(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 108604276 B
(45)授权公告日 2022.04.29

(21)申请号 201680080848.7
(22)申请日 2016.12.02
(65)同一申请的已公布的文献号
申请公布号 CN 108604276 A
(43)申请公布日 2018.09.28
(30)优先权数据
1521394.5 2015.12.03 GB
(85)PCT国际申请进入国家阶段日
2018.08.02
(86)PCT国际申请的申请数据
PCT/EP2016/079667 2016.12.02
(87)PCT国际申请的公布数据
WO2017/093533 EN 2017.06.08
(73)专利权人 格里森技术有限责任公司
地址 英国伦敦
(72)发明人 H•哈里森
(74)专利代理机构 隆天知识产权代理有限公司 72003
代理人 石海霞 李玉锁
(51)Int.Cl.
G06F 21/57(2013.01)
G06F 9/4401(2018.01)
(56)对比文件
US 2011162077 A1,2011.06.30
US 2011162077 A1,2011.06.30
CN 102081534 A,2011.06.01
US 7007077 B1,2006.02.28
CN 102971742 A,2013.03.13
CN 103914658 A,2014.07.09
CN 103870745 A,2014.06.18
CN 1506813 A,2004.06.23
US 2005172280 A1,2005.08.04
US 7664984 B2,2010.02.16
CN 101361077 A,2009.02.04
CN 102693379 A,2012.09.26
US 8812830 B2,2014.08.19
US 2014089650 A1,2014.03.27
审查员 彭玢

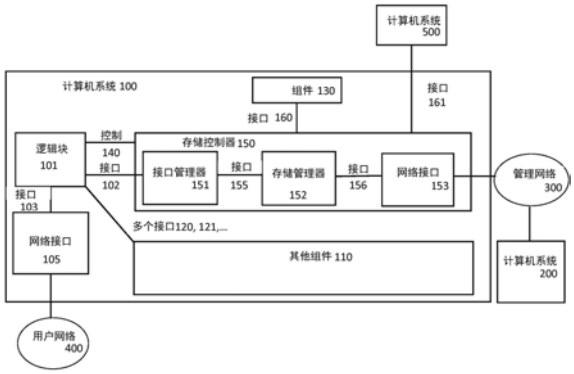
权利要求书3页 说明书15页 附图7页

(54)发明名称

可通过网络更新的安全启动代码缓存

(57)摘要

提供了一种安全启动计算机系统。系统包括逻辑块(101),包括执行指令的一个或多个处理单元(101a、101b),逻辑块被配置为根据第一通信协议在开启或重置逻辑块时通过第一接口(102)请求启动指令。控制器组件(150)被配置为根据第一通信协议通过第一接口与逻辑块通信,控制器还被配置为实现到第二计算机系统(200)的通信链路(300),并且被配置为从第二计算机系统接收启动指令。逻辑块被预先配置为根据第一通信协议以不能被由逻辑块执行的指令更改的方式通过第一接口与控制器通信。控制器被配置为阻止完成任何来自所述逻辑块的写入请求。系统在逻辑块与控制器之间还包括控制连接(140)。控制器还被配置为使用控制连接来开启或重置逻辑块,以便将逻辑块置于执行启动指令而产生的预定活动状态,使得在开启或重置逻辑块之前在逻辑块上操作的软件不能影响预定活动状态。还提供了相应的方法和第二计算机系统(200)。



CN 108604276 B

1.一种安全启动计算机系统,包括逻辑块和控制器,其中:

所述逻辑块包括执行指令的一个或多个处理单元,所述逻辑块被预配置为根据第一通信协议在开启或重置所述逻辑块时通过第一接口与所述控制器通信以通过所述第一接口请求启动指令,其中所述逻辑块的预先配置不能被所述逻辑块执行的软件指令改变;

所述控制器包括接口管理器和存储管理器,所述控制器被配置为实现到第二计算机系统的通信链路,并且被配置为通过所述通信链路从所述第二计算机系统接收启动指令;

所述接口管理器被配置为:根据所述第一通信协议通过所述第一接口与所述逻辑块通信,通过所述第一接口识别来自所述逻辑块的对数据的请求,并且通过第二接口将所述请求传递到所述存储管理器;

所述第二接口始终是只读的,使得除了指定应该经由所述接口管理器从所述存储管理器读取的数据这一目的之外,数据不能从所述逻辑块流到所述存储管理器;

所述存储管理器被配置为通过第三接口接收所述控制器通过所述通信链路从所述第二计算机系统接收到的启动指令,所述第三接口是与所述第二接口完全独立的,使得所述启动指令能够通过所述第三接口被改变,但是不能通过所述第二接口被改变;

所述系统还包括在所述逻辑块与所述控制器之间的控制连接,所述控制器被配置为使用所述控制连接来开启或重置所述逻辑块,以便将所述逻辑块置于执行所述启动指令而产生的预定活动状态,使得在开启或重置所述逻辑块之前在所述逻辑块上操作的软件不能影响所述预定活动状态。

2.根据权利要求1所述的安全启动计算机系统,其中,所述逻辑块使用如下器件或指令而被预先配置:

• 一个或多个专用集成电路和/或一个或多个可编程逻辑器件;或者

• 在只读存储器上存储的指令。

3.根据权利要求1或2所述的安全启动计算机系统,其中,所述控制器包括用于存储所述启动指令的存储器,由此所述启动指令可以从所述第二计算机系统被接收,并且在开启或重置所述安全启动计算机系统之前被存储。

4.根据权利要求1或2所述的安全启动计算机系统,其中,所述控制器被配置为,在接收到来自所述逻辑块的对所述启动指令的请求时,响应于来自所述逻辑块的对所述启动指令的请求,从所述第二计算机系统请求所述启动指令。

5.根据权利要求1或2所述的安全启动计算机系统,其中,重置所述逻辑块包括如下中的一个或多个:

• 关闭,然后开启所述逻辑块;以及

• 执行硬件重置。

6.根据权利要求1或2所述的安全启动计算机系统,包括连接到所述逻辑块的RAM,其中,所述控制器还被配置为使用所述控制连接来控制连接到所述逻辑块的RAM的电力状态。

7.根据权利要求1或2所述的安全启动计算机系统,其中,所述控制器被配置为响应于如下通信而使用所述控制连接来关闭或重置所述逻辑块:

• 通过所述第一接口从所述逻辑块发出的表明所述逻辑块已经完成数据处理或处于响应的通信;和/或

• 通过所述第一接口从所述逻辑块发出的表明恶意软件已经控制了所述逻辑块的通

信。

8.根据权利要求1或2所述的安全启动计算机系统，其中，所述控制器被配置为，响应于所述安全启动计算机系统在所述通信链路上接收到的通信，使用所述控制连接来开启、关闭或重置所述逻辑块。

9.根据权利要求1或2所述的安全启动计算机系统，其中，所述控制器被配置为，响应于来自传感器的输入和/或来自用户的输入，开启、关闭或重置所述逻辑块。

10.根据权利要求1或2所述的安全启动计算机系统，还配置为允许所述逻辑块通过网络接口与用户网络通信。

11.根据权利要求1或2所述的安全启动计算机系统，还包括一个或多个附加逻辑块，所述附加逻辑块包括一个或多个处理单元，所述一个或多个附加逻辑块中的每一个附加逻辑块被配置为在开启或重置相应的逻辑块时，根据第一通信协议通过相应的第一接口来请求启动指令；其中：

所述控制器还包括一个或多个附加接口管理器；

其中，每个相应的附加接口管理器还被配置为根据所述第一通信协议通过各个第一接口中的一个相应的第一接口与所述一个或多个附加逻辑块中的相应附加逻辑块通信，以识别通过所述相应的第一接口从相应逻辑块接收到的对数据的请求，并且通过相应的第二接口将所述请求发送至所述存储管理器；

所述一个或多个附加逻辑块中的每一个附加逻辑块被预先配置为根据所述第一通信协议在开启或重置所述附加逻辑块时通过相应的第一接口请求启动指令，其中所述附加逻辑块的预先配置不能被所述附加逻辑块执行的软件指令改变；

每个相应的第二接口始终是只读的，使得除了指定应该经由相应的接口管理器从所述存储管理器读取的数据这一目的之外，数据不能从相应的逻辑块流到所述存储管理器；

所述第三接口与每个相应的第二接口是完全独立的，使得所述启动指令能够通过所述第三接口被改变，但是不能通过任何相应的第二接口被改变；

所述系统还包括在所述控制器与所述一个或多个附加逻辑块中的每个附加逻辑块之间的相应的控制连接，所述控制器被配置为使用所述控制连接开启或重置所述逻辑块，以便将所述逻辑块置于执行所述启动指令而产生的预定活动状态，使得在开启或重置逻辑块之前在逻辑块上操作的软件不能影响所述预定活动状态。

12.根据权利要求1或2所述的安全启动计算机系统，其中，所述启动指令包括完整的操作系统。

13.一种安全启动计算机系统的方法，所述计算机系统包括逻辑块和控制器，所述逻辑块被预配置为根据第一通信协议在开启或重置所述逻辑块时通过第一接口与所述控制器通信以通过所述第一接口请求启动指令，其中所述逻辑块的预先配置不能被所述逻辑块执行的软件指令改变，所述控制器包括接口管理器和存储管理器，所述接口管理器被配置为根据所述第一通信协议通过所述第一接口与所述逻辑块通信，所述系统还包括所述逻辑块和所述控制器之间的控制连接，所述控制器还被配置为使用所述控制连接来开启或重置所述逻辑块，以便将所述逻辑块置于执行所述启动指令而产生的预定活动状态，使得在开启或重置所述逻辑块之前在所述逻辑块上操作的软件不能影响所述预定活动状态；

所述方法包括：

通过所述控制连接将指示开启或重置所述逻辑块的电力命令从所述控制器发送到所述逻辑块；

在开启或重置所述逻辑块时，根据所述第一通信协议经由逻辑块启动指令通过第一接口进行请求；

通过所述接口管理器识别通过所述第一接口从所述逻辑块接收到的对数据的请求；

由所述接口管理器通过第二接口将所述请求传输给所述存储管理器，其中所述第二接口始终是只读的，使得除了指定应该经由所述接口管理器从所述存储管理器读取的数据这一目的之外，数据不能从所述逻辑块流到所述存储管理器；

在所述控制器处实现到第二计算机系统的通信链路，以及由所述存储管理器通过第三接口从所述第二计算机系统接收启动指令，其中所述第三接口是与所述第二接口完全独立的，使得所述启动指令能够通过所述第三接口被改变，但是不能通过所述第二接口被改变；

经由所述接口管理器从所述存储管理器向所述逻辑块提供所述启动指令。

可通过网络更新的安全启动代码缓存

技术领域

[0001] 本发明涉及计算机系统的启动，尤其涉及一种用于安全启动的计算机系统和方法。

背景技术

[0002] 恶意软件是所有计算机系统的长久以来存在的问题。特别值得关注的是恶意软件在计算机系统重启后持续存在的能力(通过更改永久存储器(如硬盘或闪存)，使得在重新启动计算机系统时，这些更改可确保重新加载恶意软件)。

