(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 102667722 B
(45)授权公告日 2016.10.19

(21)申请号 201080047900.1
(22)申请日 2010.08.23
(65)同一申请的已公布的文献号
申请公布号 CN 102667722 A
(43)申请公布日 2012.09.12
(30)优先权数据
0918501.8 2009.10.21 GB
(85)PCT国际申请进入国家阶段日
2012.04.23
(86)PCT国际申请的申请数据
PCT/GB2010/051388 2010.08.23
(87)PCT国际申请的公布数据
WO2011/048395 EN 2011.04.28
(73)专利权人 ARM有限公司
地址 英国剑桥
(72)发明人 迈克尔·约翰·威廉斯
斯图亚特·大卫·贝尔斯
(74)专利代理机构 北京东方亿思知识产权代理有限责任公司 11258
代理人 宋鹤
(51)Int.Cl.
G06F 9/50(2006.01)
(56)对比文件
US 6160734 A,2000.12.12,
US 2005183065 A1,2005.08.18,
US 2006026385 A1,2006.02.02,
US 2005132365 A1,2005.06.16,
CN 101490646 A,2009.07.22,
US 4253145 A,1981.02.24,
审查员 张霞

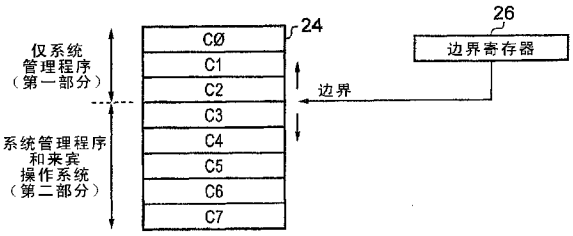
权利要求书4页 说明书11页 附图6页

(54)发明名称

数据处理系统中的硬件资源管理

(57)摘要

一种处理器(6)被提供,该处理器(6)具有多个硬件资源,诸如性能监视器(12)和上下文指针(18)。边界指示电路(14、20)储存边界值,该边界值是可编程的并且指示将硬件资源分割成第一部份和第二部份的边界位置。资源控制电路(16、22)控制对硬件资源的访问,以使得当程序执行电路(8)正执行第一程序时,其响应于关于所述多个硬件资源有多少的查询而传回第一值,而当程序执行电路正执行第二程序时,其通过传回对应于第二部份内的那些硬件资源的值来响应这样的查询。



CN 102667722 B

1.一种用于在多个程序控制下处理数据的装置，该装置包含：

多个硬件资源，这多个硬件资源具有预定的序列，以使得这多个硬件资源中的每一硬件资源具有在所述序列内的一预定位置；

边界指示电路，该边界指示电路被配置为储存指示在所述序列内的边界位置的边界值，所述边界位置将所述序列分割为处于所述序列内的所述边界位置的第一侧上的第一部分以及处于所述序列内的所述边界位置的第二侧上的第二部分；

程序指令执行电路，该程序指令执行电路被配置为执行程序指令；

资源控制电路，该资源控制电路被耦合至所述边界指示电路并且被配置为：

当所述程序指令执行电路正执行第一程序时，响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回第一值，其中所述第一值对应于所述第一部分和所述第二部分内的硬件资源的数量的总和；并且

当所述程序指令执行电路正执行第二程序时，响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述第二部分内的那些硬件资源的值。

2.如权利要求1所述的装置，其中

所述程序指令执行电路被配置为以多个权限等级中可选择的一个权限等级进行操作，所述多个权限等级内的不同权限等级对所述多个硬件资源具有不同访问权；并且

所述资源控制电路被配置为使得：

当所述程序指令执行电路正以运行第一程序的第一权限等级进行操作时，允许所述边界值在程序指令控制下被写；及

当所述程序指令执行电路正以运行第二程序的第二权限等级进行操作时，不允许所述边界值在程序指令控制下被写。

3.如权利要求2所述的装置，其中，所述资源控制电路被配置为使得：当所述程序指令执行电路以所述第二权限等级进行操作时，试图访问所述第一部分内的硬件资源的一个或多个程序指令被允许对所述第一部分内的硬件资源具有与当所述程序指令执行电路正以所述第一权限等级进行操作时相比更少的访问权。

4.如权利要求3所述的装置，其中，当所述程序指令执行电路以所述第二权限等级进行操作时，试图访问所述第一部分内的硬件资源的所述一个或多个程序指令被允许对所述第一部分内的硬件资源无访问权。

5.如权利要求1所述的装置，其中，当所述第一程序的一个或多个程序指令查询在所述装置所述多个硬件资源有多少时，传回对于所述装置内所述多个硬件资源有多少的真实值，并且所述第二程序的一个或多个程序指令查询所述装置内所述多个硬件资源有多少时，传回由所述边界值所指定的值。

6.如权利要求3所述的装置，其中，所述多个权限等级包含：权限等级的层次，所述层次中的所述第一权限等级高于所述层次内的所述第二权限等级和任何给定的权限等级，所述第一权限等级相比于所述层次中低于所述给定的等级的任何权限等级具有对所述硬件资源的适当超集的访问权。

7.如权利要求1所述的装置，其中，所述边界指示电路被配置为储存进一步的边界值，所述进一步的边界值将所述第二部分分割为进一步的第一部分和进一步的第二部分，所述资源控制电路被配置为：

当所述程序指令执行电路正以第二权限等级进行操作时，在程序指令控制下允许所述进一步的边界值被写入，并且响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述进一步的第一部分和所述进一步的第二部分内的那些硬件资源的总和的值；及

当所述程序指令执行电路正以第三权限等级进行操作时，在程序指令控制下不允许所述进一步的边界值被写入，响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述进一步的第二部分内的那些硬件资源的值。

8.如权利要求1所述的装置，其中，所述边界指示电路被配置为储存N个进一步的边界值，其中N是非零的正整数，所述N个进一步的边界值形成以所述边界值开始的边界值序列，所述边界值序列内的第n个边界值将所述边界值序列内的由第n-1个边界值所定义的第二部分分割为第n个第一部分和第n个第二部分，所述资源控制电路被配置为：

当所述程序指令执行电路正以针对所述第n-1个边界值的第二权限等级进行操作时，允许所述第n个边界值在程序指令控制下被写入，并且响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述第n个第一部分和所述第n个第二部分内的那些硬件资源的总和的值；及

当所述程序指令执行电路正以针对所述第n个边界值的第二权限等级进行操作时，不允许所述第n个边界值在程序指令控制下被写入，并且响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述第n个第二部分内的那些硬件资源的值。

9.如权利要求1所述的装置，其中，所述多个硬件资源包含：指向存储器管理单元的配置数据的指针表内的多个条目。

10.如权利要求9所述的装置，其中，所述配置数据包含如下内容中的一者或多者：

转换表基指针寄存器值；

故障地址和上下文寄存器值；

转换后备缓冲器维持寄存器值；

虚拟地址到物理地址运算寄存器值；

存储器管理单元配置和控制寄存器值；

旁路属性寄存器值；以及

用于映射数据流的交易识别寄存器值。

11.如权利要求1所述的装置，其中，所述多个硬件资源包含：多个性能监视电路。

12.如权利要求11所述的装置，其中，所述多个性能监视电路包含如下内容中的一者或多者：

多个计数器，这多个计数器包含：一个或多个计数器寄存器、使能寄存器、溢出状态标志以及溢出事件中断使能寄存器；以及

事件选择电路，该事件选择电路用于从多个事件中选择用以计数的事件。

13.如权利要求11所述的装置，包含如下内容中一者或多者：

全局使能寄存器，该全局使能寄存器用于选择性地使能所述多个性能监视电路；

阴影全局使能寄存器以及架构全局使能全局使能寄存器，所述阴影全局使能寄存器可访问所述第一程序以选择性地使能所述第一部分内的所述多个性能监视电路，所述架构全

局使能缓存器可访问所述第一程序和所述第二程序以选择性地使能所述第二部分内的所述多个性能监视电路。

14.如权利要求1所述的装置，其中，所述多个硬件资源包含：作为整体被访问并且为配置寄存器和状态寄存器中之一的寄存器的多个字段，所述多个字段中的每个字段表示各自的进一步资源，所述边界值指定所述第一程序可访问所述多个字段中的哪个字段以及所述第二程序可访问所述多个字段中的哪个字段。

15.如权利要求1所述的装置，其中，所述多个硬件资源包含：对应于多个性能监视电路中的各性能监视电路的配置寄存器内的一个或多个比特的字段，各字段控制所述多个性能监视电路中的相应性能监视电路是否被使能。

