(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 103455733 B
(45)授权公告日 2018.03.27

(21)申请号 201310206388.6

(22)申请日 2013.05.29

(65)同一申请的已公布的文献号
申请公布号 CN 103455733 A

(43)申请公布日 2013.12.18

(30)优先权数据
13/485,078 2012.05.31 US

(73)专利权人 恩智浦美国有限公司
地址 美国得克萨斯

(72)发明人 D·M·麦卡赛 J·C·西尔赛罗
K·A·豪斯曼

(74)专利代理机构 中国国际贸易促进委员会专利商标事务所 11038
代理人 刘倜

(51)Int.Cl.
G06F 21/10(2013.01)

(56)对比文件
CN 1954302 A,2007.04.25,
CN 101281459 A,2008.10.08,
US 2004/0122834 A1,2004.06.24,
US 6282657 B1,2001.08.28,

审查员 李佳曦

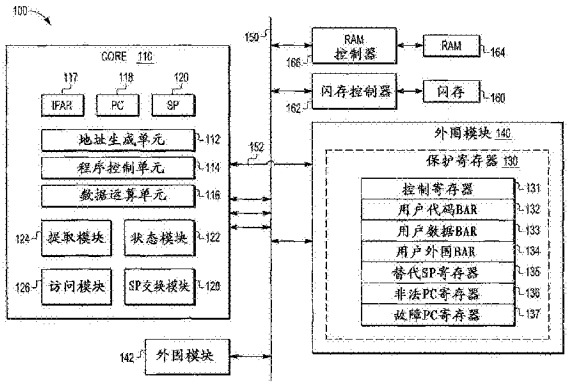
权利要求书4页 说明书17页 附图4页

(54)发明名称

处理器资源和执行保护方法及装置

(57)摘要

实施例包括处理系统(110),其基于存储在第一寄存器(132,212)中的指令地址范围指示器确定下一个指令提取地址是对应于位于与当前权限状态相关联的第一存储器区域(216,218)内的位置还是位于与不同的权限状态相关联的第二存储器区域(216,218)内的位置。当下一个指令提取地址不位于第一存储器区域内(216,218)时,仅仅在到不同的权限状态的转换是合法的时候,允许提取下一个指令。在另一个实施例中,在为指令生成数据访问地址(316)时,基于存储在第二寄存器(133,222)中的数据地址范围指示器,确定是否允许访问与数据访问地址对应的存储器位置。当当前权限状态是允许访问存储器位置的权限状态时,访问被允许(318)。



CN 103455733 B

1.一种处理系统,包括:

数据存储器,其具有被分配给与第一权限状态相关联的第一数据的第一数据存储器区域,以及被分配给与第二权限状态相关联的第二数据的第二数据存储器区域,其中所述第一数据存储器区域包括其中保持第一堆栈的第一堆栈区域,所述第二数据存储器区域包括其中保持第二堆栈的第二堆栈区域;

堆栈指针,被配置用于当当前权限状态是第一权限状态时存储与第一堆栈的顶部元素对应的第一地址,或者当当前权限状态是第二权限状态时存储与第二堆栈的顶部元素对应的第二地址;

替代堆栈指针寄存器,被配置用于当当前权限状态是第二权限状态时存储与第一堆栈的顶部元素对应的第一地址,或者当当前权限状态是第一权限状态时存储与所述第二堆栈的顶部元素对应的第二地址;

第一模块,被配置为基于提取自代码存储器的指令的第三地址,确定当前权限状态是第一权限状态还是第二权限状态;以及

第二模块,被配置为:通过将下一个指令提取地址和至少一个指令地址范围指示器进行比较确定所述下一个指令提取地址是否对应于位于与所述当前权限状态相关联的第一代码存储器区域内的第一存储器位置或与不同的权限状态相关联的第二代码存储器区域内的第二存储器位置,并且当所述下一个指令提取地址不位于与所述当前权限状态相关联的所述第一代码存储器区域内时,仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时允许提取存储在所述下一个指令提取地址的下一个计算机指令,以及当执行从所述当前权限状态到所述不同的权限状态的转换时,用存储在替代堆栈指针寄存器中的地址交换所述堆栈指针中存储的地址。

2.根据权利要求1所述的处理系统,其中所述第一权限状态是具有第一权限级别的管理者状态,以及所述第二权限状态是具有比第一权限级别低的第二权限级别的用户状态。

3.根据权利要求1所述的处理系统,其中:

所述至少一个指令地址范围指示器定义了被分配用于存储与管理者状态相关联的管理者代码的管理者代码存储器区域,并且定义了被分配用于存储与用户状态相关联的用户代码的用户代码存储器区域,并且

所述第二模块被配置为:当所述下一个指令提取地址落入所述管理者代码存储器区域内时,确定所述下一个指令提取地址位于与所述管理者状态相关联的代码存储器区域内,以及当所述下一个指令提取地址落入所述用户代码存储器区域内时,确定所述下一个指令提取地址在与所述用户状态相关联的代码存储器区域内。

4.根据权利要求1所述的处理系统,其中:

所述至少一个指令地址范围指示器定义被分配用于存储与所述第一权限状态相关联的计算机指令的一个或多个第三代码存储器区域,并定义被分配用于存储与所述第二权限状态相关联的计算机指令的一个或多个第四代码存储器区域,并且

所述第二模块被配置为:当所述下一个指令提取地址落入所述一个或多个第三代码存储器区域之一内时,确定所述下一个指令提取地址位于与所述第一权限状态相关联的代码存储器区域内,以及当所述下一个指令提取地址落入所述一个或多个第四代码存储器区域之一内时,确定所述下一个指令提取地址与所述第二权限状态相关联。

5.根据权利要求1所述的处理系统，其中：

所述第二模块还被配置为：当从所述当前权限状态到所述不同的权限状态的转换不是合法时，产生故障并且不允许提取所述下一个计算机指令。

6.根据权利要求1所述的处理系统，还包括：

寄存器，被配置为存储所述至少一个指令地址范围指示器。

7.根据权利要求1所述的处理系统，还包括：

代码存储器，其中所述至少一个指令地址范围指示器在所述代码存储器内定义被分配用于存储与所述第一权限状态相关联的计算机指令的一个或多个第三代码存储器区域，并且所述至少一个指令地址范围指示器在所述代码存储器内定义被分配用于存储与所述第二权限状态相关联的计算机指令的一个或多个第四代码存储器区域。

8.根据权利要求1所述的处理系统，还包括：

第三模块，被配置为：确定为计算机指令生成数据访问地址，通过将所述数据访问地址与至少一个数据地址范围指示器进行比较确定所述数据访问地址是与所述第一权限状态相关联还是与所述第二权限状态相关联，以及当所述当前权限状态是允许访问与所述数据访问地址对应的第三存储器位置的权限状态时允许访问所述第三存储器位置。

9.根据权利要求8所述的处理系统，其中：

所述至少一个数据地址范围指示器定义所述第一数据存储器区域以及被分配用于存储与所述第一权限状态相关联的数据的一个或多个第三数据存储器区域，并定义所述第二数据存储器区域以及被分配用于存储与所述第二权限状态相关联的数据的一个或多个第四数据存储器区域，并且

