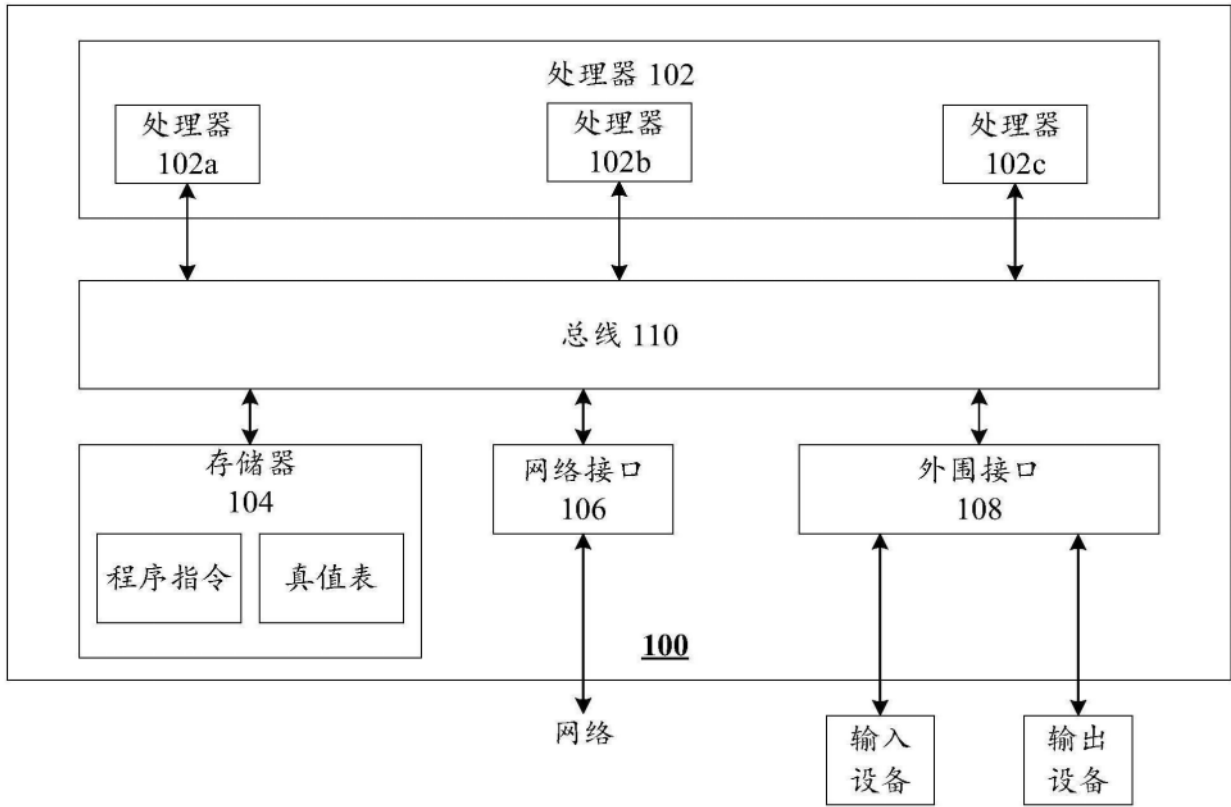


图1A

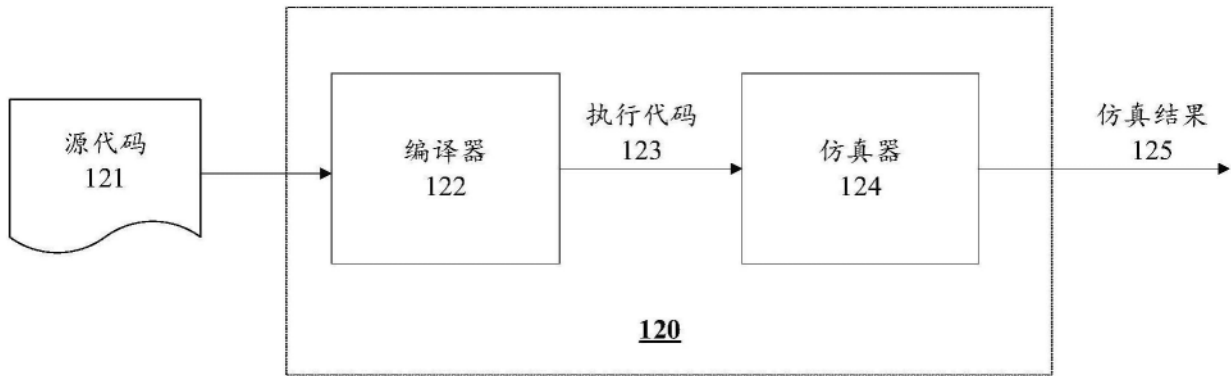


图1B

$i1 \quad i2: q : q^+$
 (01) 0: 1 : 0;
 (01) 1: ? : 1;