16.如权利要求1所述的装置，其中，所述多个硬件资源包含：供所述第一程序或所述第二程序使用的多个寄存器。

17.如权利要求1所述的装置，其中，所述多个硬件资源包含：多个映射寄存器，用于储存将被使用的在被访问的数据流和所述装置的相关处理上下文之间的映射。

18.如权利要求1所述的装置，其中，所述多个硬件资源包含：多个断点寄存器和多个观察点寄存器中的至少一者。

19.如权利要求1所述的装置，其中，如果所述第二程序试图读取所述第二部分内的一硬件资源，则针对该硬件资源的默认值被传回。

20.如权利要求1所述的装置，其中，如果所述第二程序试图写入所述第二部分内的一硬件资源，则所述写入不被执行。

21.如权利要求1所述的装置，其中，如果所述第二程序试图访问所述第二部分内的一硬件资源，则所述访问被放弃并且异常处理被调用。

22.如权利要求2所述的装置，其中，所述多个权限等级包含：管理程序权限等级，系统管理程序以该系统管理程序权限等级运行；操作系统权限等级，操作系统程序以该操作系统权限等级运行，所述系统管理程序访问所述第一部分内的那些硬件资源和所述第二部分内的那些硬件资源，并且所述操作系统程序访问所述第二部分内的那些硬件资源。

23.如权利要求2所述的装置，其中，所述多个权限等级包含：至少一个安全域内的安全权限等级，安全程序以该安全权限等级运行；以及至少一个非安全域内的非安全权限等级，非安全程序以该非安全权限等级运行，所述安全程序访问所述第一部分内的那些硬件资源和所俗话第二部分内的那些硬件资源，并且所述非安全程序访问所述第二部分内的那些硬件资源。

24.如权利要求1所述的装置，其中，所述资源控制电路根据所述边界值重新标引所述多个硬件资源。

25.一种用于利用装置在多个程序的控制下处理数据的方法，其中，所述装置具有多个硬件资源，这多个硬件资源具有预定的序列，以使得这多个硬件资源中的每一硬件资源具有在所述序列内的一预定位置，所述方法包含以下步骤：

储存指示在所述序列内的边界位置的边界值，所述边界位置将所述序列分割为所述序列内的所述边界位置的第一侧上的一第一部分以及所述序列内的所述边界位置的第二侧上的第二部分；

当程序执行电路执行第一程序时，响应于查询所述装置内所述多个硬件资源有多少的

一个或多个程序指令，传回第一值，其中所述第一值对应于所述第一部分和所述第二部分内的硬件资源的数量的总和；并且

当所述程序指令执行电路执行第二程序时，响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述第二部分内的那些硬件资源的值。

数据处理系统中的硬件资源管理

技术领域

[0001] 本发明涉及数据处理系统领域。更具体而言，本发明涉及数据处理系统内的硬件资源的管理。

背景技术

[0002] 数据处理系统通常具有很多硬件资源，诸如性能监视器计数器、配置储存寄存器、调试事件产生资源、追踪(trace)资源等等。在特定处理器内提供的这些资源的数目可特定于具体实施方式。

[0003] 在数据处理系统领域内，朝向使用虚拟化的趋势与日俱增。这样的虚拟化可允许软件被提供一运行环境，该运行环境看起来具有就软件角度而言的形式，然而实际上其具有不同的底层(underlying)实体形式。举例而言，处理器可被提供为运行负责管理虚拟化的系统管理程序(hypervisor)软件以及通常在系统管理程序软件权限等级以下的权限等级运行的一个或多个来宾操作系统。来宾操作系统可由系统管理程序软件呈现一通向处理器的接口，诸如对可用特定硬件资源的数量目的指示，其不同于物理事实。此虚拟化通常由系统管理程序软件管理以捕捉(trapping)对软件资源的访问，而后运行系统管理程序以管理对底层来宾操作系统的响应，以给予与被呈现给来宾操作系统的虚拟系统相匹配的适当响应。捕捉访问和在系统管理程序控制下产生适当的软件响应的处理会消耗数百个处理器周期并且且运行相对较慢。在一半虚拟化(paravirtual)方法中，来宾操作系统可具有与系统管理程序相同的权限特级，并且被“信任”为除非藉由对系统管理程序的已发布接口的适当呼叫否则不使用某些资源。

[0004] 希望被虚拟化的某些硬件资源不适于捕捉访问并产生软件的适当响应的方法。举例而言，在处理器内提供硬件性能计数器以监视系统性能的各方面变得越来越有用。系统管理程序可使用这样的性能计数器来控制诸如电压和频率尺度之类的参数，以减少处理器的能量使用并同时仍满足所要求的性能标准。在另一等级，来宾操作系统或使用来宾操作系统运行的应用程序可利用硬件性能计数器来控制其自己的操作或作为该应用软件或来宾操作系统开发期间所执行的诊断/调试操作的部份。在这样的性能计数器的上下文(context)中，捕捉对性能计数器的访问以支持这些硬件资源的虚拟化的开销大大影响被监视的软件的性能和行为。并从而影响所产生的结果的有效性。还希望能同时支持用于高等级控制的系统管理程序和用于不同目的的诸如来宾操作系统等其它程序二者对硬件资源的使用。此外，在实体实施中所提供的硬件资源的数目可改变。

[0005] 需要在虚拟化系统内管理的硬件资源的另一示例为用于指向系统存储器管理单元的配置数据之上下文指针。在支持安全域和非安全域(例如ARM Trustzone)的系统中，通常需要根据被呈现给正运行的软件的环境来交换硬件存储器管理单元的配置。指示配置数据储存于何处的硬件指针提供用于快速访问此数据以便于保存和重新存储的目的的机制。因此，捕捉对于这些指针的软件访问对切换上下文的速度有不利的影响。然而，重要的是，该系统应能够提供适当环境，包含这些指针的数目以及支持可改变指针数目的不同实体实

施的能力。

发明内容

[0006] 根据本发明的一方面，提供一种用于在多个程序控制下处理数据的装置，该装置包含：多个硬件资源，这多个硬件资源具有预定的序列，以使得这多个硬件资源中的每一硬件资源具有所述序列内的一预定位置；边界指示电路，该边界指示电路被配置为储存指示在所述序列内的边界位置的边界值，所述边界位置将所述序列分割为处于所述序列内的所述边界位置的第一侧上的第一部份以及处于所述序列内的所述边界位置的第二侧上的第二部份；程序指令执行电路，该程序指令执行电路被配置为执行程序指令；资源控制电路，该资源控制电路被耦合至所述边界指示电路并且被配置为：当所述程序执行电路正执行第一程序时，响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回一第一值；并且当所述程序执行电路正执行第二程序时，响应于查询所述该装置所述多个硬件资源有多少的一个或多个程序指令，传回对应于所述第二部份的那些硬件资源的值。

[0007] 本发明技术提供按照可编程边界值所指示将多个硬件资源分割为第一部份和第二部份的硬件支持。资源控制电路被提供，以使得当第一程序查询所述多个硬件资源有多少时，传回第一值，然而当第二程序查询所述多个硬件资源有多少时，传回对应于所述第二部份内的那些硬件资源的值。因此，第一程序和第二程序可被提供关于现有的硬件资源的数目的不同观点。第一程序和第二程序可查询此数目，以考虑在不同实施中可以有不同数目的硬件资源的事实。此外，针对此查询对第一程序和第二程序传回的不同结果允许第一程序和第二程序可做出硬件资源使用之间的分隔。

[0008] 举例而言，传回第一程序的对资源数目的第一值对应于在所述第一部份和所述第二部份内的硬件资源的总和。第一程序因此可被允许控制所有的硬件资源，然而第二程序仅可感知所述第二部份内的那些硬件资源并且将其互动限制到所述第二部份内的那些硬件资源。在一替代性实例中，传回第一程序的第一值可对应于所述第一部份内的硬件资源，这给予第一程序对这些硬件资源的访问。

[0009] 第一程序和第二程序可以程序指令执行电路所支持的多个权限等级内的不同权限等级进行操作。在此环境中，资源控制电路可被配置为使得以第一权限等级运行的第一程序被允许在程序指令控制下写入边界值，然而以第二权限等级运行的第二程序不被允许写入边界值。以此方式，第一程序可控制呈现给第二程序的硬件资源的示图(view)。资源控制电路可被配置为当该系统处于第二权限等级时以不同的方式响应对第一部份内的那些硬件资源的访问。一般而言，相比于程序执行电路以第一权限等级进行操作时，当以第二权限等级进行操作时，资源控制电路可给予对所述第一部份内的那些硬件资源较少的访问权。就此的一个示例是当程序执行电路以第二权限等级进行操作时，不被给予对所述第一部份内的硬件资源的访问权。

[0010] 传回第一程序或第二程序的对关于有多少硬件资源的查询的精确响应可改变。在一实施例中，可向第一程序传回存在的硬件资源的数目的真实值，并且向第二程序传回由边界值所指定的值。

[0011] 多个权限等级可包含：权限等级层次，该层中的至少一些较高的权限等级相比于该层次中较低的权限等级具有对硬件资源的适当超集(superset)的访问权。

[0012] 在前述中，已描述了结合有资源控制电路响应的边界值的系统。在进一步的实施例中，边界指示电路可被配置为储存进一步的边界值，所述进一步的边界值将所述第二部份分割为进一步的第一部份和进一步的第二部份。当以第二权限等级进行操作中，程序将被允许写入该进一步的边界值并且以对应于进一步的第一部份和进一步的第二部份内的那些硬件资源的总和的值来响应关于有多少硬件资源的查询。同时，以第三权限等级进行操作的程序将不被允许写入该进一步的边界值，并且将传回对应于进一步的第二部份内的那些硬件资源的值来响应关于有多少硬件资源的查询。