所述第三模块被配置为：当所述数据访问地址落入所述第一数据存储器区域或所述一个或多个第三数据存储器区域中的一方内时，确定所述数据访问地址与所述第一权限状态相关联，以及当所述数据访问地址落入所述第二数据存储器区域或所述一个或多个第四数据存储器区域中的一方内时，确定所述数据访问地址与所述第二权限状态相关联。

10.根据权利要求8所述的处理系统，还包括：

寄存器，被配置用于存储所述至少一个数据地址范围指示器。

11.根据权利要求8所述的处理系统，其中所述至少一个数据地址范围指示器定义所述第一数据存储器区域和所述第二数据存储器区域。

12.根据权利要求8所述的处理系统，其中：

所述第三模块还被配置用于：当所述当前权限状态是不允许访问所述第三存储器位置的权限状态时，产生故障并且禁止访问与所述数据访问地址对应的所述第三存储器位置。

13.一种处理方法，包括：

存储至少一个数据地址范围指示器，所述至少一个数据地址范围指示器定义数据存储器内被分配给与第一权限状态相关联的第一数据的一个或多个第一数据存储器区域，以及定义数据存储器内被分配给与第二权限状态相关联的第二数据的一个或多个第二数据存储器区域，其中所述一个或多个第一数据存储器区域包括其中保持第一堆栈的第一堆栈区域，所述一个或多个第二数据存储器区域包括其中保持第二堆栈的第二堆栈区域；

存储至少一个指令地址范围指示器，所述至少一个指令地址范围指示器定义被分配用于存储与第一权限状态相关联的计算机指令的一个或多个第一代码存储器区域，并且定义

被分配用于存储与第二权限状态相关联的计算机指令的一个或多个第二代码存储器区域；

基于取自代码存储器的指令的第一地址将当前权限状态定义为所述第一权限状态或所述第二权限状态；

当当前权限状态是第一权限状态时，在堆栈指针内存储与第一堆栈的顶部元素对应的第二地址，或者，当当前权限状态是第二权限状态时，在堆栈指针内存储与第二堆栈的顶部元素对应的第三地址；

当当前权限状态是第二权限状态时，在替代堆栈指针寄存器内存储与第一堆栈的顶部元素对应的第二地址，或者，当当前权限状态是第一权限状态时，在替代堆栈指针寄存器内存储与所述第二堆栈的顶部元素对应的第三地址；

通过将下一个指令提取地址与所述至少一个指令地址范围指示器进行比较，确定所述下一个指令提取地址是对应于位于与所述当前权限状态相关联的代码存储器区域内的第一存储器位置还是对应于与不同的权限状态相关联的代码存储器区域内的第二存储器位置；

当所述下一个指令提取地址不位于与所述当前权限状态相关联的代码存储器区域内时，仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时，允许提取存储在所述下一个指令提取地址的下一个计算机指令；以及

当执行从所述当前权限状态到所述不同的权限状态的转换时，用存储在替代堆栈指针寄存器中的地址交换所述堆栈指针中存储的地址。

14.根据权利要求13所述的方法，其中所述第一权限状态是具有第一权限级别的管理者状态，而所述第二权限状态是具有比第一权限级别低的第二权限级别的用户状态。

15.根据权利要求13所述的方法，还包括：

当从所述当前权限状态到所述不同的权限状态的转换不是合法的时候，产生故障并且不允许提取所述下一个计算机指令。

16.根据权利要求13所述的方法，还包括：

确定计算机指令生成数据访问地址，其中所述数据访问地址对应于第三存储器位置；

当生成所述数据访问地址时，通过将所述数据访问地址与所述至少一个数据地址范围指示器进行比较，确定所述第三存储器位置是与所述第一权限状态相关联的存储器位置还是与所述第二权限状态相关联的存储器位置；以及

当所述当前权限状态是允许访问与所述数据访问地址对应的所述第三存储器位置的权限状态时，允许访问所述第三存储器位置。

17.根据权利要求16所述的方法，还包括：

当所述当前权限状态是不允许访问所述第三存储器位置的权限状态时，产生故障并且禁止访问与所述数据访问地址对应的所述第三存储器位置。

18.一种处理方法，包括：

存储至少一个数据地址范围指示器，所述至少一个数据地址范围指示器定义数据存储器内被分配给与管理者状态相关联的第一数据的一个或多个第一数据存储器区域，以及定义数据存储器内被分配给与用户状态相关联的第二数据的一个或多个第二数据存储器区域，其中所述一个或多个第一数据存储器区域包括其中保持第一堆栈的第一堆栈区域，所述一个或多个第二数据存储器区域包括其中保持第二堆栈的第二堆栈区域；

将当前权限状态定义为所述管理者状态或所述用户状态；

当当前权限状态是管理者状态时，在堆栈指针内存储与第一堆栈的顶部元素对应的第二地址，或者，当当前权限状态是用户状态时，在堆栈指针内在存储与第二堆栈的顶部元素对应的第三地址；

当当前权限状态是用户状态时，在替代堆栈指针寄存器内存储与第一堆栈的顶部元素对应的第二地址，或者，当当前权限状态是管理者状态时，在替代堆栈指针寄存器内存储与所述第二堆栈的顶部元素对应的第三地址；

通过将下一个指令提取地址与指令地址范围指示器进行比较，确定所述下一个指令提取地址是对应于位于与所述当前权限状态相关联的代码存储器区域内的第一存储器位置还是位于与不同的权限状态相关联的代码存储器区域内的第二存储器位置，其中所述指令地址范围指示器定义被分配用于存储管理者代码的管理者代码存储器区域和被分配用于存储用户代码的用户代码存储器区域；

当所述下一个指令提取地址不位于与所述当前权限状态相关联的存储器区域内时，仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时，允许提取存储在所述下一个指令提取地址的下一个计算机指令；以及

当执行从所述当前权限状态到所述不同的权限状态的转换时，用存储在替代堆栈指针寄存器中的地址交换所述堆栈指针中存储的地址。

19.根据权利要求18所述的方法，其中：

当所述当前权限状态是所述管理者状态而所述不同的权限状态是所述用户状态时，当一组指令内的指令被执行时从所述管理者状态到所述用户状态的转换是合法的，其中该组指令包括从中断返回指令。

20.根据权利要求18所述的方法，其中：

当所述当前权限状态是所述用户状态而所述不同的权限状态是所述管理者状态时，当一组指令内的指令被执行时，其中该组指令包括软件中断指令和陷阱指令，或者当检测到异常情况时，从所述用户状态到所述管理者状态的转换是合法的。

21.根据权利要求18所述的方法，还包括：

确定计算机指令生成数据访问地址，其中所述数据访问地址对应于第三存储器位置；

当生成所述数据访问地址时，通过将所述数据访问地址与所述至少一个数据地址范围指示器进行比较，确定所述第三存储器位置是与所述管理者状态相关联的存储器位置还是与所述用户状态相关联的存储器位置；