[0003] 虽然从只读存储器(ROM)启动(引导)的计算机可以解决此问题，但由于无法修补或升级启动软件(引导加载程序和/或操作系统)以解决诸如错误(bug)或安全漏洞之类的缺陷而引入了其他问题。因此，实际上，计算机不能从真正的ROM启动，尽管它们在某些情况下可能从“可编程ROM”(PROM)或“现场可编程ROM”(FPROM)启动。可编程ROM与标准ROM不同，因为PROM中的数据在制造后才被编写。在实践中，PROM是持久存储的一种形式。

[0004] 在美国专利申请US2011/0035808中描述了一种解决持久恶意软件问题的尝试。该方法描述了使用自定义存储控制器，该控制器阻止计算机系统对启动映像(boot image)进行更改，以避免重启后恶意软件持久的可能性。然后US2011/035808描述了一种机制，计算机系统凭此机制可以使某些情况下的启动图像的更改被插入物理加密令牌。本文描述的本发明的实施例不允许这样做，而是提供一种机制，由此可以由第二计算机系统对启动映像进行改变，该第二计算机系统独立于第一计算机系统而连接到存储控制器。

[0005] 在美国专利第6,546,489号中描述了解决该问题的另一种尝试。它描述了使用耦接到主计算机的磁盘驱动器来启动主计算机。磁盘驱动器使用于将主计算机的处理器置于主机处理器无法访问其存储器的非活动状态(inactive state)的信号有效(assert)。接下来，磁盘驱动器从磁盘驱动器的受保护区域写入一种模板，用于将主计算机引导到主机处理器的存储器。接下来，磁盘驱动器使上述信号无效(de-assert)，以允许主机处理器从主机存储器阵列启动。根据该方法，存储在磁盘的受保护区域中的启动映像的任何合理更改必须由在主计算机上运行的软件进行。因此，必须存在可以对启动影响进行更改的主计算机状态，并且原则上恶意用户可以使用恶意软件来生成该状态并改变启动映像。本发明的实施例不允许这样做，而是提供一种机制，由此可以由第二计算机系统对启动映像进行改变，该第二计算机系统独立于第一计算机系统而连接到存储控制器。

[0006] 美国专利7,293,165描述了一种系统，其中具有CPU的服务器还包括自主操作的基板管理控制器(BMC)，其通过LPC总线与CPU通信。BMC包括存储系统BIOS的存储器以及解码和响应总线访问请求(包括具有与启动操作相关联的存储器地址范围的请求)的总线接口。当CPU发出具有与启动操作相关联的地址范围的请求时，总线接口将该请求转发到存储器接口，该存储器接口从存储器读取系统BIOS并通过总线返回系统BIOS。存储器实现为SRAM，并且系统BIOS可以通过网络接口由远程计算机修补和更新。但是，没有考虑如何确保CPU可以恢复到已知的良好状态。

发明内容

[0007] 在现在参考的独立权利要求中限定了本发明。优选特征在从属权利要求中列出。

[0008] 根据本发明的第一方面，提供了一种安全启动计算机系统，包括逻辑块和控制器。逻辑块包括用于执行指令的一个或多个处理单元，并且被配置为根据第一通信协议，在开启或重置计算机系统时通过第一接口请求启动指令。控制器(可以实现为存储控制器)被配置为根据第一通信协议通过第一接口与逻辑块通信，并且还被配置为实现到第二计算机系统的通信链路，并从第二计算机系统接收启动指令。逻辑块被预先配置为根据第一通信协议以不能被由逻辑块执行的指令更改的方式通过第一接口进行通信。控制器被配置为防止或拒绝来自逻辑块的任何写入请求。该系统在逻辑块和控制器之间还包括控制连接。控制器还被配置为使用控制连接来开启或重置逻辑块以将逻辑块置于执行启动指令而产生的预定活动状态，使得在开启或重置逻辑块之前在逻辑块上操作的软件不能影响预定活动状态。

[0009] 预先配置逻辑块，使其在开启或重置时固定，从而根据第一协议通过第一接口与存储控制器通信，并且还防止安全启动计算机系统完成来自逻辑块的写入请求，这样就提供了一种计算机系统，可以防止可能试图破坏启动指令或启动映像的恶意软件的持久存在，同时允许使用单独的计算机系统更改将要进行的启动指令，并防止在安全启动计算机系统上运行的恶意软件对操作系统文件进行更改(这种更改将使得恶意软件在重新启动后可以持久存在)。这提供了一个安全的系统，该系统还允许更新启动映像。通过将该功能与控制功能相结合，存储控制器能够将逻辑块置于期望的预定活动结束状态，使得在该过程开始之前在逻辑块上操作的软件不会影响结束状态。因此，存储控制器能够将逻辑块置于活动状态，以执行或准备执行启动之后的其他指令，在此过程中可以确定逻辑块没有运行恶意软件。通过关闭然后再次打开，或者重置，并如上所述安排启动机制，可以强制逻辑块进入已知的良好状态，同时保留更新开启映像/指令的能力，所有这些都是以最低程度的损害风险进行的。

[0010] 存储控制器可选地包括接口管理器和存储管理器，接口管理器被配置为根据第一通信协议通过第一接口与逻辑块通信，并通过第二接口与存储管理器通信。接口管理器还可以被配置为通过第一接口识别对数据的请求，并通过第二接口将请求传送到存储控制器。

[0011] 第二接口可以是单向接口，使得除了指定应该经由接口管理器从存储管理器读取的数据之外，数据不可能从逻辑块流到存储管理器。通过始终使第二个接口为只读，不能通过第二个接口更改启动映像。

[0012] 逻辑块和存储控制器可以在同一物理外壳内实现，并且可以位于公共电路板上，或者位于同一集成电路上。重要的是，应该不能更改逻辑块和存储控制器之间的用于接收初始启动指令的初始通信。这可以通过将逻辑快硬连接以根据第一通信协议在第一接口上进行通信来实现。术语“逻辑上硬连线”在本文中用于指定不可由软件改变或不访问设备内部的任何配置。这可以包括使用一个或多个专用集成电路和/或一个或多个可编程逻辑器件或使用存储在只读存储器上的指令将功能物理硬件连线到逻辑块中。

[0013] 存储控制器可以在启动之前的时间点(在该时间点，逻辑块执行指令以启动计算机系统)处被配置为拒绝来自逻辑块的任何写入请求。存储控制器可以被编程为执行该功

能，例如使用形式上可验证或高保证的指令。或者，存储控制器可选地也可以以硬连线方式实现。可以看出，在实践中，如果可以拒绝来自逻辑块的写入，则精确实现不是必需的。

[0014] 通过通信链路传递的数据可能是安全启动计算机系统的任何其他组件不可访问的。控制器可以包括物理通信端口，该物理通信端口可以通过线缆连接到第二计算机系统。或者，控制器可以实现到第二计算机系统的加密链路，使得安全启动计算机系统的其他组件可以访问加密后的流量但是不能访问解密或加密所需的密钥。

[0015] 控制器可以包括用于存储启动指令的永久存储器，由此可以从第二计算机系统接收启动指令并在开启/重置或启动安全引导计算机系统之前存储启动指令。

[0016] 控制器可以被配置为，响应于来自逻辑块的对启动指令的请求，在接收来自逻辑块的请求时从第二计算机系统请求启动指令。

[0017] 第一接口上的协议可以是预先配置的简单接口协议，例如可以用于与闪存设备或硬盘通信的协议。

[0018] 启动指令可以包括重置向量。然后，控制器还可以提供位于由重置向量标识的地址处的指令。

[0019] 控制器还可以被配置为使用控制连接来关闭逻辑块，然后开启逻辑块，以便将逻辑块置于由执行启动指令而产生的预定活动状态。

[0020] 重置逻辑块可以包括关闭逻辑块，然后开启所述逻辑块，并执行逻辑块的硬件重置中的一个或多个。硬件重置是一种强迫逻辑块在不通过关闭/开启序列的情况下执行器初始开启过程的硬件级方法。

[0021] 控制器还可以被配置为使用控制连接来控制连接到逻辑块的RAM的电力状态。如果已运行恶意软件的逻辑块被关闭或重置，但RAM没有关闭或重置，则存在这种风险：一些数据可能留在RAM中，并且这将影响开启或重置逻辑块(101)之后的逻辑块(101)结束状态。重启后，此类数据可能会导致重复出现的危害。关闭RAM有助于确保重启前设备上运行的软件不会产生残留影响。

[0022] 控制连接可以连接到逻辑块的一个或多个电压输入引脚。然后，控制器可以通过控制是否将电压施加到一个或多个电压引脚来开启、关闭或重置逻辑块。

[0023] 控制连接可以连接到逻辑块的重置引脚。然后，控制器可以通过控制是否将电压施加到重置引脚来重置逻辑块。

[0024] 安全启动系统还可以包括电源管理集成电路(PMIC)。然后，控制连接可以在PMIC和逻辑块之间。响应于通过第一接口从逻辑块发出的表明逻辑块已经完成处理数据的通信，控制器可以被配置为使用控制连接来关闭或重置逻辑块。

[0025] 响应于通过第一接口从逻辑块发出的表明恶意软件已经控制了逻辑块的通信，控制器可以被配置为使用控制连接来关闭或重置逻辑块。控制器可以被配置为分析逻辑块通过第一接口发送的通信，以确定恶意软件是否已经控制逻辑块。

[0026] 通过第一接口从逻辑块发出的通信可以是一个或多个读取或写入请求。控制器可以被配置为响应于来自逻辑块的任何写入请求而关闭或重置逻辑块。

[0027] 控制器可以被配置为，响应于通信链路上的通信，使用控制连接来开启、关闭或重置逻辑块。通信链路上的通信可以来自第二计算机系统，或者可以来自不同于第二计算机系统的计算机系统。通信链路上的通信可以是用于开启关闭或重置逻辑块的请求。

[0028] 控制器可以被配置为,响应于来自传感器和/或用户的输入,开启、关闭或重置所述逻辑块。

[0029] 任何关闭命令之后可以是开启命令,以便将逻辑块置回到预定的活动状态。

[0030] 安全启动系统还可以被配置为允许逻辑块通过网络接口与用户网络通信。逻辑块可以被配置为经由控制器通过第一接口或者通过不同于第一接口的接口与网络接口通信。控制器可以被配置为,响应于通过第一接口从所述逻辑块发出的用于读取或写入网络数据的请求,将所述请求转发到网络接口。

[0031] 安全启动系统可以包括一个或多个附加逻辑块,所述附加逻辑块包括一个或多个处理单元。一个或多个附加逻辑块中的每一个附加逻辑块都将被配置为在开启或重置相应的逻辑块时,根据第一通信协议通过相应的第一接口来请求启动指令。控制器还将被配置为根据第一通信协议通过相应的第一接口与所述一个或多个附加逻辑块通信。一个或多个附加逻辑块中的每一个附加逻辑块将被预先配置为根据第一通信协议以不能被由逻辑块执行的指令更改的方式通过相应的第一接口与控制器通信。控制器将被配置为阻止完成来自一个或多个附加逻辑块的任何写入请求。安全引导系统在控制器与所述一个或多个附加逻辑块中的每个附加逻辑块之间还将包括相应的控制连接,由此控制器被配置为使用控制连接开启或重置所述逻辑块,以便将逻辑块置于执行启动指令而产生的预定活动状态,使得在开启或重置逻辑块之前在逻辑块上操作的软件不能影响所述预定活动状态。