[0013] 因此，可见，硬件资源被分割为第一部份和第二部份可具有第二部份被进一步分割为进一步的第一部份和进一步的第二部份的层次。以类似的方式，该进一步的第二部份等等可藉由另外的边界值被进行进一步的子分割。

[0014] 虽然可理解由本发明技术所管理的硬件资源可采取各种不同的形式，但是本发明技术适用于其中硬件资源包含指向存储器管理单元的配置数据的指针表内的多个条目的实施例。

[0015] 在这些实施例中，配置数据可包含：一个或多个转换表基指针寄存器值；故障地址和上下文寄存器值；转换查考缓冲区维持寄存器值(translation look aside buffer maintenance register values)；虚拟地址大盘物理地址运算寄存器值；存储器管理单元配置和控制寄存器值；上下文库值，该上下文库值包含：转换表基指针寄存器值、故障地址和上下文寄存器值、转换查考缓冲区维持寄存器值、虚拟地址到物理地址运算寄存器值和存储器管理单元配置和控制寄存器值；旁路属性寄存器；响应于由第一程序和第二程序中的一个或多个发起的活动而引起的中断输出；及用于映映射数据流的交易识别寄存器值。

[0016] 在其它适用于本发明技术的实施例中，多个硬件资源可包含多个性能监视电路。这些性能监视电路可包含如下内容中的一者或多者：多个计数器，这多个计数器包含：一个或多个计数器寄存器、使能寄存器、溢出状态标志以及溢出事件中断使能寄存器；以及事件选择电路，该事件选择电路用于从多个事件中选择用以计数的事件。

[0017] 本发明技术可应用的多个硬件资源的又一示例是如下示例，多个硬件资源包含：作为整体被访问并且为配置寄存器和状态寄存器中之一的寄存器的多个字段，所述多个字段中的每个字段表示各自的进一步资源，所述边界值指定所述第一程序可访问所述多个字段中的哪个字段以及所述第二程序可访问所述多个字段中的哪个字段。

[0018] 在此上下文中，多个硬件资源可包含：对应于性能监视电路中的各性能监视电路的配置寄存器内的比特，每一比特控制相应性能监视电路的某方面，诸如该电路是否被使能。这也可被应用于状态寄存器，此时每一比特报告而非控制相应电路的某方面。可具有多对一的比特映射，例如针对每个电路的比特为一个字段。

[0019] 本发明技术可应用的多个硬件资源的又一示例为：多个硬件资源包含供第一程序或第二程序使用的多个寄存器。举例而言，这些寄存器可为映射寄存器，储存将被使用的在被访问的数据流和该装置所关心的相关处理上下文之间的映射。

[0020] 如果第二程序试图读取第一部份内的某硬件资源(例如，其不具有适当访问的部份)，则在一些实施例中，针对此硬件资源的默认值可被传回。以类似的方式，如果第二程序试图对第一部份内的某硬件资源写入，则该写入将不被执行，并且异常(故障)处理可被触发。

[0021] 前面所讨论的多个不同权限等级可具有各种不同的形式和用途。在本发明技术可使用的一示例中，这多个权限等级包含：系统管理程序权限等级，系统管理程序以该系统管理程序权限等级运行虚拟化的底层硬件；以及操作系统权限等级，操作系统程序以该操作系统权限等级运行。系统管理程序访问第一部份内的那些硬件资源和第二部份内的那些硬件资源，并且操作系统程序仅访问第二部份内的那些资源。

[0022] 本发明技术可在其中被使用的另一示例环境是如下环境，其中，多个不同的权限等级包含：安全程序在其中运行的安全域内的至少一个安全权限等级，以及非安全程序在其中运行的非安全域内的至少一个非安全权限等级。在此上下文中，安全程序可访问第一部份内的那些硬件资源以及第二部份内的那些硬件资源，然而非安全程序仅访问第二部份内的那些硬件资源。

[0023] 资源控制电路以及对硬件资源的门控访问(gating access)可根据边界值重新标引多个硬件资源。以此方式，不同的程序能够就好像他们从索引值序列内的某设定的索引值开始而访问资源，即使那些索引被映射的物理资源根据哪部份被访问以及第一位置和第二位置之间的边界的位置而改变。

[0024] 根据本发明的另一方面，提供：多个硬件资源构件，这多个硬件资源构件具有预定的序列，以使得这多个硬件资源构件具有所述序列内的一预定位置；边界指示构件，该边界指示构建用于储存指示在所述序列内的边界位置的边界值，所述边界位置将所述序列分割为处于所述序列内的所述边界位置的第一侧上的第一部份和处于所述序列内的所述边界位置的第二侧上的第二部份；程序指令执行构件，该程序指令执行构件用于执行程序指令；资源控制构件，该资源控制构件被耦合至边界指示构件，用于：当程序指示构件正执行第一程序时，响应于查询所述装置内所述多个硬件资源构件有多少的一个或多个程序指令，传回第一值；并且当程序执行构件正执行第二程序时，响应于查询所述装置内所述多个硬件资源构件有多少的一个或多个程序指令，传回对应于第二部份内的那些硬件资源的值。

[0025] 根据本发明的又一方面，提供：储存指示在所述序列内的边界位置的边界值，所述边界位置将所述序列分割为处于所述序列内的所述边界位置的第一侧上的第一部份和处于所述序列内的所述边界位置的第二侧上的第二部份；当程序执行电路正执行第一程序时，响应于查询所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回第一值；并且当程序执行电路正执行第二程序时，响应于查询在所述装置内所述多个硬件资源有多少的一个或多个程序指令，传回对应于第二部份内的那些硬件资源的值。

附图说明

[0026] 现在将参考附图借助于示例来描述本发明的实施例，在附图中：

[0027] 图1示意性地图示用于在程序指令的控制下处理数据并且包含有不同类型的硬件资源的装置；

[0028] 图2示意性地图示多个硬件资源，这多个硬件资源在由边界寄存器内所储存的边界值所指定的边界处被分割成第一部份和第二部份；

[0029] 图3示意性地图示多个资源，这多个资源为寄存器内的控制比特并且由边界寄存器内所保持的边界值被分割成第一部份和第二部份；

[0030] 图4图示以不同权限等级操作的不同程序的层次；

[0031] 图5图示在具有安全域和非安全域的系统中的权限等级的另一布置；

[0032] 图6是图示对第一部份内的硬件资源访问的控制的流程图；

[0033] 图7图示使用一边界和另一边界对硬件资源的分割；

[0034] 图8示意性地图示多个边界，这多个边界使用由不同权限等级的程序所写的各边界值来分割多个资源；

[0035] 图9图示具有指向存储器管理单元配置的储存位置的指针的表格形式的多个硬件资源；

[0036] 图10示意性地图示用于读取资源大小指示的电路；

[0037] 图11示意性地图示用于当本发明水印技术(watermarking technique)被利用时用于读取资源的电路；

[0038] 图12示意性地图示用于当本发明水印技术被利用时用于写入资源的电路。

具体实施方式

[0039] 图1示意性地图示用于在多个程序控制下处理数据的装置2。这些程序被储存在存储器4内，存储器4被耦合至片上系统(system on chip)集成电路6。集成电路6包括处理器核心8，处理器核心8执行从存储器4所读取的程序指令。集成电路6还包括系统存储器管理单元(MMU)10，MMU 10用于在系统等级控制例如经由连接其他组件的ARM AXI总线(未示出)对存储器4的访问。集成电路6包括根据下面进一步描述的技术的可用于以不同权限等级运行的不同程序的硬件资源。

[0040] 在本示例中，硬件资源的一种形式包括具有性能计数器形式的性能监视电路12。与这些性能监视电路12(对应于多个硬件资源)相关联，提供有性能监视器边界寄存器14(边界指示电路)和性能监视器控制电路16(资源控制电路)。性能监视器边界寄存器14储存指示性能监视电路12的第一部份内的性能计数器和性能监视电路12的第二部份内的性能计数器之间的分割的边界值。性能监视电路内的性能计数器具有预定的序列，从而使得每个硬件资源在该序列内具有预定位置。因此，通过在该序列内的某点定义边界，可确定哪些性能计数器落于该边界的一侧的第一部份内，以及哪些性能监视器落于该边界的另一侧的第二部份内。

[0041] 性能监视器控制电路16被提供指示处理器核心8正操作和执行处理指令的当前权限等级的信号。性能监视器控制电路16响应于此权限等级来确定当由处理器核心8执行的程序试图访问硬件资源(例如性能监视器或用于性能监视器的控制开关)中的一个硬件资源时是否允许该访问。若允许此访问，则允许例如进行写入或读取。若不允许此访问，则该访问将不被执行，并且可执行默认动作，例如传回一默认值、触发异常(exception)处理、允许读取但不允许写入或简单地忽视写入。

[0042] 图1还图示出另一形式的包含多个上下文指针18的硬件资源。这些上下文指针储存指示用于编程系统MMU 10的上下文数据就其当前上下文/模式被储存于存储器4内的地址值。因此，当上下文/模式改变时，对系统MMU 10的配置数据可快速地被撷取到MMU 10，并且当前的数据被储存至存储器4。上下文/模式可对应于处理器核心8当前是在ARM Trustzone系统的安全域还是非安全域操作。