当所述当前权限状态是所述管理者状态时，允许访问所述第三存储器位置；以及

当所述当前权限状态是所述用户状态时，仅仅当所述第三存储器位置是位于所述一个或多个第二数据存储器区域内时，允许访问所述第三存储器位置。

处理器资源和执行保护方法及装置

技术领域

[0001] 实施例通常涉及处理器资源和执行保护方法及装置。

背景技术

[0002] 典型的计算机系统可以向在处理器上执行的代码提供不同级别的系统资源访问。这常常是通过例如将多个分层级的权限级别中的一个与可以在系统上运行的每一类型的代码相关联来实现。在这样系统中，最受新任的代码(例如，内核(kernel)代码)被授予最高级别的权限，并且不太受信任的代码(例如，装置驱动程序、应用程序和其它用户代码)被授予较低级别的权限。权限级别越高，代码就具有越多的对系统资源的访问。相反，权限级别越低，代码就具有越少的对系统资源的访问。例如，内核代码可能实际上不受限定地访问系统资源，而应用程序代码可能会被限定访问某些存储区域、输入/输出(I/O)等等。

[0003] 为了减少恶意代码或错误代码可能会影响系统的功能和/或数据的风险，分层级的权限方案通常是硬件实施的。例如，处理器可以实现“管理者”和“用户”操作模式以实现分层级的权限方案。在这样系统中，可以实现以硬件为媒介的标志，其中所述标志的状态用于确定是否允许各种资源影响操作的执行。当系统处于管理者模式的时候所述标志可以被设置，例如，允许高权限代码广泛访问系统资源。相反，当系统处于用户模式的时候所述标志可以被清除，从而限定较低权限的代码对某些资源的访问。

[0004] 在处理器上以各种不同权限级别执行代码的能力通过利用这些权限级别允许构建软件，以使得两个或两个以上级别的软件可以共存，同时保持硬件实现的隔离。然而，这种能力通常的代价是大量处理器逻辑跟踪权限状态和控制状态变化。通常，提供分层级的权限方案的硬件实施的逻辑是处理器设计的组成部分，并且就硬件开发和周期时间而言，将这种功能添加到现有处理器架构可能是成本高昂的。

[0005] 概述

[0006] 根据本公开的一个方面，提供了一种处理系统，包括：第一模块，被配置为基于取自存储器的指令的地址保持当前权限状态是第一权限状态还是第二权限状态的知识；以及第二模块，被配置为：通过将下一个指令提取地址和至少一个指令地址范围指示器进行比较确定所述下一个指令提取地址是否对应于位于与所述当前权限状态相关联的第一存储器区域内的位置或与不同的权限状态相关联的第二存储器区域内的位置，并且当所述下一个指令提取地址不位于与所述当前权限状态相关联的所述第一存储器区域内时，仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时允许提取下一个计算机指令。

[0007] 根据本公开的另一方面，提供了一种处理方法，包括：存储至少一个指令地址范围指示器，所述至少一个指令地址范围指示器定义一个或多个被分配用于存储与第一权限状态相关联的计算机指令的第一存储器区域，并且定义一个或多个被分配用于存储与第二权限状态相关联的计算机指令的第二存储器区域；基于取自存储器的指令的地址将当前权限状态定义为所述第一权限状态或所述第二权限状态；通过将下一个指令提取地址与所述至少一个指令地址范围指示器进行比较，确定所述下一个指令提取地址是对应于位于与所述

当前权限状态相关联的存储器区域内的位置还是位于与不同的权限状态相关联的存储器区域内的位置;以及当所述下一个指令提取地址不对应于位于与所述当前权限状态相关联的存储器区域内的位置时,仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时,允许提取所述下一个计算机指令。

[0008] 根据本公开的又一方面,提供了一种处理方法,包括:将当前权限状态定义为管理者状态或用户状态;通过将下一个指令提取地址与指令地址范围指示器进行比较,确定所述下一个指令提取地址是对应于位于与所述当前权限状态相关联的存储器区域内的位置还是位于与不同的权限状态相关联的存储器区域内的位置,其中所述指令地址范围指示器定义被分配用于存储管理者代码的管理者代码存储器区域和被分配用于存储用户代码的用户代码存储器区域;以及

[0009] 当所述下一个指令提取地址不对应于位于与所述当前权限状态相关联的存储器区域内的位置时,仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时,允许提取所述下一个计算机指令。

附图说明

[0010] 图1是根据一个示例实施例的处理系统的简化框图;

[0011] 图2是根据一个示例实施例的划分的代码和数据存储器的简化例子;

[0012] 图3是根据一个示例实施例的操作实现资源和执行保护的系统的方法的流程图;

[0013] 图4是说明了根据一个示例实施例的管理者状态和用户状态以及它们之间的转换的简化状态图。

具体实施方式

[0014] 本发明描述的实施例包括可以在处理系统中实施的资源和执行保护方法以及装置。实质上,实施例使得资源和执行保护在硬件方面以很低的成本实现。例如,可以通过使用在处理器中实现的相对简单的逻辑,连同一组被用于存储可以被具有不同权限级别的代码访问的资源的指示的寄存器,在系统中实现资源和执行保护。现在将描述其中已经整合了资源和执行保护的实施例的处理系统的例子。如将在后面解释的,图1的示例系统并不是限制性的,并且这些实施例也可以在显著不同的架构中实现。

[0015] 图1是根据一个示例实施例的处理系统100的简化框图。至少,系统100包括一个或多个处理器核心110、存储器资源(例如,非易失性闪存存储器160和易失性随机存取存储器(RAM)164)、以及一组“保护”寄存器130(或其它数据存储结构);与资源和执行保护相关的信息可以被存储在所述寄存器中。此外,系统100可以包括一个或多个外围模块140、142。例如,根据更加具体的实施例,该组保护寄存器130被实现为外围模块140的一部分,其对于核心110可通过系统总线150访问。此外或替代地,在核心110和外围模块140之间可以存在直接信令连接152(例如,以有助于核心110和保护寄存器130之间的值的交换)。在其它实施例中,一个或多个保护寄存器130可以位于其它位置(例如,在核心110中或某些其它位置)。如在后面将更详细地解释的,保护寄存器130被用于实现由系统100进行的资源和执行保护机制的实施例。

[0016] 闪存存储器160和RAM164还可以分别地通过闪存控制器162和RAM控制器166经过

系统总线150访问核心110。系统100可以也包括一个或多个其它处理器核心、总线、外围模块、代码以及数据存储结构、输入/输出(I/O)、以及其它组件。在不同实施例中，系统100可以在单一的集成电路(例如，作为片上系统)或在多个集成电路(例如，作为封装中系统或包括多个单独封装的集成电路的系统)上实施。

[0017] 在一个实施例中，核心110包括地址生成单元112、程序控制单元114、数据运算单元116、指令提取地址寄存器(IFAR)117、程序计数器(PC)寄存器118、堆栈指针(SP)寄存器120、以及各种模块122、124、126、128。在其它实施例中，核心110可以包括附加的或不同的组件。虽然IFAR117、PC寄存器118、SP寄存器120、以及模块122、124、126、128被显示为与地址生成单元112、程序控制单元114、以及数据运算单元116不同，但这仅仅是为了描述的方便和清晰。在各种实施例中，IFAR117、PC寄存器118、SP寄存器120、和/或模块122、124、126、128可以被结合到下列中的任何一个或多个中：地址生成单元112、程序控制单元114、以及数据运算单元116。