[0032] 在包括一个或多个附加逻辑块的系统中,控制器可以包括每个逻辑块的相应接口管理器和公共存储管理器。每个相应的接口管理器将被配置为根据第一协议通过相应的第一接口与其相应的逻辑块通信。

[0033] 第一通信协议可以是SD卡协议、SDIO协议和eMMC协议中的一种。

[0034] 启动指令可以包括完整的操作系统。

[0035] 可以提供安全启动计算机系统的相应方法。计算机系统包括逻辑块和控制器,逻辑块被预先配置为根据第一通信协议以不能被由逻辑块执行的指令更改的方式通过第一接口与控制器通信,并且控制器被配置为阻止完成任何来自逻辑块的写入请求。该系统在逻辑块与控制器之间还包括控制连接。控制器被配置为使用所述控制连接来开启或重置逻辑块,以便将逻辑块置于执行启动指令而产生的预定活动状态,使得在开启或重置逻辑块之前在逻辑块上操作的软件不能影响所述预定活动状态。该方法包括:通过指示开启或重置逻辑块的控制连接将命令从控制器发送到逻辑块;在开启或重置逻辑块时,根据第一通信协议经由逻辑块启动指令通过第一接口进行请求;在控制器处实现到第二计算机系统的通信链路,并且从第二计算机系统接收启动指令;从控制器向逻辑块提供启动指令;以及随后在所述控制器处阻止完成任何来自逻辑块的写入请求。

[0036] 根据本发明的第二方面,提供了一种管理计算机系统,被配置为与本文描述的安全启动计算系统一起使用。管理计算机系统被配置为实现与安全启动计算系统的控制器的通信链路,并将启动指令发送到控制器。

[0037] 本发明的实施例将特别涉及计算机系统由除该计算机系统的用户之外的人员管理的场景。这可以存在于公司(云服务提供商)管理随后由第三方使用的计算机系统的云计算场景中,或者存在于IT部门管理雇员使用的计算机系统的公司场景中。

[0038] 在这种场景下,管理人员可以使用特殊管理计算机系统来对受管理计算机系统的

启动映像进行更改，而被引入到这些受管理计算机系统中的一个上的任何恶意软件在重新启动受管理计算机系统之后都不能持久存在。

附图说明

[0039] 现在将仅通过示例并参考附图进一步描述本发明，其中：

[0040] 图1是根据本发明实施例的第一和第二计算机系统的示意图；

[0041] 图2是对图1的第一和第二计算机系统提供了附加细节的第一和第二计算机系统的另一示意图；

[0042] 图3是还包括控制特征的第一和第二计算机系统的另一示意图；

[0043] 图4是还包括控制特征的第一和第二计算机系统的另一示意图；

[0044] 图5是包括集成网络接口的第一和第二计算机系统的另一示意图；

[0045] 图6是包括多个逻辑块的第一和第二计算机系统的示意图；

[0046] 图7是包括多个逻辑块的第一和第二计算机系统的另一示意图。

具体实施方式

[0047] 图1示出了由第二计算机系统(200)管理的第一计算机系统(100)。为了便于称呼，第一计算机系统将被称为安全启动计算机系统，第二计算机系统将被称为管理计算机系统。

[0048] 安全启动计算机系统(100)通常包括一个或多个CPU(101a、101b等)，一些RAM(115)和专用存储控制器(150)。安全启动计算机系统(100)还可以包含其他组件(110)，其他组件可能包括永久存储器，但是被配置为从存储控制器(150)寻找其启动映像。除了物理拆解计算机系统(100)之外，CPU(一个或多个)与存储控制器之间的这种配置不应该通过任何方式改变。

[0049] 存储控制器(150)实现到管理计算机系统(200)的通信链路(300)。通信链路可以实现为安全启动计算机系统与管理计算机系统之间的物理有线连接，或者实现为有线和/或无线网络连接。通信链路可以以这样的方式实现：该链路(300)上的通信不能被安全启动计算机系统(100)的任何其他组件访问。例如，在一个实施方式中，存储控制器(150)包含物理通信端口，该物理通信端口经由线缆连接到管理计算机系统(200)。在另一实施方式中，存储控制器(150)以这样的方式实现到管理计算机系统(200)的加密链路：使安全启动计算机系统(100)的其他组件可能能够访问加密后的流量，但是不能访问解密或加密所需的密钥。

[0050] 在开机或重置之后，当安全启动计算机系统(100)被启动并请求存储控制器(150)提供启动映像时，由存储控制器(150)提供的字节由计算机系统(200)来确定。

[0051] 管理计算机系统可以在启动时将启动映像提供给存储控制器，或者它可以在启动之前的一个时刻提供启动映像。例如，在一个实施方式中，存储控制器(150)包含本地存储器，例如磁盘或存储器(例如闪存)，并且在启动之前，管理计算机系统(200)先前已经指示存储控制器(150)要供应来自这个本地存储器的哪些数据，如果这些数据还没有出现在本地存储介质上则可以向存储控制器(150)供应这些数据。为避免疑义，可以向存储控制器(150)提供预先安装的启动数据，然后可以根据需要由管理计算机(200)更新启动数据。在

另一实施方式中，当安全启动计算机系统(100)请求启动映像时，存储控制器(150)联系管理计算机系统(200)以确定应该提供什么数据以响应该请求，这些数据此时通过通信链路(300)被供应到存储控制器(150)。

[0052] 存储控制器(150)响应来自安全启动计算机系统(100)的其他读取请求的行为也可以由管理计算机系统(200)确定。实际上，典型的启动序列由一系列读取组成，其中第一次读取来自定义的存储位置，随后的读取位置是根据先前读取的结果确定的。管理计算机系统(200)通常将指示存储控制器(150)如何最低限度地对这一系列读取作出响应。

[0053] 使用该功能，管理计算机系统(200)可以升级、修补或完全替换要由安全启动计算机系统(100)启动的操作系统。

[0054] 还应考虑存储控制器(150)响应来自安全启动计算机系统(100)的写入请求的行为。通过防止安全启动计算机系统(100)完成写入请求，存储控制器(150)防止在安全启动计算机系统(100)上运行的恶意软件对操作系统文件进行将在重启之后允许持续存在的更改。为了最有效的安全性，存储控制器(150)应该以这样的方式实现：最小化安全启动计算机系统(100)上操作的恶意软件可以使用存储控制器(150)和安全启动计算机系统(100)的其余组件之间的接口作为损害(compromise)存储控制器(150)并改变其上述行为的手段的可能性。在实际实施方式中，该接口可以使用由读取和写入请求以及相关响应组成的简单协议。

[0055] 使用简单协议主要是由于需要将逻辑块硬连线。可以使用更复杂的协议，但是实践中以硬连线方式实现是昂贵且有风险的。这种接口也适合于在存储控制器处使用特别难以损害的方法来实现。这可以包括使用一个或多个专用集成电路(ASIC)或者一个或多个可编程逻辑器件(PLD)(例如，诸如现场可编程门阵列(FPGA)之类的器件)的实现方式。或者，可以使用利用形式化方法开发的软件，例如形式上可验证或高保证指令。如今，大多数现有存储控制器使用ASIC技术实现这种类型的接口。

[0056] 图2更详细地示出了第一计算机系统的某些组件和连接实体。与图1一样，在顶层，系统由两个计算机系统组成。第一计算机系统(100)(安全启动系统)连接到诸如互联网或企业网络之类的用户网络(400)。第二计算机系统(200)(管理计算机系统)连接到管理网络(300)。管理网络(300)是专用于管理安全启动计算机系统(例如，安全启动计算机系统(100))的离散网络环境。

[0057] 如前所述，安全启动计算机系统(100)包括存储控制器(150)，该存储控制器现在更详细地示出。存储控制器(150)连接到管理网络(300)，并且以这种方式，它可以与管理计算机系统(200)进行通信。

[0058] 可以看出，安全启动计算机系统(100)由许多逻辑组件组成。这些组件可以是连接在印刷电路板(PCB)上的分立集成电路(IC)，或者可以是连接在单个IC内的不同逻辑块。

[0059] 逻辑块(101)是计算机系统的核心，并且包含一个或多个能够执行软件指令的处理单元，例如图1的处理器101a和101b。逻辑块执行在开机或重置时启动系统所需的初始外部提取指令。通过外部提取意味着指令是从逻辑块的逻辑和/或物理边界之外的组件(如果需要实现第一通信协议，则除了真正的ROM)提取的。对于单个CPU，这可能是通过CPU总线接收的第一组用于处理的指令。

[0060] 逻辑块(101)通过接口102连接到存储控制器(150)。以这种方式，在开机或重置之

后，逻辑块101将从存储控制器(150)寻找其初始指令。逻辑块(101)的设计使得该行为是“硬连线的”，“已接线的”或“已制造的”。换句话说，在逻辑块(101)上执行的软件指令无法改变这种行为。

[0061] 在大多数情况下，这是通过例如在印刷电路板上以特定方式布线逻辑块101的可启动IC的引脚来实现的。可以通过向特定引脚施加特定电压或电流来指定启动接口和协议。或者，通过仅对响应特定协议的系统布线特定引脚，而不布线接其他引脚，可以对这些引脚布线，使得只能启动一个接口和协议。另一种方法是使用附加的ROM，其中IC寻求存储在附加ROM上的配置细节。该ROM可以是真正的(不可重写的)ROM。或者，可以使用EPROM。EPROM可以由软件进行一次编程，但随后需要在重新编程之前擦除(例如，通过暴露于UV光，这需要进入装置的内部)。作为另一种选择，可以使用修改的EEPROM。EEPROM通常不能实现该目的，因为附接到IC的EEPROM可以由在逻辑块上执行的软件指令重新编程以改变启动接口和/或协议。但是，某些EEPROM可以通过布线特定引脚来设置为“只读”模式。例如，可以布线IC和EEPROM组合，使得在IC接电时EEPROM被设置为只读模式，这实现了目标，因为为了运行软件IC必须通电。例如，如果EEPROM上的相关布线要将引脚设置为高电平，则可以将该引脚简单地连接到IC上的电源引脚。

[0062] 另一种替代方法是在一些IC上使用板载熔丝。可以熔断这种板载熔丝(例如，通过向相关引脚施加过电压)，以便产生不可逆的启动配置。在某些情况下，这无法实现目标，因为软件可能能够进一步熔断熔丝并响应地改变启动配置。然而，一些IC仅允许通过施加外部电压(其不能通过软件改变)来熔断熔丝，在这种情况下，熔断适当的熔丝配置将实现该目的。

[0063] 逻辑块(101)通常被连接到许多附加组件。这些附加组件中的一个组件可以是网络接口(105)，其允许在逻辑块(101)上执行的软件与连接到用户网络(400)的其他计算机系统进行通信。其他组件(110)可以包括RAM和永久存储器，以及其他网络接口。

[0064] 存储控制器(150)包含接口管理器(151)，该接口管理器负责管理公共接口(102)上与逻辑块(101)的通信。接口管理器(151)还通过接口155与存储管理器功能单元(152)通信。存储管理器可以是在存储控制器内的专用处理器上执行的软件。