[0043] 与多个上下文指针(具有预定序列(例如指针编号)的硬件资源)相关联，提供有上

下文指针边界寄存器20，上下文指针边界寄存器20储存将上下文指针分割成一第一部份和第二部份的边界值；以及上下文指针控制电路22，上下文指针控制电路22响应于处理器核心8正在其中操作的域来通过程序指令来控制对上下文指针18的访问。如前文关于性能监视电路12所描述的，上下文指针18在他们的预定序列内通过下文指针边界寄存器20内储存的上下文指针边界值被分割成第一部份和第二部份。响应上下文指针的控制逻辑对查询有多少硬件资源的运行第一程序指令作出响应，以传回第一值，该第一值可为第一部份内的上下文指针的数目或第一部份和第二部份内的上下文指针的总和。访问第一部份的第一程序被允许在上下文指针边界寄存器20内写入边界值。访问上下文指针18的第二部份的第二程序不被允许写入上下文指针边界值，并且当其查询存在的上下文指针的数目时，对应于来自第二部份的上下文指针的数目的值被传回。

[0044] 图2示意性地图示多个硬件资源C0到C7。这些硬件资源24具有对应于各自在这多个硬件资源内的数值位置的预定序列。储存在边界寄存器26内的边界值指示此序列内的边界位置。该边界值是一变量，并且相应地边界的位置可通过向边界寄存器26写入而被改变。

[0045] 在所图示出的示例中，硬件资源的第一部份包括寄存器C3至C7。硬件资源的第二部份对应于寄存器C0到C2。硬件资源的第一部份仅可由在安全域中运行的安全程序访问。硬件资源的第二部份可由在安全域中运行的安全程序和在非安全域中运行的非安全程序访问。如本领域技术人所熟悉的，安全程序通常负责对集成电路6的底层硬件架构至少部份地虚拟化并提供对非安全程序的虚拟硬件接口，以便以受保护的方式对非安全软件提供服务。

[0046] 安全程序和非安全系统二者需要查询可用硬件资源24的数目，因为这可随着实施情况而改变。然而，根据本发明的技术，响应于此查询而传回的值将根据正在发出查询的是在安全域中运行的安全程序还是在非安全域中操作的非安全程序而改变。安全程序将被传回存在的资源的真实总数目，而非安全程序将被传回对应于第二部份内存在的那些硬件资源的数目。以此方式，第一部份内的硬件资源可相对非安全程序被隐藏，并被预留以供安全程序使用。

[0047] 由储存于边界寄存器26内的边界值所控制的边界的位置可通过向边界寄存器26写入边界值来改变。安全程序可向边界寄存器26写入。非安域中的非安全程序或其它程序不允许向边界寄存器26写入。

[0048] 系统MMU 10按照类似于处理器MMU为处理器产生的访问提供地址转换和保护服务的方式来为设备产生的访问提供地址转换和保护服务。与系统MMU的主要差别在于所支持的同时发生的转换配置的数目。

[0049] 处理器MMU一般仅支持一个活动上下文，因为其一次支持对软件的一个“世界”的转换。类似于系统管理程序的管理性软件当对来宾操作系统的运行进行时间划片时，其将进行布置以切换此状态，并且当对应用的运行进行时间划片时，每个来宾操作系统将执行等同的操作。

[0050] 系统MMU 10支持对在短时间内/同时地来自多个设备的访问进行转换，这潜在地提供了对每个设备的单独转换/保护。

[0051] 使用运用ARM TrustZone技术的系统作为示例，系统MMU 10可支持对来自两组设备的交易进行转换：

[0052] 1.由安全软件配置的那些设备

[0053] 2.由非安全软件配置的那些设备

[0054] 转换交易的处理需要配置转换处理的一些上下文(例如转换表基指针寄存器、转换表控制寄存器)。在系统MMU 10下,每个不同组所希望的转换将需要单独的转换上下文。因而,系统MMU 10将提供一转换上下文池,并且希望找到系统MMU 10可以简单方式在安全和非安全软件之间共享这些上下文的方式。

[0055] 与利用ARM TrustZone技术的系统的安全性方法相一致,非安全软件不应能够观察或影响具有安全性的转换上下文。此外,一旦安全软件需要使用由系统MMU提供的服务,非安全软件必须不能请求所有的转换上下文。所提出的解决方案是实现边界寄存器,边界寄存器将转换上下文池分为二组:一组用于安全软件,而一组用于非安全软件。此方法具有为非安全软件提供对上下文池的适当部份的直接访问而不会有安全软件干预的优点。

[0056] 此系统MMU示例的另外的优点是:根据资源驻留于第一部分还是第二部份,希望资源有不同行为。例如,驻留于第一部份的转换上下文可具有与驻留于第二部份的转换上下文不同的格式。此格式选择可根据边界值做出,从而使得资源被设计为能够支持二种格式并且基于他们落入第一部分还是第二部份来选择他们呈现何种格式。

[0057] 在系统MMU 10中,使用边界值将转换上下文划分为两个群组(安全、非安全),非安全群组中的转换上下文使用进一步的边界值被划分为两个部份。这些部份提供第一阶段转换上下文格式或第二阶段转换上下文格式,这些转换上下文格式分别提供从虚拟地址至中间物理地址及从中间物理地址至物理地址的转换。第一阶段转换上下文格式被提供以供来宾操作系统使用;第二阶段转换上下文格式被提供用于系统管理程序。另外,系统MMU 10可被配置用于第一阶段转换的输出馈入第二阶段转换的输入的嵌套(nested)转换,因此在虚拟化系统的上下文中执行从虚拟地址至物理地址的地址转换。

[0058] 系统管理程序可允许虚拟化的来宾操作系统对非安全群组中的执行第一阶段转换的部份的访问;执行第二阶段转换的部份对于来宾操作系统而言是不能访问的。

[0059] 图3图示可根据本发明技术管理的多个硬件资源的另一示例。在此示例中,多个硬件资源包括作为整体被访问(即,作为整体被读取或写入)的控制寄存器内的控制比特。图3中所示的寄存器内的每一比特用于控制相对应的性能监视器(例如,性能计数器电路)是被使能还是被禁用。储存于边界寄存器28内的边界值将性能监视使能寄存器30内的比特分割为第一部份和第二部份。第一部份内的使能比特可被系统管理程序进行读取和写入访问,但是不能被来宾操作系统有效地访问。作为当访问不被允许时的默认动作的示例,来宾操作系统对这些比特中的某比特的写入可被忽略,并且如果来宾操作系统读取这些比特中某比特,则“0”值被传回,而不论由系统管理程序对相关比特设定的实际值如何(或实际值可被传回)。使能寄存器30的第二部份可被系统管理程序和来宾操作系统二者进行读取和写入访问。

[0060] 在本示例中,应了解,性能监视器使能寄存器30内的使能比特是在储存于边界寄存器28内的边界值的控制下被授权选择性访问的硬件资源。底层性能计数器也是允许所选访问的硬件资源。

[0061] 这些技术的另一可能特征为提供了全局使能寄存器(诸如用于硬件特征(例如切换动态电压标定(scaling))的全局使能寄存器)并且当存在资源供应的分割时提供了这些

增益阴影(gain shadow)值。因此，替代单个全局使能寄存器(其为呈现给来宾操作系统的虚拟机的程序员模型)，可有以下二个：控制第一部份的第一全局使能寄存器；及控制第二部份的第二全局使能寄存器。第二使能寄存器对应于虚拟机的全局使能。

[0062] 图4示意性地图示可由处理器6执行的多个程序的布置。系统管理程序32负责处理器6对二个底层来宾操作系统34、36的虚拟化。这些来宾操作系统34、36的每一者被提供有到由系统管理程序32所管理的处理器6的接口，以给予来宾操作系统34、36对应于不是实体处理器6的真实实体形式的虚拟处理器的环境。例如，处理器6可具有并控制比来宾操作系统34、36允许访问的硬件资源数目更多的硬件资源，这些硬件资源中一些被预留以供系统管理程序32使用。当来宾操作系统34、36查询可用硬件资源的数目时，不代表存在的硬件资源的数目的真实值的值被传回。

[0063] 图4还图示在各个来宾操作系统34、36的控制下执行的各个应用程序以及与所图示的不同程序相关联的不同权限等级(即，系统管理程序权限等级、特许权限等级、用户权限等级)。本领域技术人员将熟知本技术领域中的此布置。

[0064] 图5图示具有不同权限等级的不同程序的布置的另一示例。在图5的布置中，也有安全域S和非安全域NS。在安全域内，具有在特许权限等级操作的特许等级代码38和在用户权限特级操作的用户代码40。在非安全域内，类似地具有在特许权限等级操作的特许等级代码42和在用户权限等级操作的用户代码44。在此环境内，仅有在安全域内运行的程序代码访问硬件资源的第一部份，而在非安全域中操作的程序代码不访问第一部份内的硬件资源。