[0018] 实质上，核心110被配置来提取、译码和执行计算机指令形式的代码。例如，但不是限制的方式，核心110可以提取存储在核心110外部的存储器内的代码(例如，存储在闪存存储器160内的代码)。在代码执行期间，核心110可以执行各种运算操作、逻辑操作以及I/O操作，连同通过系统总线150对存储器(例如，RAM164)以及各种外围装置(例如，外围模块140、142和其它的外围模块(未显示))的访问(例如，读取/写入)。核心110执行的大多数指令指定一个或多个操作对象(operand)。此外，在某些情况下，指令可以包括或指定位于存储器中或其它地方(例如，闪存存储器160、RAM164、外围模块140、142、或某些其它位置)的地址。例如，指令可以包括闪存存储器160内的地址，该地址表示执行线程应该跳转到的指令的位置。替代地，指令可以包括或指定从中提取数据值或数据值应当要被存储的位于RAM164、外围设备140、142、或一些其它位置的地址。这样指令的执行被认为是生成数据访问事件(例如，读或写事件)。

[0019] 如在下面将更详细地描述的，并且根据不同实施例，核心110被配置为使用模块122、124、126、128以及存储在保护寄存器130中的信息，来实现资源和执行保护机制。实质上，模块122、124、126、128包括硬件(例如，逻辑和其它电路)以使得能够执行下面描述的功能。在一些实施例中，用于执行下面描述的功能的“模块”可以使用硬件和软件实现。因此，术语“模块”意指包括仅仅使用硬件(例如，逻辑门配置以及其它相关联的电路)实现的以及使用硬件和软件的组合实现的实施例。此外，虽然模块122、124、126、128被显示为彼此不同，并且与核心110的其它部分不同，但是模块122、124、126、128中的部分或全部可以完全地或部分地相互集成和/或与核心110的其它部分集成。基于本发明的描述，本领域技术人员将理解如何实现模块122、124、126、128，并且因此，在此不详细讨论模块122、124、126、128的特定配置。

[0020] 本发明描述的资源和执行保护机制的实施例支持具有相对高的权限级别的代码和数据与具有相对低的权限级别的代码和数据的硬件实施的分离。如在此所使用的，术语“管理者权限级别”指在系统100中实现的最高权限级别，而“用户权限级别”指较低的权限级别。同样地，术语“管理者代码”指指定具有管理者权限级别的代码(例如，操作系统内核代码和其它高度受信任的代码)，而“用户代码”指指定具有用户权限级别的代码(例如，其执行是由内核代码管理的应用程序代码)。虽然在此详细讨论的实施例仅仅涉及两个权限

级别(即,管理者权限级别和用户权限级别),然而其它实施例可以实现具有两个以上权限级别的代码的硬件实现的分离,并且意图将这些实施例包括在本发明的主题范围内。在详细讨论资源和执行保护机制的实施例之前,将首先讨论核心110的基本操作。

[0021] 核心110可以有流水线架构,其中指令被作为在连续时钟周期期间执行的步骤序列而执行。例如,对于任何特定指令,核心110可以执行以下的步骤:指令提取,译码指令,执行指令,执行存储器访问(当指令指定了位于闪存存储器160、RAM164、或其它地方中的地址时),以及在某些情况下,指令执行结果的寄存器回写。在一个实施例中,核心110包括两个指令地址寄存器以便于流水线架构的实现。这些指令地址寄存器包括IFAR117和PC寄存器118。

[0022] IFAR117被用于定义当前位于指令提取流水线顶部的指令的存储器地址,在此称为"指令提取地址"。IFAR117可以加载有关于每个指令流"不连续"(例如,不包括对存储在存储器内的连续地址中的指令的访问,但相反包括对存储在非连续地址中的指令的访问的指令流转换)的指令提取地址。例如,由于所采取的分支指令,响应于特定指令(例如,软件中断、陷阱指令,等等)的执行,或响应于异常情况的发生,可能出现指令流不连续。一旦加载以指令提取地址,则发起相应的指令提取。更具体地,指令提取地址进行到系统总线150上,并且到存储器(例如,闪存存储器160)中的目标目的地。随后,目的地存储器将提取的指令返回到核心110。随着这个以及每个后续提取周期的完成,IFAR117中的地址被增加以指令提取的宽度(或"提取宽度"),除非出现另一指令提取不连续。如果不存在指令提取不连续,则指令提取流水线继续连续地生成地址并访问适当的目的地存储器。当发生重新定向提取流的指令提取不连续的时候,IFAR117被重新加载新的、非连续的指令提取地址,目标目的地存储器被访问,并且指令提取流水线恢复其连续地址生成和目的地存储器访问过程。

[0023] 随着提取的指令返回到核心110,该指令被加载到执行流水线中。PC寄存器118被用于定义当前驻留在执行流水线顶部的指令的存储器地址,在此被称为"PC地址"。一旦指令通过执行流水线开始进行,则基于指令长度,与下一个指令到达执行流水线的顶部(来自内部指令缓冲器或来自系统总线150上的提取周期)同时地使PC寄存器118内的PC地址递增。在某些情况下,指令提取宽度(例如,指令提取总线的宽度)以及指令长度可以相等(例如,两者都是2字节,或一些其它长度),在这种情况下,IFAR117和PC寄存器118内的地址以相同的值(即,指令长度)递增。在其它情况下,指令提取宽度可以与指令长度不同(例如,比指令长度宽)。例如,指令提取宽度可以是指令长度的两倍(例如,分别是4字节和2字节,或一些其它长度),在这种情况下,每个指令提取可以导致同时将两个指令提取到核心110。在这样的实施例中,直到遇到指令流不连续,IFAR117内的地址将以指令提取宽度(例如,2x字节)递增,而PC寄存器118内的地址以FAR117递增速率两倍的速率以指令长度(例如,x字节)递增。为了便于解释,下面的讨论可能涉及基于IFAR117内的地址提取"指令"。基于本发明的描述,本领域技术人员将理解,当基于IFAR117内的地址提取多个指令的时候(例如,当指令提取宽度比指令长度宽的时候),可以如何修改本发明主题的实施例。

[0024] 一旦系统启动(例如,通电),地址生成单元112将第一指令提取地址(例如,复位向量)加载到IFAR117。第一指令提取地址对应于系统的初始化代码(例如,包括基本输入输出系统(BIOS)和核心初始化代码)的在存储器内(例如,闪存存储器160内)的起始地址。然后