[0065] 接口管理器(151)负责识别来自逻辑块(101)的请求，以从一些定义的存储标识符读取数据或向一些定义的存储标识符写入数据。取决于接口102上使用的协议，这种标识符可以是存储器地址或块地址。接口管理器将从逻辑块识别有效的读取请求，并将这些请求传送给存储管理器(152)，存储管理器可以返回数据以作为响应。如果存储管理器返回数据作为响应，则接口管理器将此数据返回到逻辑块。

[0066] 接口管理器(151)还可以识别来自逻辑块(101)的写入请求。可选地，尽管接口管理器可以如同写入请求已被成功执行那样(尽管没有被成功执行)被配置为响应逻辑块，但这些请求将不被传送到存储管理器(152)。通过这种方式，逻辑块可以从存储管理器读取数据，但无法向其写入数据。

[0067] 接口155可以始终是单向的，具有完全独立的接口156以用于改变提供给逻辑块101的启动映像。可以通过接口156合法地改变启动映像，但是不能通过接口155改变启动映像。如果在接口管理器151中存在实施缺陷，则只能通过接口155改变启动映像。接口管理器151比(例如)在主计算机上运行的操作系统简单得多，因此接口管理器151中存在缺陷的可

能性比操作系统中可利用的缺陷的可能性小得多，这使得当前方案非常安全。此外，该方案允许在没有任何物理干预的情况下更新启动映像。

[0068] 存储管理器(152)通过网络接口(153)连接到管理网络(300)，并且通过该网络可以与管理计算机系统(200)通信。使用该通信信道，管理计算机系统可以指示存储管理器要返回什么数据来响应从接口管理器(151)接收的读取请求。该指令可以在接收到来自接口管理器的读取请求时发生，并且存储管理器可以在接收到这样的读取请求时向管理计算机系统发出请求以确定要返回什么数据。或者，该指令可以在来自接口管理器的读取请求之前发生，并且管理计算机系统可以预先指示存储管理器应该返回什么数据来响应来自接口管理器151的读取请求，其中存储管理器将指令数据存储在存储器中。

[0069] 因为接口102和接口155通常都是简单的低级接口，所以接口管理器(151)可以使用“高保证(high assurance)”方法来实现，例如在硬连线逻辑中通过ASIC或PLD或者使用形式上可验证的软件指令实现。期望这样以便最小化在逻辑块(101)上执行的恶意软件可能损害接口管理器(151)的行为的风险。然而，应该理解，存储控制器不是必须使用高保证方法来实现。为了实现安全目的，需要的是，在启动后的某个时间，如果恶意软件设法控制逻辑块(101)，则该恶意软件将发现难以使控制器(150)以其设计者的意图不同的方式来运转。由于控制器的功能相对简单(主要是从管理计算机接收启动指令并拒绝来自逻辑块的写入)，所以在没有通过形式方法被专门设计的情况下，实施方式可能在实践中本来就难以破解。

[0070] 作为一个极端的示例，存储管理器(152)的实施方式可能(错误地或恶意地)包括一种代码，其允许恶意软件获得对存储管理器的控制，并且理论上修改其行为，使得在后续启动时，它返回与管理计算机系统(200)指示其返回的数据不同的数据。但是，这种情况不太可能在实践中引起最低限度的安全问题。更实际的问题是接口管理器(151)的不良实施方式可能允许非预期输入访问(或通过)接口102使接口管理器开始执行恶意代码。一个更现实的问题的原因是，虽然接口管理器(151)和存储管理器(152)之间的协议可能非常简单(例如仅仅是对特定存储块的请求)，但是运行在接口102上的协议可能相对更复杂(例如在下面描述的PXE启动的情况下)，因此找到导致接口管理(151)的实施以不期望的方式运行的非预期输入的机会相对较高。

[0071] 通过以“高保证”方式开发接口管理器(151)，可以降低跨接口102的损害风险。可以实现这一目标的方法包括：

[0072] -使用ASIC或PLD而不是图灵机中实现的逻辑设计。注意到，巧合的是，这可能涉及通过逻辑硬连线而被预先配置的接口管理器，但只要PLD不由逻辑块(101)重新编程，则PLD原则上同样可以由存储管理器(152)重新编程；

[0073] -在图灵机上运行的软件，前提是该软件是使用形式化方法开发的，因此已知不会包含任何非预期行为的范围。这不需要在逻辑上是硬连线的；如果不能从逻辑块(101)访问该存储器，则可以从可读写存储器装载该软件。

[0074] 实际上，这些方法对于PXE启动或类似的布置是不切实际的，但对于更简单的协议来说是实际的。

[0075] 为了理解本发明实施例的优点，与已知的网络启动技术进行比较是有帮助的，例如基于Intel 8086CPU的x86技术的PXE启动。

[0076] 在网络启动场景中,计算机系统配备有一个或多个“启动ROM”,其用于引导计算机并指示其从网络服务器取回启动映像。在x86系统的预启动执行环境(PXE)的情况下:

[0077] -第一个启动阶段来自BIOS或统一可扩展固件接口(UEFI)ROM。

[0078] -此阶段标识PXE ROM。

[0079] -PXE ROM包含使用DHCP获取IP地址的指令。

[0080] -然后,PXE ROM包含使用TFTP从网络服务器取回文件的指令。

[0081] -然后将控制传递给取回的映像文件中的指令。

[0082] 可以将网络服务器配置为禁止TFTP置入/写入,并且可以根据需要在网络服务器上更改映像文件。这允许更新映像文件,但不允许更新PXE ROM。

[0083] 值得注意的是,在实践中,BIOS/UEFI和PXE ROM从不是真正的ROM,事实上是可编程ROM(如上所述,它是永久存储器的形式)。这是必要的,因为这种网络启动过程涉及相当大的复杂度,很可能在某些时候需要进行升级和修补。

[0084] 本发明的实施例在许多方面不同于这种已知的网络启动方案。

[0085] 在计算机系统(100)内以真正的硬连线形式提供对CPU(101)的逻辑边界与存储控制器(150)之间的协议的支持,或者通过在真正(不可重写)ROM中提供的指令提供支持,或者通过硅片上的硬连线逻辑提供支持。逻辑块101是真正的“只读”,而在PXE启动中,逻辑块包含可擦写PROM。因此,该协议可能是一个简单的预配置接口协议,例如,用于从闪存芯片或硬盘取回字节的协议。其他示例可以包括用于从RAM或SD卡取回数据的协议。使用硬连线布置(实现这类基础通信协议)使得在不具有对计算机系统的内部的物理访问的情况下难以篡改启动过程。接口102上使用更简单的协议使得使用“高保证”工程方法来开发接口管理器(151)在实践上是可行的。

[0086] 通过使用对于CPU设计或包含多个CPU的逻辑块来说是不可或缺的预配置通信协议,存储控制器(150)可以以这样的方式在系统中定位:存储控制器(150)响应最早或最一开始的请求,在该请求中,CPU(101)使得从其物理和/或逻辑边界之外的独立组件取回数据,以便启动系统。因此,在x86启动过程的情况下,存储控制器(150)将模拟BIOS/UEFI ROM,向CPU(101)提供重置向量和该地址处的指令。

[0087] 鉴于以上两点,恶意软件无法将该请求之前执行的指令存储到存储控制器(150)。CPU(101)和存储控制器(150)之间的协议预先配置在CPU(101)的架构内。因为协议非常简单,并且对于CPU的操作来说是基础的,所以它不能通过软件手段改变,并且存储控制器(150)可以选择性地使用硬连线逻辑或形式上可验证的微控制器指令串来管理其协议的结束,这意味着在计算机系统(100)上运行的恶意软件也将永远不能使用该协议作为损害存储控制器(150)中的逻辑的手段。这与IP/以太网网络的复杂性形成了对照,IP/以太网网络通常由许多组件组成,例如交换机、DHCP服务器等,并与基于软件的TFTP服务器相结合。这种复杂的系统不适合使用硬连线逻辑或形式上可验证的指令来实现,因此可能包含可能被在计算机系统上运行的恶意软件利用的漏洞。

[0088] 另一个有用的比较是与允许恢复“程序块(bricked)”系统的系统的比较。某些系统(至少包含CPU和启动PROM的PCB)允许附接线缆,这允许第二系统对启动PROM重新编程。如此设置的原因是为了在系统在“程序块化”的情况下允许恢复,即在CPU上运行的软件不正确地对PROM重新编程并且这样甚至不可能实现最低程度的CPU启动。一般用户无法使用

此类线缆。在这类布置中，CPU上运行的软件可以重新刷新PROM。根据本发明的实施例，这是不可能的。

[0089] 应当注意，许多处理器具有两级启动过程，由此它们首先从第一源(source)寻求指令，例如EEPROM。来自第一源的指令指示处理器从第二源(例如SD卡)寻求进一步的指令。本发明的任何实施例可以在这样的系统中实现，其中启动过程的第一阶段由上述控制器组件和逻辑块代替，该逻辑块被预先配置为根据第一通信协议用由计算机系统执行的指令不能改变的方式在第一接口上与控制器通信。

[0090] 现在将提供可用于实现本发明实施例的各种组件的具体示例。应当理解，可以使用不同的等效组件，同时仍然实现本文描述的相同功能步骤。

[0091] 逻辑块101可以是Freescale™ i.MX6处理器，其被配置为使用4线SPI协议通过接口102来寻求其初始启动指令。接口管理器151可以使用FPGA实现，并且存储管理器152可以使用Texas Instruments™开放多媒体应用平台(OMAP)处理器和相关联的双端口存储器来实现。FPGA可以被配置为充当串行外设接口(SPI)从设备，通过接口102与充当SPI主设备的i.MX6处理器进行通信。FPGA还可以被配置为使用接口155从存储管理器的双端口存储器读取。

[0092] 在SPI启动后，FPGA从SPI主设备输出从设备输入(MOSI)线读取比特，并识别i.MX6处理器发出的任何指令，如ERASE、ERAL、EWDS、EWEN、READ、WRITE和WRAL。对于除READ之外的所有指令，FPGA如协议指定的那样回复i.MX6，但不采取进一步行动。

[0093] 在通过SPI MOSI(主设备输出，从设备输入)线接收到READ指令的情况下，FPGA可以解析作为READ指令一部分发送的寻址存储器位置。然后，FPGA使用接口155从存储器中的适当地址执行读取。FPGA通过SPI MISO(主设备输入，从输出输出)线向i.MX6输出8比特读数。如果SPI从设备(从属设备)选择(SS)线由i.MX6保持高电平，FPGA随后通过SPI MISO线从存储器返回接下来的8个比特。这可能会持续到SPI SS线不再保持高电平为止。READ指令可以这种方式执行，直到i.MX6启动为止。

[0094] 如果需要对启动数据进行更改，则TI OMAP处理器可以使用网络接口153取回数据，并使用第二端口将其写入存储器。如上所述，简单的协议是期望的，因为从经济和技术上看它们在硬件上实施都是可行的，这使得系统即使不是不可能也很难被损害。然而，一系列协议可能是可行的，其中一些协议比其他协议更复杂，并且更复杂的协议可能功能会增加或者是有利的。在实际实施中，“最佳”协议因此可以被认为是从经济角度和技术角度上实施起来是可行的最复杂协议。这将根据应用而有所不同。