200

图2A

i1	i2	q	q⁺
00	00	0	0
00	00	1	1
00	00	x	
00	00	0	
00	00	1	
00	00	x	
00	00	0	
00	00	1	
00	00	x	
00	01	0	
...			
xx	xx	x	

} 9^N*3

210

图2B

元素位置	I1	I2	q	元素值			
				I1+=0	I1+=1	I1+=x	I1+=z
0	x	x	x	39	75	3	3
4	0	0	0				
8	0	0	1				
12	0	0	x				
16	0	1	0				
20	0	1	1				
24	0	1	x				
28	0	x	0				
32	0	x	1				
36	0	x	x				
...							
104	x	x	1				

} 3^{N+1*4}

220

图2C

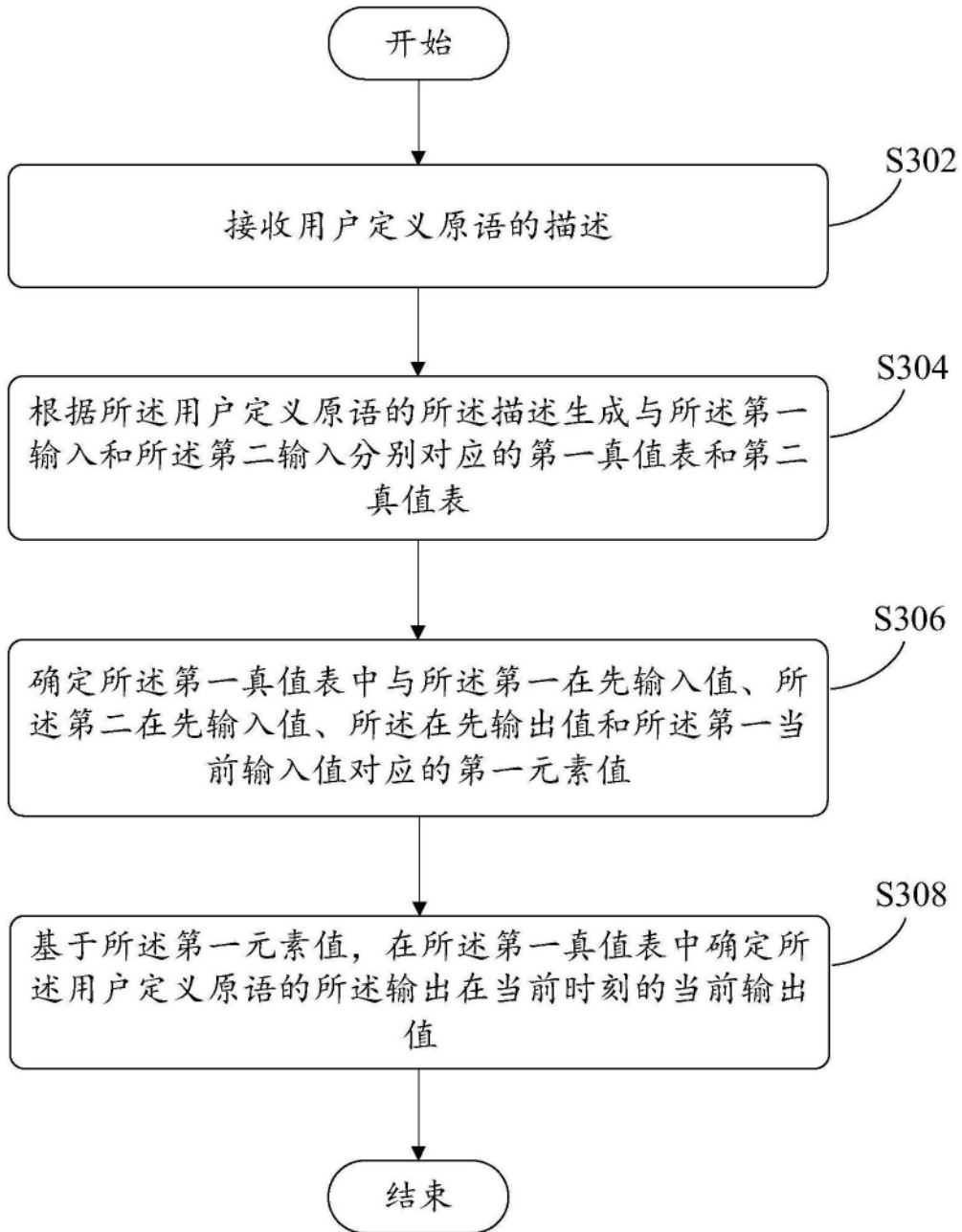


图3