核心110从与IFAR117内指定的指令提取地址对应的一个或多个存储器位置(例如,闪存160内)提取一个或多个第一指令(取决于指令提取宽度)。使用第一提取指令初始化PC寄存器118,并且相应的,提取的指令然后进入执行流水线。在进入到执行流水线中时,程序控制单元114对指令进行译码并且控制其执行。随着每个指令提取被发起,地址生成单元112确定IFAR117内的指令提取地址是否要递增,或者,当指令中指示有指令流不连续的时候,地址生成单元112确定提取与下一个指令对应的指令提取地址。无论哪种方式,IFAR117内的指令提取地址可以根据所述确定而被修改。

[0025] 如前面所示的,除了执行管理者代码(例如,内核代码),核心110还可以执行具有相对低的权限级别的用户代码(与管理者代码的权限级别相比时)。多种实施例构思核心110执行管理者代码和用户代码之间有序转换。更具体地,在一个实施例中,核心110实现了至少包括“管理者状态”(例如,图4的管理者状态410)和“用户状态”(例如,图4的用户状态420)的状态机。如后面将结合图3和图4更详细地描述的,基于核心110已经允许从存储器提取(例如,基于IFAR117内的指令提取地址)的最新近指令的地址确定所述状态。例如,当核心110已经允许提取的最新近指令是来自分配给管理者代码的存储器部分(例如,图2的管理者代码存储区216)的时候,核心110处于管理者状态。相反,当核心110已经允许提取的最新近指令是来自分配给用户代码的存储器部分(例如,图2的用户代码存储区218)的时候,核心110处于用户状态。在一个实施例中,核心110最新近允许提取的指令的权限级别定义“当前权限状态”。如在后面将更详细地描述的,核心110包括被配置用于确定和保持当前权限状态的知识的模块(在图1被示出为状态模块122)。更具体地,在一个实施例中,核心110包括被配置用于通过将至少一个指令地址范围指示器(例如,后面讨论的存储在图1和图2的用户代码基址寄存器(BAR)132、212中的值)和最新近允许的指令提取的地址(例如,基于IFAR117内的当前指令提取地址的地址)进行比较,确定当前权限状态是管理态还是用户状态的模块(例如,状态模块122)。在一个实施例中,状态模块122可以形成程序控制单元114的一部分。在其它实施例中,用于执行此功能的硬件可以位于其它位置。

[0026] 在一个实施例中,一旦IFAR117内的值被更新,则基于当前权限状态、提取下一个指令的地址(例如,基于IFAR117内的指令提取地址的地址)、以及存储在保护寄存器130内的指令地址范围指示器,核心110可以允许或禁止下一个指令被提取。如在后面将更详细地描述的,核心110包括被配置来确定是否允许或禁止指令的提取的模块(在图1被示出为提取模块124)。更具体地,在一个实施例中,核心110包括被配置来通过将下一个指令提取地址和至少一个指令地址范围指示器(例如,后面讨论的存储在用户代码BAR132中的值)进行比较,确定下一个要提取的指令的地址(例如,基于IFAR117内的指令提取地址的地址)是否位于与当前权限状态相关联的存储器区域(例如,闪存存储器160)内或位于与不同的权限状态相关联的存储器区域内的模块(例如,提取模块124)。当下一个指令提取地址不位于与当前权限状态相关联的存储器区域内的时候,核心110被配置为仅仅在特定条件下允许下一个计算机指令被提取、解码、以及执行,如后面将要讨论的。在一个实施例中,提取模块124可以形成程序控制单元114的一部分。在其它实施例中,用于执行此功能的硬件可以位于其它位置。假定允许提取指令,程序控制单元114控制指令译码和执行过程。例如,在程序控制单元114的控制下,数据运算单元116(或核心110的其它部分)可以执行对运行(例如,执行)指令来说必要的任何各种运算和逻辑操作。

[0027] 如前面所示的，指令可以定义数据引用（例如，地址），其又可以发起对系统资源（例如，对RAM164，外围模块140、142，或其它地方中的位置）的访问。在一个实施例中，关于在系统100内实现的资源保护，核心110可以基于指令的权限级别（或者，当前权限状态或从其指令提取的地址），以及存储在保护寄存器130中的数据地址范围指示器，允许或拒绝对资源的访问。如在后面将更详细地描述的，核心110包括被配置来确定是否允许或拒绝对特定系统资源的访问的模块（在图1被示出为访问模块126）。更具体地，在一个实施例中，核心110包括被配置来确定在执行指令时是否生成数据访问地址（例如，引用RAM164的地址），并且通过将所述数据访问地址与至少一个数据地址范围指示器（例如，存储在图1和图2的用户代码BAR133、222的值，稍后将讨论）进行比较，以确定所述数据访问地址是否与管理者状态或用户状态相关联的模块（例如，访问模块126）。在当前权限状态是允许访问存储器位置（例如，RAM164中的位置）的权限状态时，核心110被配置为允许对与数据访问地址对应的存储器位置的访问。在一个实施例中，访问模块126可以形成程序控制单元114的一部分。在其它实施例中，用于执行此功能的硬件可以位于其它位置。

[0028] 如前面所示的，保护寄存器130内的信息可以由核心110结合提供资源和执行保护功能而访问。除了别的之外，系统初始化代码可以包括这样的指令，该指令在被执行时在保护寄存器130中建立各种设置或值。例如，保护寄存器130可以包括控制寄存器131、用户代码BAR132、用户数据BAR133、用户外围BAR134、替代SP寄存器135、非法PC寄存器136、以及故障PC寄存器137。

[0029] 现在简要描述控制寄存器131，而每个其它寄存器132-137的描述将在下面适当的部分被描述。在一个实施例中，控制寄存器131包括一个或多个寄存器锁定标志和使能标志。当包括单一寄存器锁定标志时，当寄存器锁定标志被清除时，向核心110指示保护寄存器130内的值可以被在核心110（假定代码有适当的权限级别）上执行的代码改变。相反，当寄存器锁定标志被设置时，保护寄存器130内的值不可以被修改（例如，外围模块140或保护寄存器130将不允许任何代码修改保护寄存器130中的值）。在替代实施例中，控制寄存器131可以包括多个寄存器锁定标志以提供更细粒度的对保护寄存器130的访问（例如，所述多个寄存器锁定标志中的每一个可以被用于控制对一个或多个保护寄存器130的组）。使能标志当被设置时，表示系统的资源和执行保护机制被使能，并且核心110应该实现这些机制。当使能标志被清除时，核心110将禁用与资源和执行保护机制相关联的操作。在另一个实施例中，核心可以选择性地绕过（bypass）与资源和执行保护机制相关联的操作。

[0030] 在一个实施例中，在系统启动或复位时，核心110确定资源和执行保护使能信号是否已被提供给系统（例如，通过使能输入）。如果是，则核心110允许初始化代码设置使能标志。此外，在系统复位时，寄存器锁定标志（或若干寄存器锁定标志）被清除，从而使得初始化代码可以随后在寄存器132-137内建立各种值。一旦这些值已在寄存器132-137内建立，初始化代码可以设置寄存器锁定标志（一个或多个）以使得其它代码不可以改变与所设置的标志对应的寄存器132-137内的值。这里的描述假定保护寄存器130（特别是，寄存器132-134）中的值在系统100的正常操作期间（例如，在系统初始化之后）不被改变。在其它实施例中，一些或全部保护寄存器132-134内的值也可以在系统初始化之后（例如，在某些情况下和/或在不同的时间）被重新加载或改变。例如，保护寄存器132-134内的值可以被改变以增加和/或减少对于管理者代码或用户代码可访问的资源的量。作为另一示例，某些保护寄存