[0095] 作为另一示例，逻辑块101被配置为使用SD卡协议或嵌入式多媒体记忆卡(eMMC)协议通过接口102寻求其初始启动指令。

[0096] 与SPI协议和其他简单存储器访问接口相比，SD卡协议和eMMC协议允许逻辑块101发出的读取请求与接口管理器151提供的数据之间的延迟更大。简单存储器访问接口必须在预定数量的时钟周期内回复所请求的数据，而SD卡协议和eMMC协议允许在接口管理器151必须回复数据之前有更大的延迟。

[0097] SD卡协议和eMMC协议所允许的更大的时序范围相比于诸如SPI的简单存储器访问接口可以为实现方式提供更宽的选择。例如，虽然简单的存储器访问接口可能需要将存储器实现为SRAM，但SD卡协议或eMMC协议可能允许使用存储管理器152中的DRAM。使用DRAM与

SRAM相比可能允许存储管理器152包含更大的数据存储，并且可以允许存储控制器150提供完整的操作系统映像而不仅仅是BIOS或引导加载程序。此外，在单个存储控制器服务多个逻辑块的实施方式中(如下面参考图6和7所述)，使用SD卡协议或eMMC协议允许并行地启动多个逻辑块，而不是依序启动。如果使用具有严格时序要求的简单存储器访问协议，则需要立即处理每个读取请求。这意味着任何并行性都受SRAM的峰值传输速率的限制。由更复杂的协议提供的更宽的延迟范围允许读取请求排队，并且对于DRAM的请求展开以实现更高的总吞吐量(即使峰值传输速率可能是相同的)。

[0098] 图3示出了对应于上面参考图2描述的第一和第二计算机系统的第一和第二计算机系统，但是这里的第一和第二计算机系统还包括控制特征。

[0099] 可以看出，图3的安全启动系统(100)包括与图2的安全启动系统(100)相同的组件、连接和接口，但还包括将逻辑块(101)连接到存储控制器(150)的控制连接(140)。存储控制器(150)可以使用控制连接(140)来开启逻辑块(101)，关闭(power-down)逻辑块，或者重置逻辑块。特别地，这允许逻辑块从非活动状态被开启或进行重置，以通过执行启动指令迫使其进入已知的良好状态。

[0100] 本文使用的术语“重置(reset)”包括逻辑块的重启循环(power-cycle)(关闭逻辑块(101)然后再开启逻辑块的序列)，以及可以称为“硬件重置”的意思。硬件重置是一种强制逻辑块执行其初始开启过程而无需通过关闭/开启序列的硬件级方法。

[0101] 本文使用的术语“开启(power-up)”是指将逻辑块从非活动关闭状态开启，在非活动关闭状态中，逻辑块已经处于非活动状态相对长的时间。可以理解的是，在逻辑块开启的任何时候，在某一时刻开启之前都是关闭。然而，本文使用的术语“开启”都是与重新启动重置相对使用的，其中，开启在在前的关闭之后的一较短时间出现。

[0102] 控制连接(140)可以将存储控制器(150)连接到实现逻辑块(101)的一个或多个集成电路的电压输入引脚。以这种方式，存储控制器(150)可以控制电压是否施加到实现逻辑块(101)的一个或多个集成电路，并因此控制逻辑块(101)的电力状态(power state)。

[0103] 可替代地或另外地，控制连接(140)可以将存储控制器(150)连接到实现逻辑块(101)的集成电路的硬件重置引脚。将硬件重置引脚上的电压拉低并将电压返回到高电平将使逻辑块(101)执行其初始启动过程。

[0104] 存储控制器(150)可以包括电源管理集成电路(PMIC)(图3中未示出)。在存储控制器包括PMIC的情况下，控制连接(140)可以将PMIC连接到实现逻辑块的一个或多个集成电路的电压输入引脚和/或硬件重置引脚。

[0105] 如前所述，逻辑块(101)被预先配置为在开启或重置之后立即从存储控制器(150)寻求启动指令。通过将该功能与控制功能相结合，存储控制器(150)能够有利地将逻辑块(101)置于期望的活动结束状态，使得在该过程开始之前在逻辑块上操作软件不能影响结束状态。如果逻辑块(101)处于未知的活动状态(其可能正在运行恶意软件)，则存储控制器(150)可以关闭然后再开启或重置逻辑块。在此之后，逻辑块(101)将在启动后处于期望的活动状态，使得可以确定存储控制器(150)没有运行恶意软件。

[0106] 控制器还可以控制连接到逻辑块(101)的RAM的电力状态，例如图1中所示的RAM 115。如果关闭或重置已运行恶意软件的逻辑块(101)，但RAM未被关闭或重置，则存在这样的风险：一些数据可能留在RAM中，并且这将影响在重置或随后开启逻辑块(101)之后的逻

辑块(101)结束状态。重启之后,此类数据可能导致重复出现的危害。关闭RAM有助于确保在重新启动之前,设备上运行的软件不会产生任何残留影响。在一些实施方式中,RAM可以被关闭得足够长以避免数据剩磁的问题。

[0107] 存储控制器(150)可以包括其他逻辑组件,用于确定是否应该使用控制连接(140)来开启、关闭或重置逻辑块(101)。例如,存储控制器(150)可以包括生命周期管理器(154),如图4所示。

[0108] 存储控制器(150)可以通过多种方式确定它应该使用控制连接(140)来开启、关闭或重置逻辑块(101)。

[0109] 存储控制器(150)可以确定其应该开启、关闭或重置逻辑块(101)的一种方式涉及存储控制器通过接口102从逻辑块接收通信。基于通信,存储控制器可以确定逻辑块(101)应该被关闭或重置。

[0110] 在这样的一个示例中,存储控制器(150)接收来自逻辑块(101)的通信,该通信指示逻辑块已完成数据的处理。然后,存储控制器可以确定逻辑块(101)可以关闭,并使用控制连接(140)来关闭逻辑块(101)。在另一示例中,存储控制器(150)从逻辑块(101)接收读取或写入请求,并且存储控制器基于读取或写入请求确定逻辑块(101)需要被关闭或重置。例如,某些读取或写入请求可能意味着恶意软件已经控制了逻辑块,在这种情况下,可能需要关闭或重置逻辑块(101)。在某些情况下,可能不期望在逻辑块(101)上运行的软件向存储控制器(150)发出任何写入请求。在这种情况下,任何写入请求都可以被视为恶意软件正在逻辑块(101)上运行的证据。

[0111] 存储控制器可以确定其应该开启、关闭或重置逻辑块的另一种方式涉及存储控制器(150)从管理网络(300)接收输入。这可以是来自管理计算机系统(200)的输入,或者可以是来自连接到管理网络(300)但未在图3或4中示出的另一系统的输入。在一个示例中,管理网络(300)发出开启、关闭或重置逻辑块(101)的请求,并且存储控制器(150)通过使用控制连接(140)来开启、关闭或重置逻辑块(101)进行响应。在一些情况下,逻辑块(101)最初可以由第一用户使用,并且管理计算机系统(200)(或其用户)可以决定应该将对逻辑块(101)的访问分配给第二用户,第二用户相比于第一用户可能是更高优先级的用户。在这种情况下,存储管理器(150)将请求逻辑块(101)被重启循环到已知的良好状态,之后可以将其分配给更高优先级的用户。

[0112] 存储控制器(150)可以确定其应该开启、关闭或重置逻辑块(101)的第三种方式涉及存储控制器(150)通过某些其他先前未指定的接口接收输入,例如接口160或接口161。接口160将安全启动系统(100)先前未指定的组件(130)连接到存储控制器(150),并且接口161将一些先前未指定的计算机系统(500)连接到存储控制器(150)。

[0113] 例如,组件130和/或计算机系统500可以基于传感器输入、用户输入或来自某些其他网络连接的输入来生成用于存储控制器(150)的输入。基于通过接口160和/或接口161接收的输入,存储控制器(150)可以确定它应该开启、关闭还是重置逻辑块(101)。在一些情况下,逻辑块(101)可以用作图3或图4中未示出的单独的处理块的“从设备”。在这种情况下,传感器或用户输入可以指示单独的处理块不再需要从设备处理块(101),使得逻辑块(101)可以被关闭或重置。

[0114] 应该理解,存储控制器(150)可以被配置为确定它应该以上述任何一种或多种方

式以及上文未描述的其他方式开启、关闭或重置逻辑块(101)。还应理解,组件130、计算机系统500和接口160和161对于实现控制功能不是必需的。

[0115] 图5示出了与上面参照图2至图4描述的计算机系统基本相同的第一和第二计算机系统,但是其中以替代方式实现网络接口105(如果存在的话)。

[0116] 在图5所示的实施例中,网络接口105没有连接到逻辑块(101),如图2至图4所示。网络接口105反而经由先前未指定的接口162连接到存储控制器(150)的接口管理器(151)。

[0117] 根据这种布置,通过接口102与存储控制器(150)通信,逻辑块(101)经由网络接口105与用户网络400进行通信。因此,接口102是集成接口,用于从存储控制器(150)请求启动指令并与用户网络(400)通信。在实现逻辑块(101)的一个或多个集成电路的可用引脚的数量有限的情况下,将接口102实现为集成接口可能是有利的。

[0118] 为了将接口102实现为集成接口,接口管理器(151)识别它通过接口102接收的通信并进行适当响应。如果接口管理器(151)将通信识别为读取或写入存储数据(例如启动映像)的请求,则接口管理器(151)将请求转发到存储管理器(如果是读取请求)或丢弃该请求(如果是写入请求),如上参考图2所述。如果接口管理器(151)将通信识别为读取或写入网络数据的请求,则接口管理器(151)通过接口162将请求转发到网络接口105。

[0119] 图5的实施例的一个示例性实施方式利用SDIO协议,该协议允许设备在同一物理接口上提供存储服务和输入/输出(I/O)服务。因此,接口管理器(151)将负责确保由存储管理器(152)处理读取存储数据的请求,确保丢弃写入数据的请求,并且确保由网络接口(105)处理网络读取和写入请求。

[0120] 应当理解,虽然图5示出了上面参考图3和图4描述的控制连接(140),但是图5的替代网络接口可以在有或没有控制连接的情况下实现。

[0121] 图6和图7示出了安全启动系统(100)和管理计算机系统(200),其中安全启动系统(100)包括多个逻辑块(101a、101b、101c)。

[0122] 从图6和图7中可以看出,安全计算机系统(100)包括多个逻辑块(101a、101b、101c)和存储控制器(150)。每个逻辑块(101a、101b、101c)通过接口(102a、102b、102c)连接到存储控制器(150),并且存储控制器(150)连接到管理网络(300),以允许存储控制器(150)与管理计算机系统(200)通信。每个逻辑块(101a、101b、101c)被配置为在开启或重置时从存储控制器(150)请求启动指令,如上面参考图1至图5所述。

[0123] 图6和图7中所示的每个单独的组件、连接和接口可以是相同的,并且可以以与上面参照图1至图5描述的相应组件、连接和接口相同的方式运转。但是,在图6和图7的实施例中,单个存储控制器(150)控制多个逻辑块(101a、101b、101c)。具体地,多个逻辑块(101a、101b、101c)在开启或重置时从同一存储控制器(150)和相同存储管理器(152)寻求启动指令。