[0065] 图6是图示响应于对硬件资源的访问请求而执行的处理控制的流程图。此处理控制由图1的性能监视控制电路16或上下文指针控制电路22执行。

[0066] 在步骤46，该处理等待，直到接收到对硬件资源的访问为止。在步骤48，判断所关注的硬件资源是否在这多个硬件资源的第一部份内。此判断可通过在所访问的硬件资源的索引值和性能监视器边界寄存器14或上下文指针边界寄存器20中指定的当前边界值之间进行比较来完成。

[0067] 若访问不是针对第一部份内的硬件资源，则处理则进行至步骤50，在此步骤中所关注的访问被允许。不在第一部份内的访问将在第二部份内，第一程序和第二程序二者不论其权限等级如何都可访问第二部份。

[0068] 若在步骤48中的判断为访问针对第一部份内的硬件资源，则步骤52判断处理器6的当前操作模式是否为系统管理程序模式。若当前操作模式为系统管理程序模式，则所关注的访问被允许并在步骤50中被执行。若当前模式不是系统管理程序模式，则处理进行至步骤54。系统管理程序模式为其中若某程序被允许访问第一部份内的硬件资源，则该程序必定运行的模式。

[0069] 步骤54判断所关注的访问是否为读取访问。若该访问为读取访问，则步骤56传回一默认值，例如“0”，以取代真实值。如在步骤54中判断该访问不是读取访问，则步骤58产生故障并且忽略写入访问。其它默认动作也是可以的，例如，忽略写入并且传回读取时的实际值。

[0070] 图7示意性地图示本发明技术如何使用一个以上边界来分割硬件资源的。由系统管理程序写入的第一边界将硬件资源分成仅可由系统管理程序访问的第一部份和可由系

统管理程序和来宾操作系统二者访问的第二部份。由系统管理程序或来宾操作系统写入的另一边界值将第二部份分割成进一步的第一部份和进一步的第二部份。进一步的第一部份可由系统管理程序和来宾操作系统程序二者访问。进一步的第二部份可由系统管理程序、来宾操作系统程序和用户程序访问。

[0071] 当用户等级的用户程序查询存在的硬件资源的数目时，对应于进一步的第二部份内的硬件资源的数目的值被传回。当在特许权限等级操作的来宾操作系统查询存在的硬件资源的数目时，对应于第二部份内的硬件资源的数目的值被传回。当系统管理程序在系统管理程序权限模式内从其位置发出关于存在的硬件资源的数目的查询时，对应于第一部份和第二部份内的硬件资源的数目的总和的硬件资源的数目的真实值被传回。

[0072] 响应于关于存在的资源的数目的查询不仅传回不同的值，资源控制电路16、22还对硬件资源重新标引，以使得每一程序的访问看起来在相同的给定点(索引值)开始，而不论他们在实体存在的硬件资源的真实序列内的实际位置如何。因此，举例而言，每一程序不论其在哪一权限等级运行并且不论其被允许访问硬件资源的哪部份，都将被提供对都以某固定值开始的那些硬件资源的索引，例如不论他们在真实的实体硬件资源内的位置如何，所有的索引都可开始于“0”值并且从此值向上延伸。在图2的示例中，第二部份索引自然地以“0”为根，并且第一部份条目被重新标引为以“0”开始。

[0073] 图8图示硬件资源被分割成不同部份的又一示例。在本示例中，图示出4个边界值，每个边界值用于进一步对硬件资源作子分割。在给出的示例中，权限从第零等级延伸至第三等级，其中第零等级边界值仅允许被在第零权限等级操作的程序写入。第一等级边界值可由在第一权限等级或更高的权限等级操作的程序写入。以类似的方式，进一步的边界值利用由在相应权限等级或更高权限等级运行的程序写入的固定边界的位置的边界值，将已被访问的硬件资源的部份进行子分割为进一步的第一部份和第二部份。因此，从图8可见，可使用可在各个权限等级写入的多个边界值来得到对硬件资源子分割的层次。

[0074] 图9示意性地图示硬件资源的另一示例，在此情况中，对应于用于存储器管理单元(MMU)的指向储存上下文数据的存储器地址的指针的表格。第一部份和第二部份之间的边界被储存在首要(override)寄存器60中。该首要寄存器60仅可由在系统管理程序权限等级运行的系统管理程序写入。系统管理程序大小寄存器62储存指示落于图9的第一部份和第二部份内的寄存器的真实总数目的值。来宾操作系统大小寄存器64被提供，当在特许权限等级操作的来宾操作系统读取时，其返回对于可用指针值的数目的值。来宾操作系统大小寄存器64内存储的的来宾操作系统大小具有对应于位置66的默认值。该默认值将响应于来自来宾户操作系统的读取而被传回，除非首要寄存器60已被系统管理程序写入了将边界移至默认位置66以外的位置的不同值。这是在图9中所示的情况。因此，硬件资源的默认数目可被提供至来宾操作系统，并且系统管理程序可在适当时选择性地向来宾操作系统提供更多硬件资源。

[0075] 在本示例中，来宾操作系统使用的指针值小于系统管理程序所需的指针值。因此，若一般仅由系统管理程序使用的指针被配置以供来宾操作系统使用，则由于需要存储储存用于来宾操作系统的较小指针，其将具有过剩的比特空间。此额外空间可简单地被忽略。

[0076] 图10示意性地图示资源大小指示电路。所提供的硬件资源为寄存器池68。大小指示寄存器70(其储存常数或只读值)指向指示所关注的系统中所实现的寄存器的总数目的

值。由较高权限等级代码写入的边界寄存器72当在较低权限等级操作时指示将被提供作为资源大小指示的边界位置。复用器74响应于对大小数据的读取请求，根据指示当前为更高权限等级的访问的复用切换信号，在提供储存于大小指示寄存器70的值和储存于边界寄存器72的值之间切换。因此，就软件而言，该大小指示可总是被认为储存于固定的寄存器内，而当该寄存器被读取时，依赖于系统正操作的权限的当前模式，所传回的值将来自大小指示寄存器70或边界寄存器72。从较低权限等级的代码来看，更高权限等级的代码可被写入至边界寄存器72并相应地可改变资源大小。储存于大小指示寄存器内的值可特定于硬件实施方式，并传回将要由较高权限等级的代码使用的值。

[0077] 图11示意性地图示用于读取资源中的资源的电路。复用器76根据经译码的地址来选择资源寄存器中的一资源寄存器进行读取。然而，若被读取的资源在该模式中可用资源寄存器的范围外，则另一复用器78将传回针对被读取的资源的内容的固定默认值。因此，复用器78将该默认值切换为读取的数据输出，除非该系统在更高权限访问模式中操作或者被访问的资源的地址小于或等于储存在边界寄存器内的值，储存在边界寄存器内该值被设定为当在较低权限访问等级操作时系统可用的资源地址的范围的限值。

[0078] 图12示意性地图示用于向资源68写入的电路。在此布置中，写入的地址被传送至译码器80，该译码器80生成独热(one-hot)信号，该独热信号作为写入使能被提供至资源68内的寄存器中的适当的寄存器。来自译码器80的信号与来自访问控制与(AND)门82的另外的信号进行与运算。访问控制AND门82将写入使能信号与指示系统在较高权限访问模式操作或被访问的寄存器的地址小于或等于该寄存器在较低权限等级可访问的边界位置的信号进行AND运算。因此，在较高的权限访问等级，任何寄存器可被写入，然而在较低权限等级，仅有处于或低于边界的寄存器可被写。在较低权限等级对所允许范围外的寄存器的写入将简单地被视为即使在该寄存器的输入端施加写入数据也不产生写入使能信号。

[0079] 图9中的由指针所指向的配置数据可采用多种不同的形式。图1的存储器管理10可使用此配置数据。此配置数据可包含以下内容中的一个或多个：转换表基指针寄存器值(translation table base pointer register value)；故障地址和上下文寄存器值；转换后备缓冲器维持寄存器值(translating look aside buffer register value)；虚拟地址至实体地址运算(operation)寄存器值；存储器管理单元配置和控制寄存器值；上下文库值，上下文库值包含转换表基指针寄存器值、故障地址和上下文寄存器值、转换后备缓冲器维持寄存器值、虚拟地址至实体地址运算寄存器值和存储器管理单元配置和控制寄存器值；旁路属性寄存器；响应于由第一程序和第二程序中的一个或多个发起的活动而发出的中断输出；以及用于映射数据流的交易识别寄存器值。

[0080] 关于图1和图3讨论的性能监视电路可具有各种不同的形式。性能监视电路可包含一个或多个计数器中的一个或多个，这一个或多个计数器包含一个或多个计数器寄存器、使能寄存器、溢出状态标志和溢出事件中断使能寄存器。性能监视电路可另外地或替代性地包含：事件选择电路，用于从多个事件中选择用以计数的事件。在此上下文中，所述装置可另外包含；多个计数器，这多个计数器包含一个或多个计数器寄存器、使能寄存器、溢出状态标志和溢出事件中断使能寄存器；及事件选择电路，其用于从多个事件选择用以计数的事件。这些元件可用于提供对图3的布置的实体实现。图3中逻辑性地图示的性能监视使能寄存器在实践中可由前文所讨论的各种使能寄存器实体地提供。