器132-134内的值可以在每当用户进程(例如,任务)开始执行时加载(或重新加载)。保护寄存器132-134内的值也可以在其它时间改变。虽然在此没有详细讨论这些实施例,但仍意图将它们包括在本发明主题的范围内。

[0031] 如在后面将更详细地讨论的,寄存器132-134被用于定义与管理者状态相关联的各种资源部分(例如,闪存存储器160、RAM164、外围装置等等)以及与用户状态相关联的资源部分。在一个实施例中,保护寄存器130可以只被管理者代码(例如,内核代码)访问,而外围模块140被认为是管理级别资源。

[0032] 除了在至少一些保护寄存器130内建立值之外,初始化代码还可以建立堆栈以供管理者代码使用(在此称为"管理堆栈"(例如,图2的管理者堆栈227))。例如,管理者堆栈可以通过将"管理者堆栈指针"加载到SP寄存器120中而建立。最初,管理者堆栈指针可以指定管理者堆栈的初始在存储器(例如,RAM164)中的地址(即,管理者堆栈的基址)。在执行管理者代码期间,管理者堆栈指针由核心110保持。更具体地,在管理者代码将数据推入(push)管理者堆栈以及将数据从管理者堆栈弹出时,SP寄存器120中的管理者堆栈指针相应地递增和递减。在其它实施例中,核心110可以使用以不同方式操作的硬件堆栈。无论哪种方式,SP寄存器120中的管理者堆栈指针总是应该指向管理者堆栈上的顶部元素。

[0033] 核心110还可以建立至少一个其它堆栈以供用户代码(例如,RAM164中的)使用。这样堆栈在此称为"用户堆栈"(例如,图2的用户堆栈229),并且当核心110在执行用户代码时,核心110也保持用于用户堆栈的堆栈指针(在此称为"用户堆栈指针")。为了便于描述,在此讨论单一用户堆栈,然而其它实施例可以实现多个用户堆栈。

[0034] 因为核心110可以在执行管理者代码和用户代码之间转换,并且因为存在与管理者代码和用户代码两者都关联的堆栈(并因此,存在与管理者代码和用户代码两者都相关联的堆栈指针),所以多种不同实施例都包括用于在管理者状态和用户态之间的转换期间交换管理者堆栈指针和用户堆栈指针的机构。在一个实施例中,系统100包括"替代"SP寄存器135,该"替代"SP寄存器135可以被用于当核心110处于不同的状态时存储管理者堆栈指针或用户堆栈指针。换句话说,当核心110处于管理者状态时,管理者堆栈指针被保持在核心的SP寄存器120中,而用户堆栈指针被保持在替代SP寄存器135中。在核心110从管理者状态转换到用户状态时,SP寄存器120中的管理者堆栈指针的值与替代SP寄存器135中的用户堆栈指针的值交换。当核心110处于用户状态时,用户堆栈指针被保持在核心的SP寄存器120中,而管理者堆栈指针被保留在替代SP寄存器135中。在核心110从用户状态转换到管理者状态时,SP寄存器120中的用户堆栈指针的值与替代SP寄存器135中的管理者堆栈指针的值进行交换。换句话说,在任何给定时间,存储在SP寄存器120中的堆栈指针对应于与当前权限状态相关联的堆栈。

[0035] 在一个实施例中,核心110包括被配置来在管理者状态和用户状态之间的转换期间将存储在SP寄存器120和替代SP寄存器135中的堆栈指针进行交换的模块(在图1示出为SP交换模块128)。更具体地,在一个实施例中,核心110包括这样的模块(例如,SP交换模块128),其被配置为在进行当前权限级别和不同的权限级别之间的转换时将堆栈指针从SP寄存器120移入到替代SP寄存器135中,并且同时将堆栈指针从替代SP寄存器135移入到SP寄存器120中。根据一个实施例,SP寄存器120和替代SP寄存器135中的堆栈指针可以被调换(或"交换"),例如,通过直接信号连接152。在一个实施例中,SP交换模块128可以形成程序

控制单元114的一部分。在其它实施例中，用于执行此功能的硬件可以位于其它位置。

[0036] 在一个实施例中，为了实现执行保护，存储指令的存储器(例如，闪存存储器160或代码存储器210)被划分成多个区域，其中每个区域对应于一权限级别(例如，一个或多个区域对应于管理权限级别，以及一个或多个不同的区域对应于用户权限级别)。相应地，核心110可以基于指令存储在存储器中(即，指令存储的区域中)的地址，确定每个指令的权限级别。

[0037] 在一个实施例中，通过使用至少一个“指令地址范围指示器”定义代码存储器区域(被一个或多个分隔分开)。地址范围指示器定义被分配用于存储与管理者状态相关联的计算机指令(“管理者代码”)的一个或多个第一存储器区域，并且定义被分配用于存储与用户状态相关联的计算机指令(“用户代码”)的一个或多个第二存储器区域。在更具体的实施例中，指令地址范围指示器包括定义与管理者代码相关联的存储器的第一区域和与用户代码相关联的存储器的第二区域之间的边界(例如，在闪存存储器160或代码存储器210中)的地址。存储在小于(或者，在另一个实施例中，等于或小于)指令地址范围指示器的地址中的任何指令对应于管理者代码，而存储在等于或大于(或者，在另一个实施例中，大于)指令地址范围指示器的地址中的任何指令对应于用户代码。在替代实施例中，也可以是相反的情况。无论哪种方式，在一个实施例中，提取模块124(或其它电路)被配置为：当下一个指令提取地址落入分配给管理者代码的存储器区域的其中一个时，确定下一个指令提取地址(例如，基于IFAR117中的值的地址)处于与管理者状态相关联的存储器区域(例如，闪存存储器160或代码存储器210中的区域)，或者，当下一个指令提取地址落入分配给用户代码的存储器区域的其中一个时，确定下一个指令提取地址与用户状态相关联。

[0038] 在一个实施例中，指令地址范围指示器指定用于用户代码在代码存储器(例如，闪存存储器160或代码存储器210)中存储的基址。替代地，指令地址范围指示器可以指定用于管理者代码在代码存储器中存储的最高地址(上边界)。无论哪种方式，指令地址范围指示器都定义了存储器中在管理者代码区域和用户代码区域之间的边界。以下的描述讨论了一个实施例，在该实施例中，指令地址范围指示器包括指定用于用户代码在代码存储器内(例如，闪存存储器160或代码存储器210内)的存储的基址的地址，并且该地址在此被称为“用户代码基址”。在一个实施例中，在系统的初始化代码的执行期间，在用户代码BAR132内建立(存储)用户代码基址。在其中多个不连续的存储器区域被分配给管理者代码和/或用户代码的实施例中，多个指令地址范围指示器可以存储在系统100中(例如，在多个寄存器中)。这样的实施例也被包括在本发明主题的范围中，但在此中不作详细讨论。