[0124] 每个逻辑块(101a、101b、101c)可以与上面参考图1至图5中的任何一个描述的逻辑块(101)相同。虽然图6和图7示出了三个逻辑块(101a、101b、101c),但是应该理解,这仅仅是出于说明的目的,并且任何大于一的数量的逻辑块与图6和图7的实施例是相符的。

[0125] 存储控制器(150)包括一个或多个接口管理器(151a、151b、151c)、存储管理器(152)和连接到管理网络(300)的网络接口(153),以允许存储控制器(150)与管理计算机系统(200)通信。图6和图7示出了控制多个逻辑块(101a、101b、101c)的单个存储管理器。然

而，安全启动系统(100)可以包括多个存储控制器(150)，每个存储控制器控制不同的多个逻辑块。

[0126] 一个或多个接口管理器(151a、151b、151c)中的每一个可以与上面参考图1至图5中的任何一个描述的接口管理器(151)相同。一个或多个接口管理器(151a、151b、151c)中的每一个通过接口(155a、155b、155c)连接到存储管理器(152)，并且一个或多个接口管理器(151a、151b、151c)中的每一个通过接口(102a、102b、102c)也连接到一个逻辑块(101a、101b、101c)。虽然图6和图7示出了相应数量的逻辑块(101a、101b、101c)和接口管理器(151a、151b、151c)，但这不是必需的。例如，可以存在使用多个接口(102a、102b、102c)连接到多个逻辑块(101a、101b、101c)的单个接口管理器(151)。

[0127] 存储管理器(152)可以与上面参考图1至图5中的任何一个描述的存储管理器(152)相同。存储管理器(152)通过一个或多个接口155a、155b、155c连接到一个或多个接口管理器(151a、151b、151c)，如上所述，并且还通过接口156连接到网络接口153。将一个或多个接口管理器(151a、151b、151c)连接到存储管理器(152)的一个或多个接口(155a、155b、155c)可以与上面参考图1至图5描述的接口(155)相同。同样地，将存储管理器(152)连接到网络接口153的接口(156)可以与上面参考图1至图5描述的接口156相同。

[0128] 网络接口153可以与上面参考图1至图5中的任何一个描述的网络接口153相同。除了通过接口156连接到存储管理器(152)之外，网络接口153还经由管理网络(300)连接到管理计算机系统(200)。管理计算机系统(200)和管理网络(300)可以用于相同的目的，并且以与参考图1至图5中的任何一个描述的相同的方式运行。

[0129] 图6的安全计算机系统(100)还可以包括网络接口105，其可以与参考图2描述的网络接口105相同，或者可以与上面参考图5描述的网络接口相同。因此，虽然图6示出了通过接口162a、162b、162c连接到一个或多个接口管理器(151a、151b、151c)的网络接口105，但是应该理解，网络接口105也可以通过接口(103)连接到每个逻辑块(101a、101b、101c)，如参考图2所述。或者，如图7所示，可以有多于一个的网络接口(105a、105b、105c)。一个或多个网络接口(105a、105b、105c)中的每一个可以连接到一个或多个逻辑块(101a、101b、101c)，如图7所示并且类似于图2，或者可以连接到一个或多个接口管理器(151a、151b、151c)，类似于图5和图6。

[0130] 将逻辑块(101a、101b、101c)连接到一个或多个接口管理器(151a、151b、151c)的每个接口(102a、102b、102c)可以与以上参考图1至图5描述的接口(102)相同。图6和7将每个接口102a、102b、102c示出为上面参考图5描述的集成接口，但是应该理解，接口102a、102b、102c中的每一个也可以是上面参考图2和图3描述的接口102。图6和图7还示出了通过控制连接(140a、140b、140c)连接到存储控制器的每个逻辑块(101a、101b、101c)，使得图6和图7的实施例可以实现以上参考图3和图4描述的控制功能。在这种情况下，控制功能可以如参考图3和图4所描述的那样实现，并且安全启动系统(100)可以包括参考图3和图4描述但在图6和图7中未示出的任何附加组件、连接和接口。还应该理解，图6和图7的实施例可以在没有控制功能的情况下实现，在这种情况下，控制连接(140a、140b、140c)不是必需的。

[0131] 在图6和图7的实施例包括控制功能的情况下，存储控制器能够独立地开启、关闭或重置多个逻辑块(101a、101b、101c)中的每一个。因此，具有单个存储管理器(152)的单个存储控制器(150)能够有利地从单个存储管理器(152)为多个逻辑块提供同一组启动指令，

但是独立地控制多个逻辑块的电力状态。

[0132] 图6和7的实施例还可以包括参考图1至图5描述但未在图6和图7中示出的任何其他组件。例如，安全启动系统(100)可以包括参考图1和图2描述的其他组件(110)，参考图3和图4描述的组件130和接口160，并且可以通过接口161连接到计算机系统500，如参考图3和图4所描述的那样。

[0133] 以上描述了具有各种可选特征的多个实施例。应当理解，除了任何互斥的特征之外，一个或多个可选特征的任何组合都是可能的。

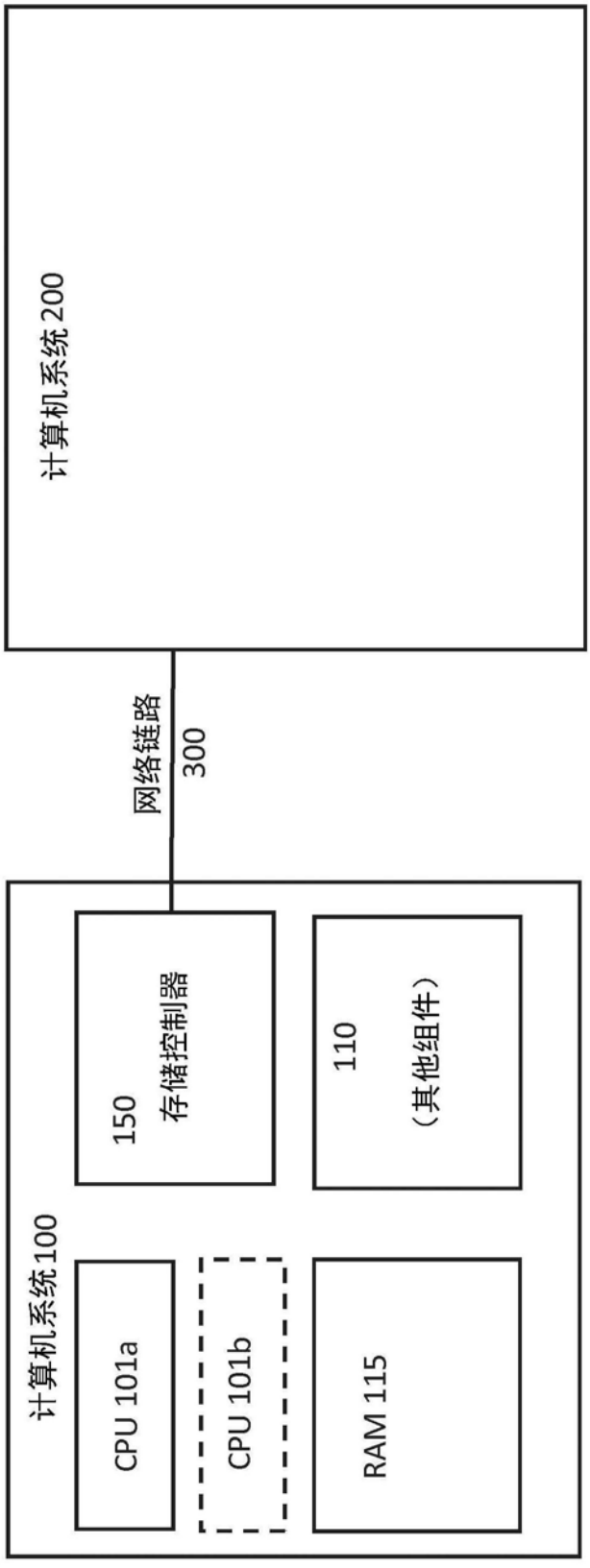

计算机系统 200
网络链路
300
计算机系统 100
150
存储控制器
110
（其他组件）
CPU 101a
CPU 101b
RAM 115

图1

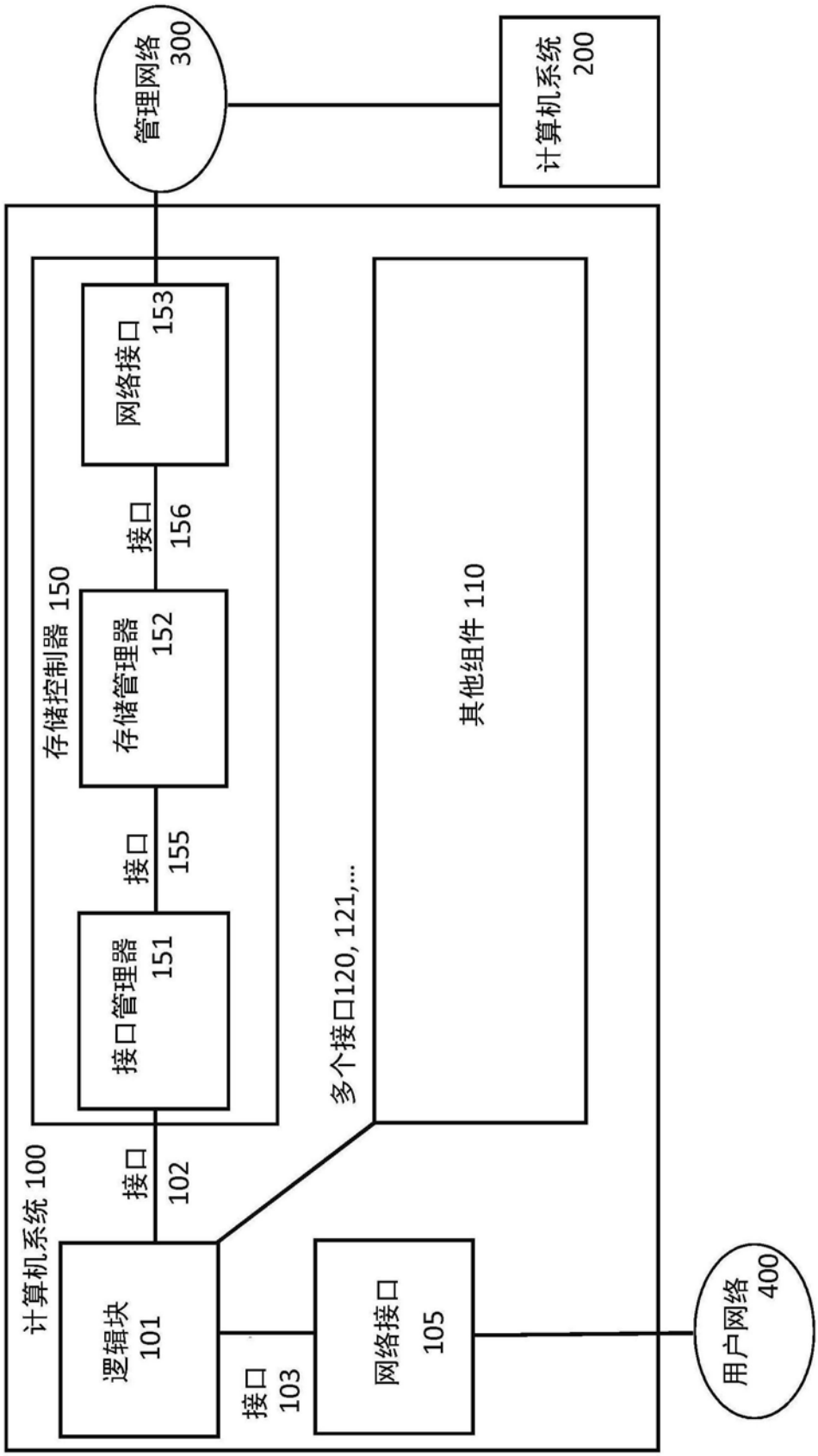

管理网络
300
计算机系统
200
网络接口
153
接口
156
存储控制器 150
存储管理器
152
接口
155
接口管理器
151
其他组件 110
多个接口120, 121,...
计算机系统 100
接口
102
逻辑块
101
接口
103
网络接口
105
用户网络
400