[0081] 本发明技术可应用的硬件资源的另一形式为供第一程序和第二程序使用的多个寄存器。例如，这些寄存器可以为多个映射寄存器，映射寄存器储存将要被使用的在被访问的数据流和所述装置相关处理上下文之间的映射。因此，当所述装置切换上下文时，新的映射可快速地可用于新的上下文，就好像其储存于包含系统的硬件资源中的某硬件资源的物理寄存器内一样。

[0082] 本发明技术还可被用在对调试/诊断事件产生资源的管理中。举例而言，实体地提供的断点(break point)和观察点(watch point)资源可根据本发明技术被虚拟化为在虚拟层次内的较低等级执行的程序。在一示例中，这些断点和观察点资源可为断点比较器和观察点比较器。根据本发明技术可管理的诊断资源的其它示例为处理器6的追踪数据产生单元内的地址比较器。根据本发明技术可管理的资源的其它示例对于本领域技术人员而言是很明显的。

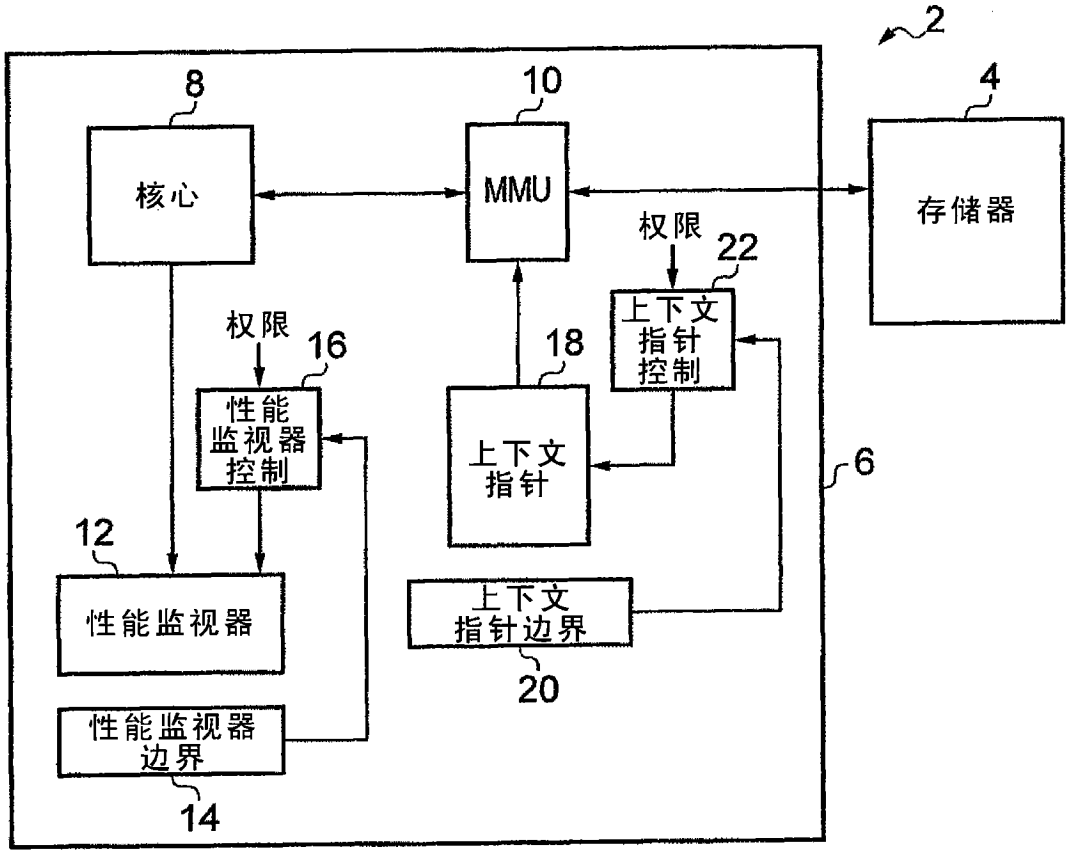

2
8
核心
10
MMU
4
存储器
权限
22
上下文
指针
控制
权限
16
性能
监视器
控制
18
上下文
指针
6
12
性能监视器
上下文
指针边界
20
性能监视器
边界
14

图1

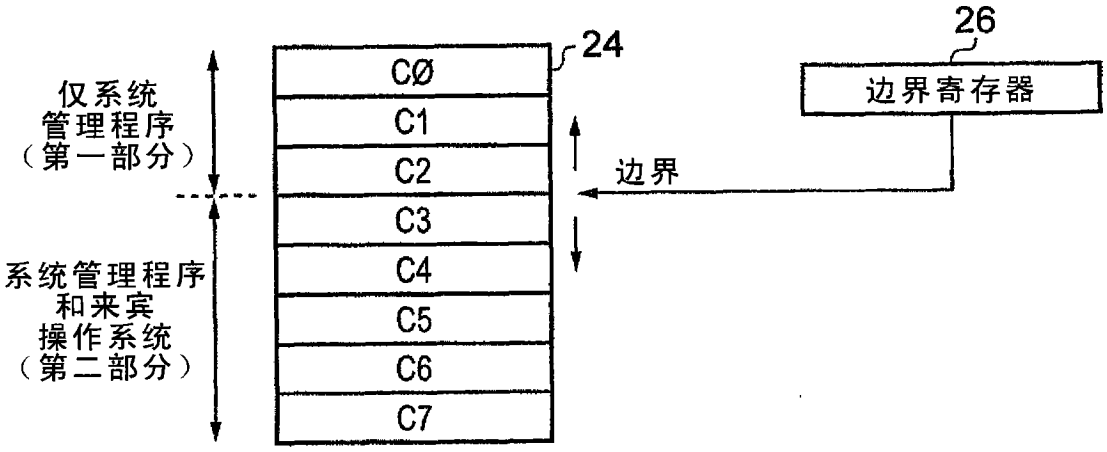

26
边界寄存器
24
C0
C1
C2
C3
C4
C5
C6
C7
边界
仅系统
管理程序
（第一部分）
系统管理程序
和来宾
操作系统
（第二部分）

图2

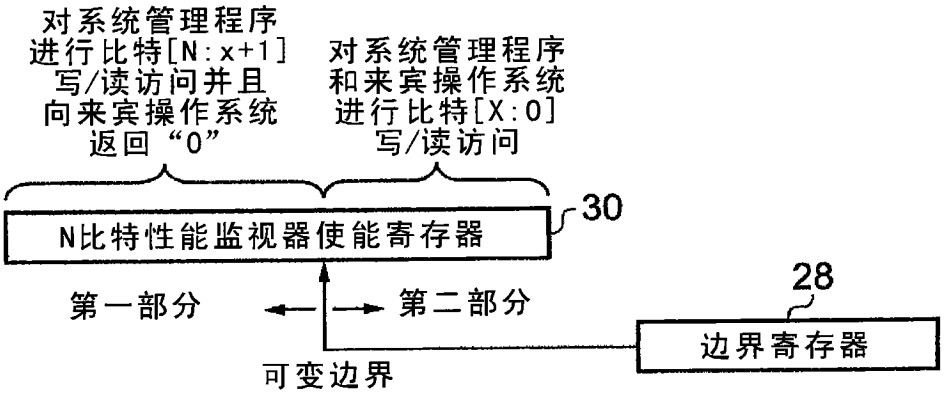

对系统管理程序
进行比特[N:x+1]
写/读访问并且
向来宾操作系统
返回“0”
对系统管理程序
和来宾操作系统
进行比特[X:0]
写/读访问
N比特性能监视器使能寄存器
30
第一部分
第二部分
28
边界寄存器
可变边界