[0039] 在一个实施例中，为了实现资源保护，将核心110可以访问的资源(例如，RAM164、外围模块140、142等等)划分成组，每组都与权限等级对应(例如，一个或多个组对应于管理权限级别，并且一个或多个组对应于用户权限级别)。相应地，核心110(或者更具体地，访问模块126)可以确定与一权限级别相关联的指令是否在试图访问具有与该指令相同或不同的权限级别的资源(例如，生成数据引用)。

[0040] 在一个实施例中，存储器类型资源(例如，RAM164或数据存储器220)可以被划分成一个或多个管理者数据区域以及划分成一个或多个用户数据区域，并且通过使用至少一个“数据地址范围指示器”来定义这些区域。数据地址范围指示器定义被分配用于存储与管理者状态相关联的数据(“管理者数据”)的一个或多个第一存储器区域，以及定义被分配用于

存储与用户状态相关联的数据(“用户数据“)的一个或多个第二存储器区域。在更具体的实施例中,数据地址范围指示器包括定义在与管理者数据相关联的第一存储器区域和与用户数据相关联的第二存储器区域之间的边界(例如,在RAM164或数据存储器220中)的地址。核心110可以允许管理者代码访问存储在任一存储器区域中的任何数据(例如,管理者数据和用户数据两者),但核心110也可以仅允许用户代码访问存储在等于或大于(或者,在另一个实施例中,大于)地址范围指示器的地址中的数据(例如,仅用户数据)。在替代实施例中,核心110可以允许用户代码仅访问存储在小于(或者,在另一个实施例中,小于或等于)地址范围指示器的地址中的数据。无论哪种方式,在一个实施例中,访问模块126(或其它电路)被配置为:当数据访问地址落入分配给管理者数据的存储器区域中的一个时,确定数据访问地址与管理者状态相关联,以及当数据访问地址落入分配给用户数据的存储器区域中的一个时,确定数据访问地址与用户状态相关联。

[0041] 在一个实施例中,数据地址范围指示器指定用于用户数据在存储器资源(例如,在RAM164或数据存储器220内)内存储的基址。替代地,数据地址范围指示器可以指定用于管理者数据在存储器资源内存储的最高地址(上边界)。无论哪种方式,数据地址范围指示器定义了在存储器中在管理者数据空间和用户数据空间之间的边界。以下的描述讨论了一个实施例,在该实施例中,数据地址范围指示器包括指定用于用户代码在存储器资源内(例如,RAM164内)的存储的基址的地址,并且该地址在此被称为“用户数据基址”。在一个实施例中,在系统的初始化代码的执行期间在用户数据BAR133内建立(存储)用户数据基址。在其中多个不连续存储器区域被分配给管理者数据和/或用户数据的实施例中,多个数据地址范围指示器可以被存储在系统100中(例如,在多个寄存器中)。这样的实施例被包括在本发明主题的范围内,但在此不再详细讨论。

[0042] 在一个实施例中,外围资源(例如,外围模块140、142,以及其它外围装置(未说明))可以被划分成一个或多个管理外围组或划分成一个或多个用户外围组,并且通过使用至少一个“外围地址范围指示器”来定义这些组。在更具体的实施例中,外围地址范围指示器包括定义在对于管理者代码可以访问的第一组外围装置(例如,外围模块140是其一部分)和对于管理者代码和用户代码两者可以访问的第二组外围装置之间的边界的地址。核心110可以允许管理者代码访问任意外围装置(例如,对管理者代码可访问的外围装置没有限制),但核心110可以仅允许用户代码访问具有等于或大于(或者,在另一个实施例中,大于)外围地址范围指示器的地址的外围装置。在替代实施例中,核心110可以仅允许用户代码访问具有小于(或者,在另一个实施例中,等于或小于)地址范围指示器的地址的外围装置。

[0043] 在一个实施例中,外围地址范围指示器指定用户代码可以访问的外围装置的最小地址。替代地,外围地址范围指示器可以指定仅仅管理者代码可以访问的外围装置的最大地址。替代地,可以使用多个外围地址范围指示器来标识对于用户代码、管理者代码或这两者可以访问的多组外围装置的地址。无论哪种方式,外围地址范围指示器定义了仅仅对于管理者代码可访问的一组外围装置,以及对于管理者代码和用户代码两者可访问的一组外围装置。下面的描述说明了一个实施例,在该实施例中,外围地址范围指示器包括指定了特定外围装置地址的地址,并且该地址在此被称为“用户外围基址。”在一个实施例中,用户外围基址在系统的初始化代码的执行期间被建立(被存储)在用户外围BAR134内。

[0044] 提供图2以使用保护寄存器130更全面地说明资源划分的概念。图2是根据一个示例实施例的划分的代码存储器210(例如,闪存存储器160)和划分的数据存储器220(例如,RAM164)的简化的例子。图2还示出了用户代码BAR212(例如,用户代码BAR132)和用户数据BAR222(例如,用户数据BAR133)。虽然在图2中未示出外围装置被划分成与管理者状态和用户状态相关联的组,但是通过本文中其它地方的描述应理解外围装置划分。

[0045] 在一个实施例中,结合资源保护,指令地址范围指示器被存储在用户代码BAR212中(例如,在系统初始化期间或在其它时间),并且指令地址范围指示器定义在被分配用于存储管理者代码的第一存储器区域216(“管理者代码存储器区域)和被分配用于存储用户代码的第二存储器区域218(“用户代码存储器区域”)之间的分隔214。在系统运行期间,核心(例如,核心110)将要提取的每个下一个指令的地址(例如,基于IFAR117中的值的地址)和指令地址范围指示器进行比较。当比较表明要提取的下一个指令地址落入第一存储器区域216内时,核心确定要提取的下一个指令地址是管理者代码。相反,当比较表明要提取的下一个指令地址落入第二存储器区域218内时,核心确定要提取的下一个指令地址是用户代码。

[0046] 然后,核心可以确定是否允许提取下一个指令和/或是否允许提取所需的状态变化。如前面所提到的,当下一个指令提取地址(例如,基于IFAR117中的值的地址)不处于与当前权限状态相关联的存储器区域内的时候,核心可以被配置为仅仅当从当前权限状态到不同的权限状态的转换被允许时允许发生状态变化以及提取下一个计算机指令。更具体地,在一个实施例中,用户状态和管理者状态之间的状态变化仅仅在特定条件下被允许。

[0047] 在一个实施例中,在当前权限状态是管理者状态,并且下一个指令提取地址处于用户代码存储器区域218内时,核心可以仅响应于“合法的管理者-到-用户状态转换事件”允许从管理者状态到用户状态的“正常”转换(以及可以允许提取下一个指令)。例如,合法的管理者-到-用户状态转换事件可以包括第一组指令(例如,包括从中断返回指令(a return from interrupt instruction))的一个的核心的执行,这导致下一个指令提取地址(例如,基于IFAR117中的值的地址)被更新。相反,当下一个指令提地址被响应于合法的管理者-到-用户状态转换事件以外的任何事情(例如,不在所述第一组指令中的指令的核心的执行)而更新时,核心可以不允许状态变化(并且不允许提取下一个指令)。