图2

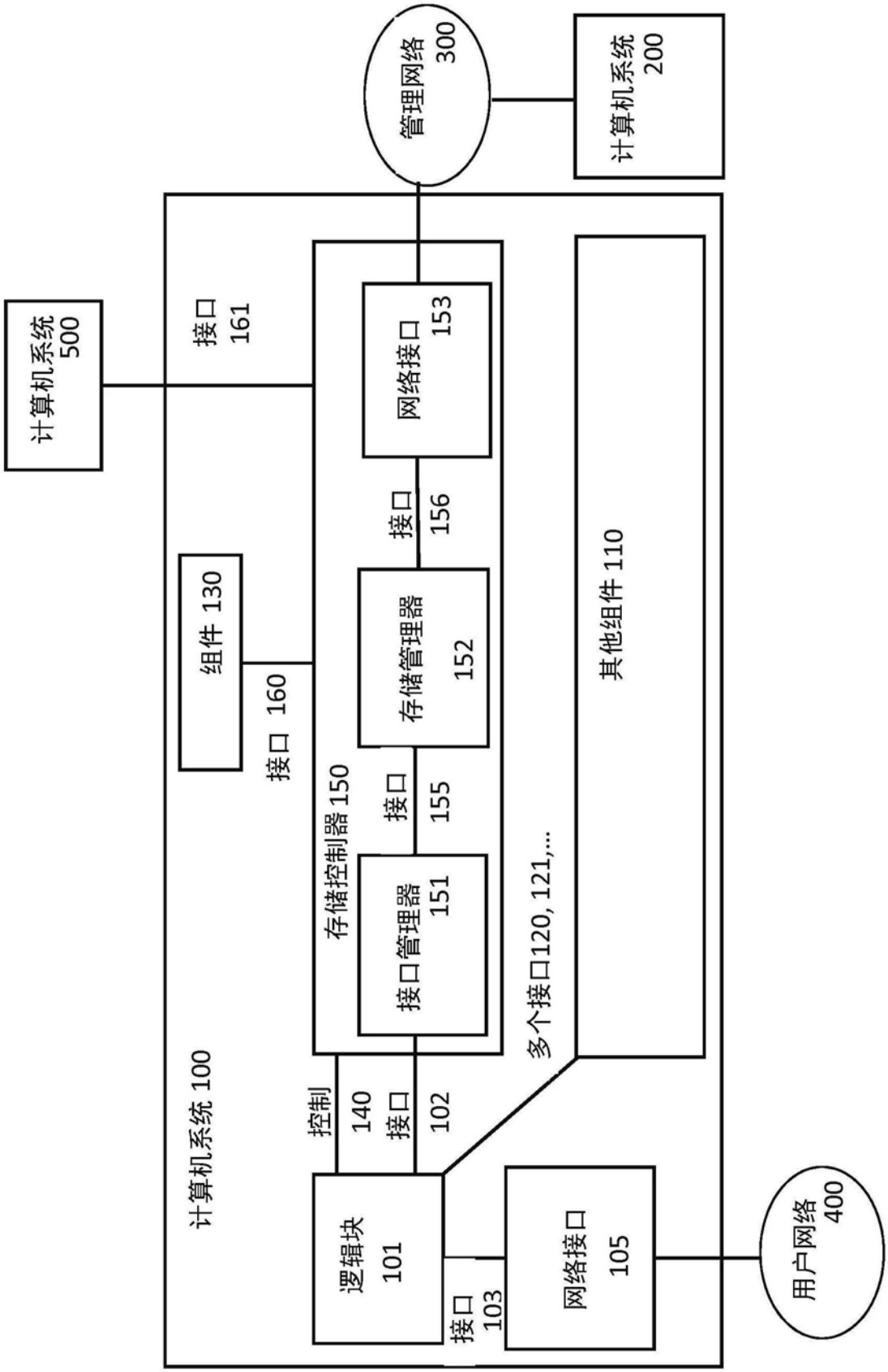

管理网络 300
计算机系统 200
计算机系统 500
接口 161
网络接口 153
接口 156
组件 130
接口 160
存储管理器 152
其他组件 110
存储控制器 150
接口 155
接口管理器 151
多个接口120, 121,...
控制 140 接口 102
逻辑块 101
接口 103
网络接口 105
用户网络 400
计算机系统 100

图3

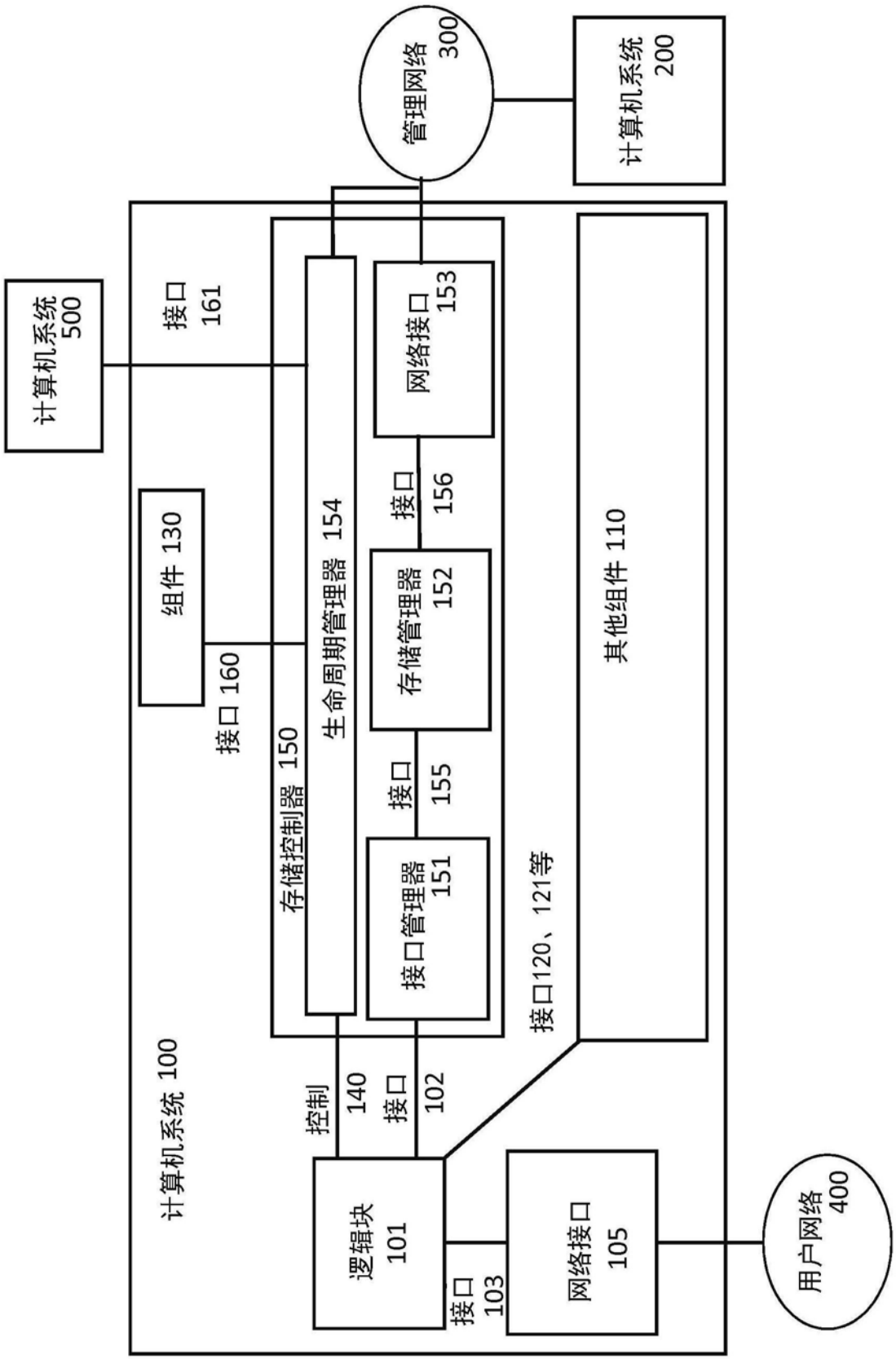

管理网络
300
计算机系统
200
计算机系统
500
接口
161
网络接口
153
组件 130
接口 160
存储控制器 150
生命周期管理器 154
接口
156
存储管理器
152
其他组件 110
接口
155
接口管理器
151
接口120、121等
计算机系统 100
控制
140
接口
102
逻辑块
101
接口
103
网络接口
105
用户网络
400