图3

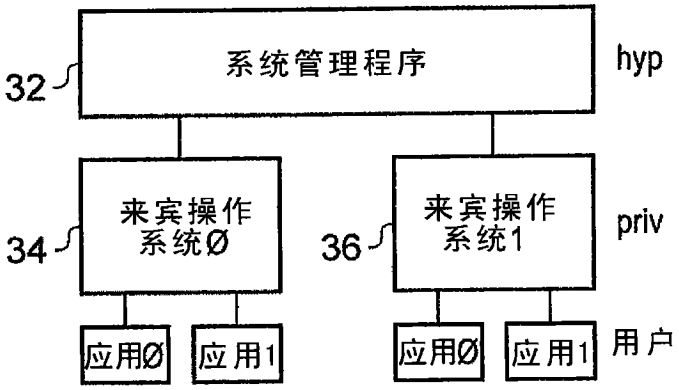

32
系统管理程序
hyp
34
来宾操作
系统Ø
36
来宾操作
系统1
priv
应用Ø
应用1
应用Ø
应用1
用户

图4

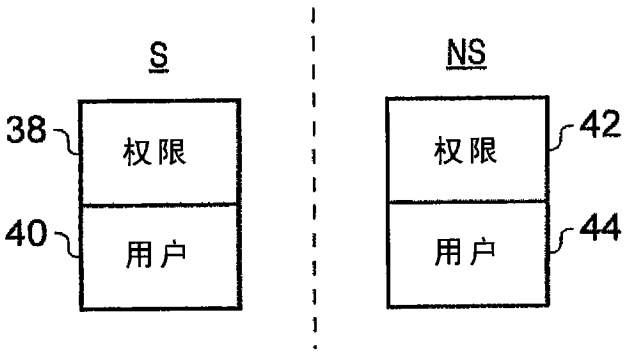

S
NS
38
权限
40
用户
权限
42
用户
44

图5

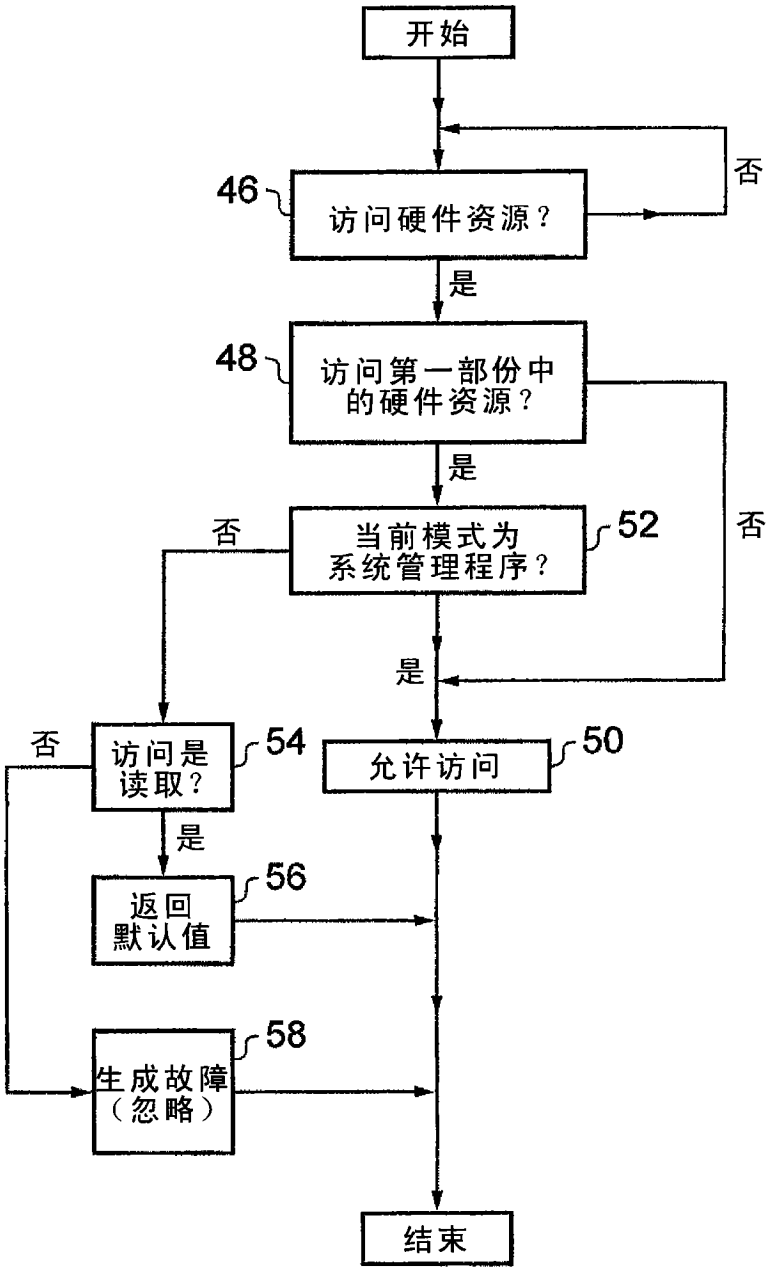

开始
46
访问硬件资源？
否
是
48
访问第一部份中的硬件资源？
是
否
否
52
当前模式为系统管理程序？
是
否
访问是读取？
54
允许访问
50
是
返回默认值
56
生成故障（忽略）
58
结束

图6

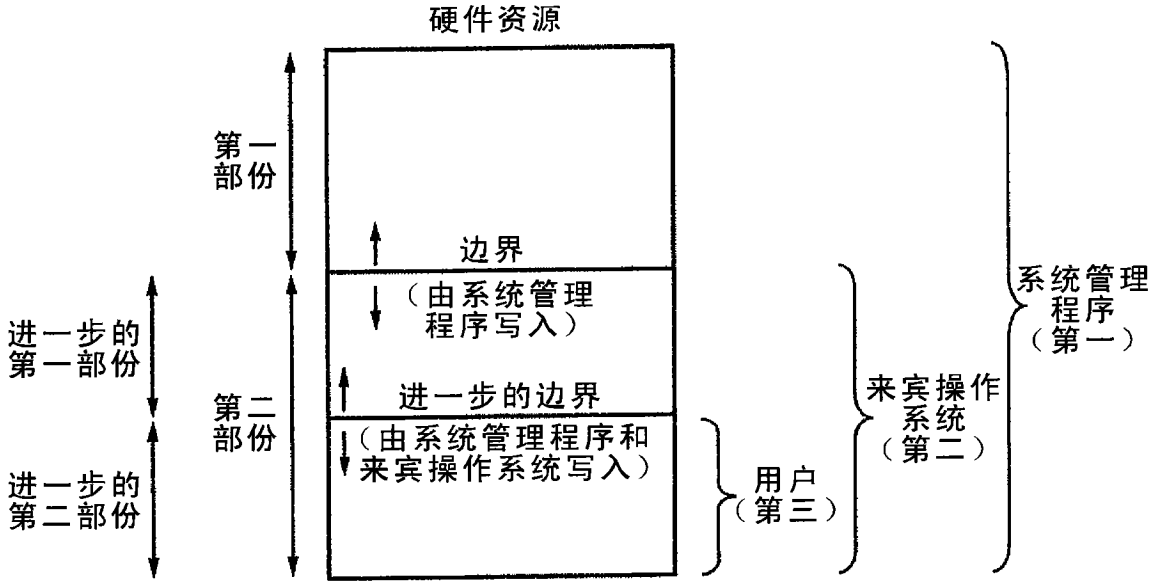

硬件资源
第一部份
第二部份
边界
（由系统管理程序写入）
进一步的边界
（由系统管理程序和来宾操作系统写入）
进一步的第一部份
进一步的第二部份
用户（第三）
来宾操作系统（第二）
系统管理程序（第一）

图7

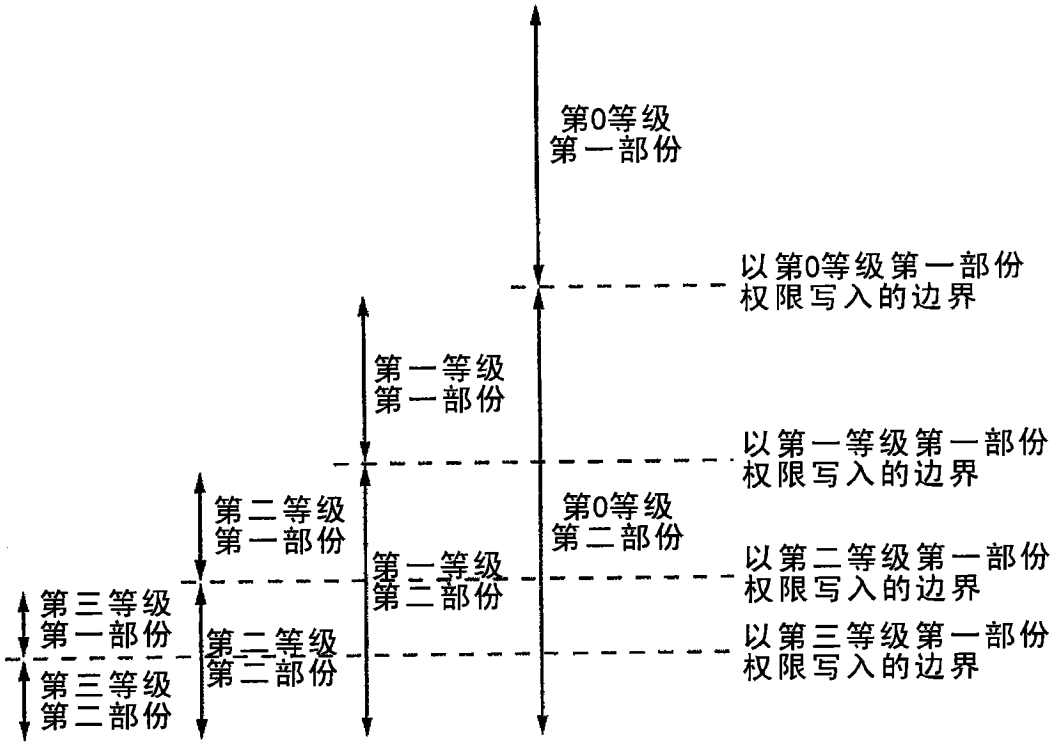

第0等级第一部份
以第0等级第一部份权限写入的边界
第一等级第一部份
以第一等级第一部份权限写入的边界
第二等级第一部份
第0等级第二部份
第一等级第二部份
以第二等级第一部份权限写入的边界
第三等级第一部份
第二等级第二部份
以第三等级第一部份权限写入的边界
第三等级第二部份