图4

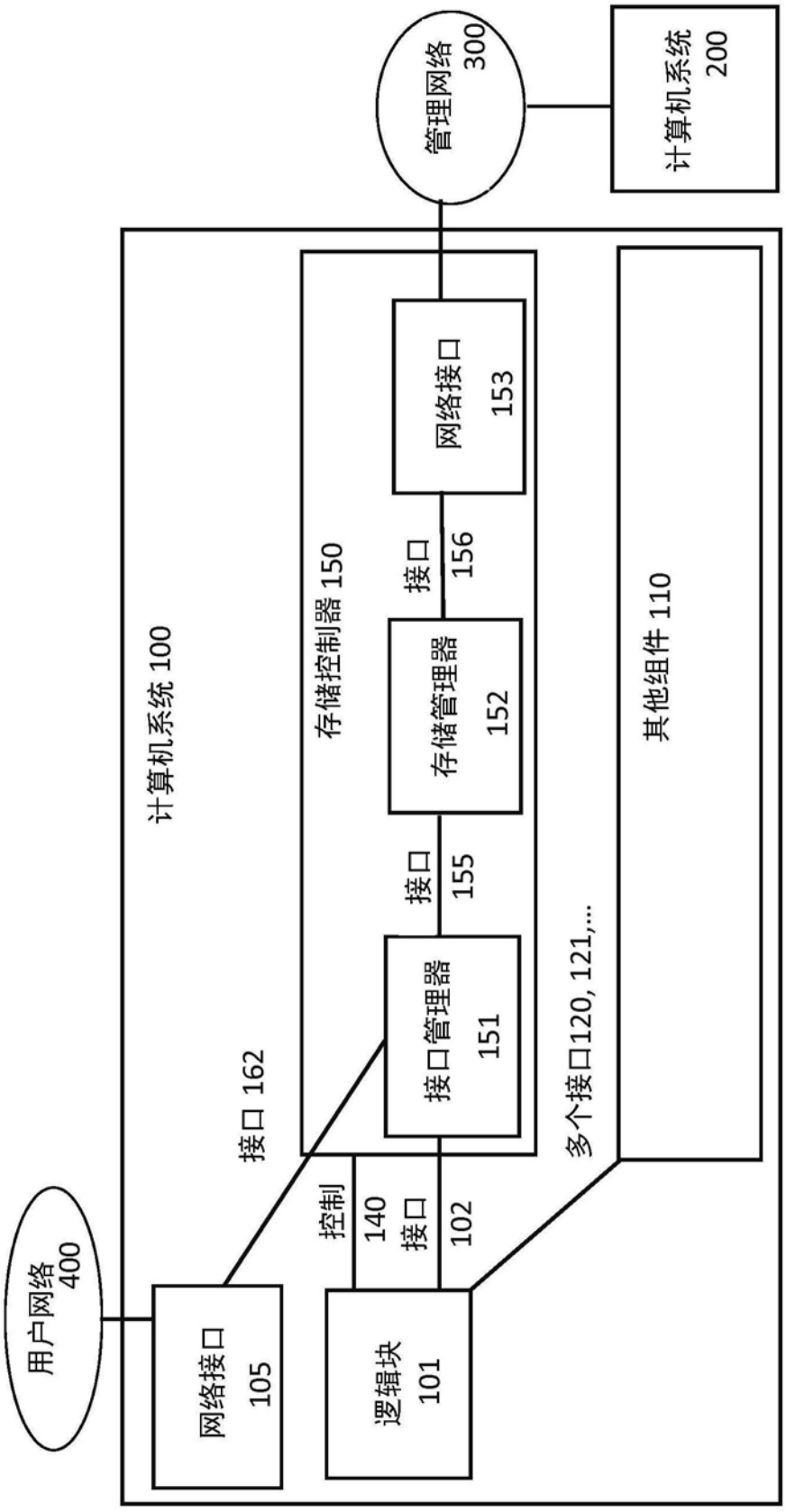

管理网络 300
计算机系统 200
计算机系统 100
存储控制器 150
网络接口 153
接口 156
存储管理器 152
接口 155
接口管理器 151
其他组件 110
接口 162
多个接口120, 121,...
控制 140
接口 102
用户网络 400
网络接口 105
逻辑块 101

图5

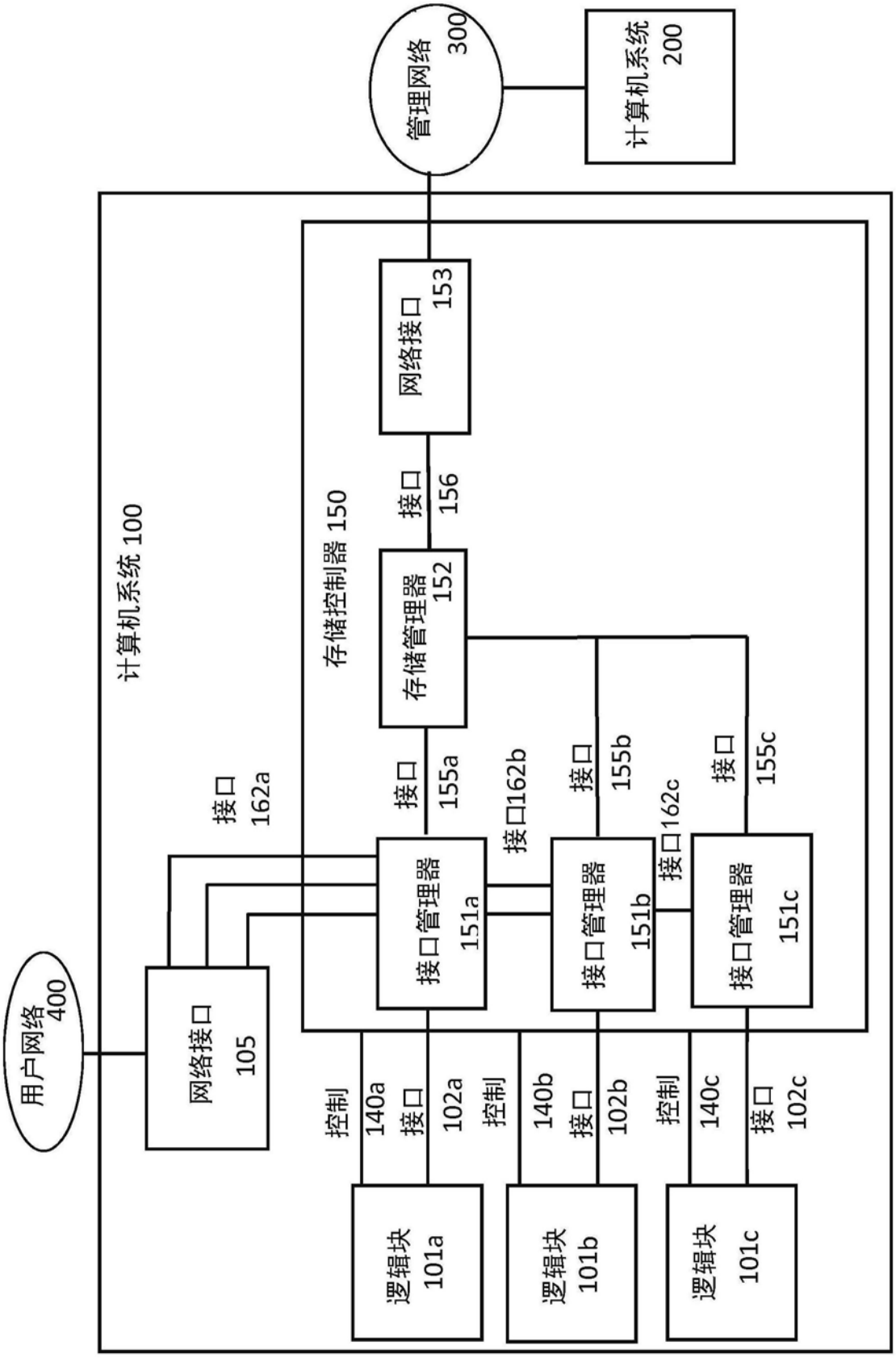

管理网络 300
计算机系统 200
计算机系统 100
存储控制器 150
网络接口 153
接口 156
存储管理器 152
接口 155a
接口162b
接口 155b
接口162c
接口 155c
接口 162a
接口管理器 151a
接口管理器 151b
接口管理器 151c
用户网络 400
网络接口 105
控制 140a
接口 102a
控制 140b
接口 102b
控制 140c
接口 102c
逻辑块 101a
逻辑块 101b
逻辑块 101c

图6

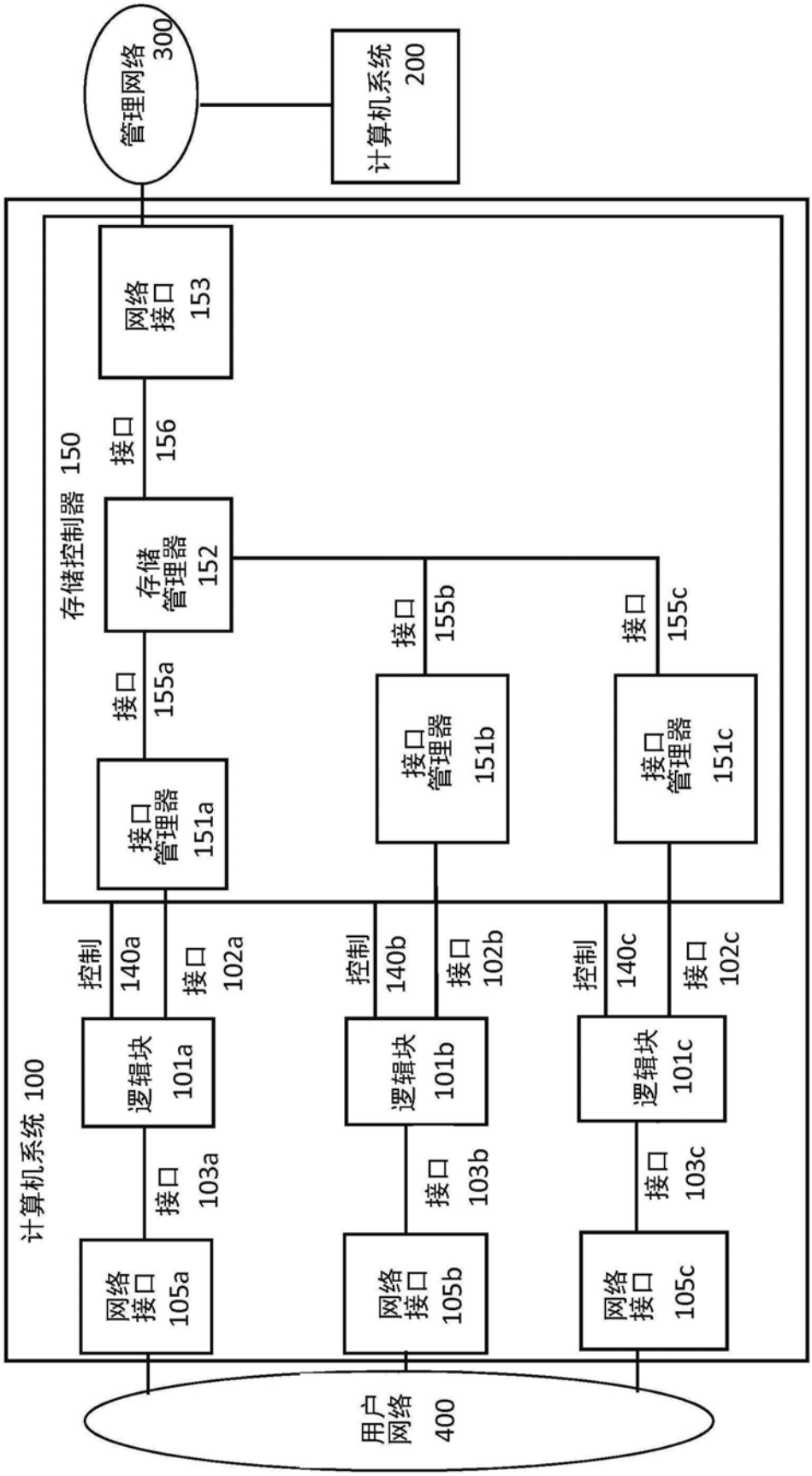

管理网络 300
计算机系统 200
计算机系统 100
存储控制器 150
网络接口 153
接口 156
存储管理器 152
接口 155a
接口 155b
接口 155c
接口管理器 151a
接口管理器 151b
接口管理器 151c
控制 140a
接口 102a
控制 140b
接口 102b
控制 140c
接口 102c
逻辑块 101a
逻辑块 101b
逻辑块 101c
接口 103a
接口 103b
接口 103c
网络接口 105a
网络接口 105b
网络接口 105c
用户网络 400

图7