图8

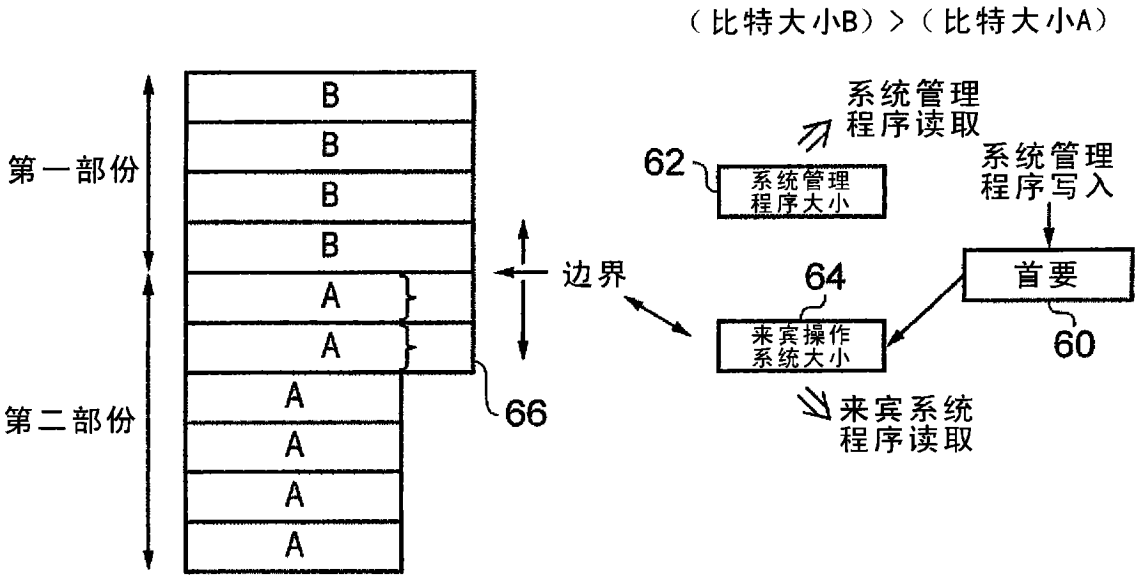

（比特大小B）>（比特大小A）
第一部份
第二部份
B
B
B
B
A
A
A
A
A
A
66
边界
62
系统管理
程序大小
系统管理
程序读取
系统管理
程序写入
64
来宾操作
系统大小
首要
60
来宾系统
程序读取

图9

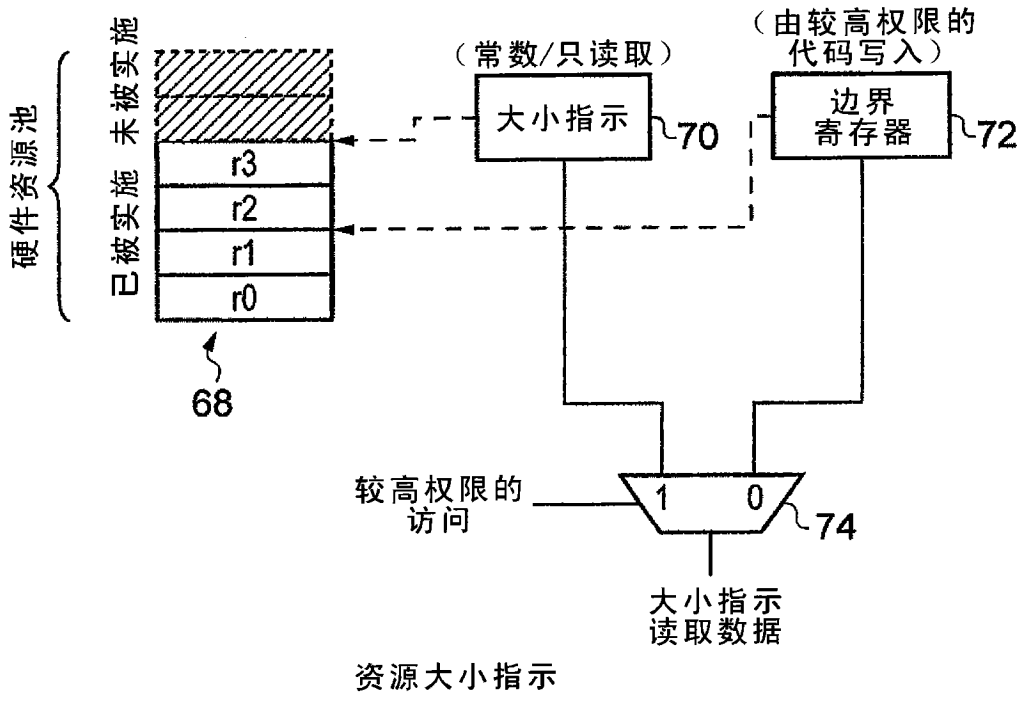
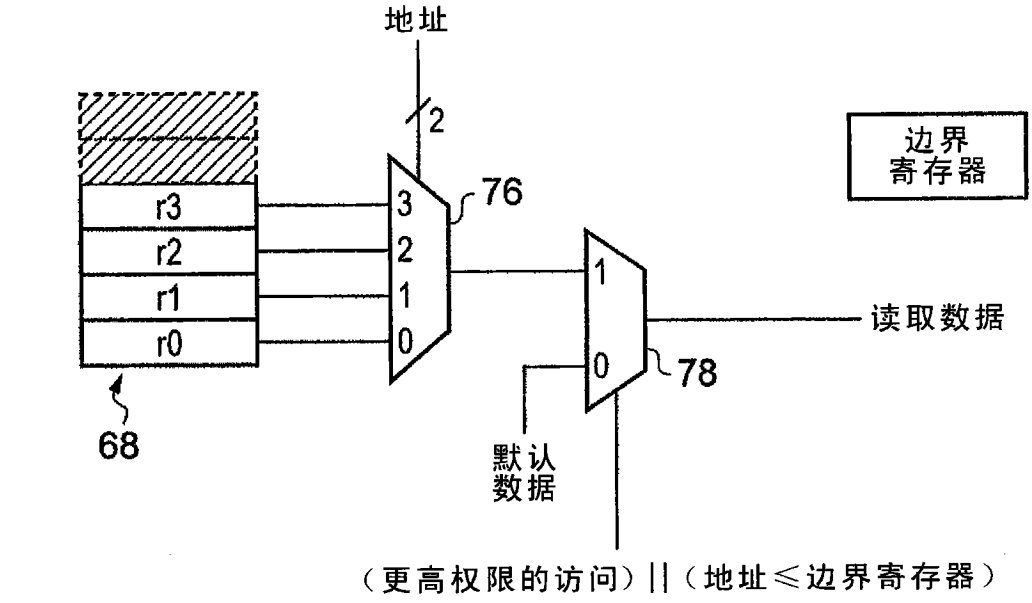
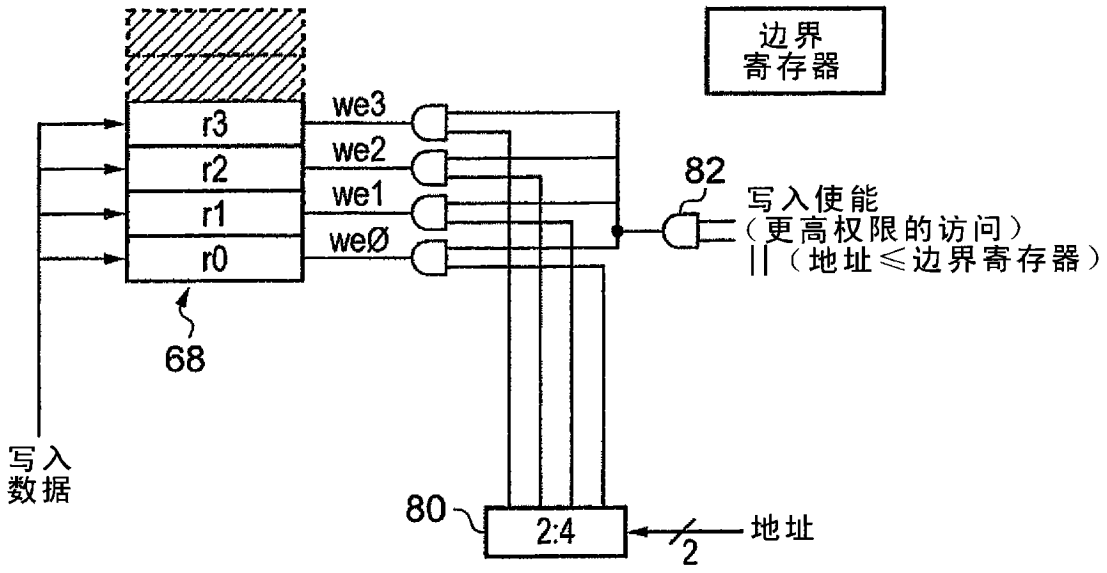

硬件资源池
已被实施
未被实施
r3
r2
r1
r0
68
（常数/只读取）
大小指示
70
（由较高权限的
代码写入）
边界
寄存器
72
较高权限的
访问
1
0
74
大小指示
读取数据
资源大小指示

图10



资源读取电路

图11



资源写入电路

图12