[0048] 根据另一个实施例,在当前权限状态是用户状态,并且下一个指令提取地址处于用户代码存储器区域216内时,核心可以仅响应于“合法的用户-到-管理者状态转换事件”允许从用户状态到管理者状态的“正常”转换(以及可以允许提取下一个指令)。例如,合法的用户-到-管理者状态转换事件可以包括:1)第二组指令(例如,包括软件中断指令、陷阱指令等等)中的一个的核心的执行,这导致下一个指令提取地址被更新;和/或2)检测到导致核心放弃(至少暂时地)其当前执行线程并且使控制返回到管理者代码(例如,到被配置来处理异常情况的内核的中断服务例程)的异常情况。例如,异常情况可以包括I/O中断信号的断言(assertion)或某些其它事件。除了从用户状态到管理者状态的“正常”转换之外,核心也可以实现“基于故障的”从用户状态到管理者状态的转换(例如,响应于错误或故障情形的检测,如稍后将结合图3的框313、322讨论)。

[0049] 结合资源保护,在一个实施例中,数据地址范围指示器被存储在用户数据BAR222(例如,在系统初始化期间或在其它时间),并且数据地址范围指示器定义在被分配用于存

储管理者数据226的第一存储器区域(“管理者数据存储器区域”)和被分配用于存储用户数据228的第二存储器区域(“用户数据存储器区域”)之间的分隔224。在系统运行期间,核心(例如,核心110)将每个所请求的数据地址的地址与数据地址范围指示器进行比较。当比较表明要访问的数据落入第一存储器区域226内时,核心确定要访问的数据是管理者数据。相反,当比较表明要访问的数据地址范围指示器落入第二存储器区域228内时,核心确定要访问的数据是用户数据。

[0050] 然后核心可以确定是否允许或拒绝数据访问。如前面所述的,管理者代码可以访问存储在任一类型存储器区域的数据(例如,存储在两个存储区域226、228中的管理者数据和用户数据),但用户代码仅可以访问存储在用户数据存储器区域中的数据(例如,仅仅存储在存储器区域228中的用户数据)。因此,在当前权限状态是管理者状态时,核心可以允许执行任何请求的数据访问。相反,在当前权限状态是用户状态并且指令包括对管理者数据存储器区域226内的数据的引用时,核心将不允许数据访问。在当前权限状态是用户状态并且指令包括对用户数据存储器区域228内的数据的引用时,核心将允许数据访问。

[0051] 在一个实施例中,管理者数据存储器区域226的一部分被分配给管理者堆栈227,而数据存储器区域228的一部分被分配给用户堆栈229。换句话说,管理者数据存储器区域226包括管理者堆栈227,而数据存储器区域228包括用户堆栈229。如前面所讨论的,系统保持管理者堆栈指针以指示管理者堆栈的顶部的地址,以及用户堆栈指针以指示用户堆栈的顶部的地址。当核心处于管理者状态时,管理者堆栈指针被存储在核心的堆栈指针寄存器(例如,图1的SP寄存器120)中,而当核心处于用户状态时,用户堆栈指针被存储在核心的堆栈指针寄存器中。对应于不是当前状态的任何状态的堆栈指针被保持在替代堆栈指针寄存器(例如,图1的替代SP寄存器135)中。

[0052] 图3是根据一个示例实施例的操作实现资源和执行保护的系统的方法的流程图。为了进一步理解,图3应结合图4观看;图4是示出了根据示例性实施例的管理者状态410态和用户状态420以及它们之间的转换的简化状态图。在图3所示的操作流程的描述中,将参考图4的管理者状态410态和用户状态420。为了便于理解,图3的流程图并未示出可以在实现流水线架构的系统中采用的并行性。例如,虽然流程图示出了更新IFAR(块306)、提取和译码指令(块314)、以及以连续的方式执行指令(块318)的步骤,但是应理解,一旦第一指令通过指令流水线开始进行,这些处理中的某些处理可以结合执行指令序列并行地进行。本领域技术人员将理解如何在包括流水线架构的系统中实现图3流程图中内涵的概念。

[0053] 在块302中,所述方法可以例如在核心(例如,图1的核心110)被加电(或复位)时开始,并且系统通过执行初始化代码而被初始化。如在此所使用的,“初始化代码”是指任何由核心执行以初始化系统直到当系统准备好执行正常的操作(例如,包括执行用户代码)的时间点的代码。例如,系统初始化可以包括执行BIOS代码、开机实时测试(POST)代码、引导程序(bootstrap)代码、以及负责引导操作系统的内核代码。更具体地,在加电时,在IFAR和PC寄存器中建立第一指令的地址(对应于复位向量的地址被加载到IFAR117中,并且相应的地址被加载到PC寄存器118中,见图1),并且相应的指令被从存储器中提取、译码和执行。随着每个指令通过执行流水线进行,IFAR和PC寄存器中的地址被更新,并且下一个指令被提取、译码和执行。最终,在初始化期间,核心开始执行内核。在一个实施例中,初始化代码被认为是管理者代码,并且核心初始将当前权限状态定义为管理者状态410。稍后,如在下面将要

讨论的，核心基于地址范围指示器(一个或多个)以及从存储器中从其提取当前在执行的指令的指令地址，定义当前权限状态为管理者状态410或用户状态420。

[0054] 在一个实施例中，通过初始化代码执行的一个处理过程被配置为通过如下来实现资源和执行保护：初始化被用于这些目的的寄存器(例如，图1的保护寄存器130)中的值。因此，如下面所描述的，初始化代码将值存储在这些寄存器值中。更特别的，以及如前面所讨论的，保护寄存器之一包括控制寄存器(例如，图1的控制寄存器131)，其包括一个或多个寄存器锁定标志和使能标志。在一个实施例中，在复位时，使能标志被设置(假定资源和执行保护使能信号已被提供给系统)，并且寄存器锁定标志(或多个寄存器锁标志)被清除。然后初始化代码可以将值写入到各种保护寄存器(例如，图1的寄存器132-134)中。根据多种实施例，利用设置的使能标志，核心将实现资源和执行保护。

[0055] 为了配置系统的资源和执行保护能力，初始化代码在保护指标寄存器中存储至少一个指令地址范围指示器和至少一个数据地址范围指示器。例如，在一个实施例中，初始化代码可以在用户代码基址寄存器(例如，图1的用户代码BAR132)中存储指令地址范围指示器(例如，引用图1的闪存存储器160的地址)，所述指令地址范围指示器定义在存储器(例如，图1的闪存存储器160)中在管理者代码区域和用户代码区域之间的分隔。更具体地，以及如前面所讨论的，指令地址范围指示器定义被分配用于存储与管理者状态(例如，管理者状态410)相关联的计算机指令的存储器区域，并且定义被分配用于存储与用户状态(例如，用户状态420)相关联的计算机指令的存储器区域。此外，初始化代码可以在数据代码基址寄存器(例如，图1的用户数据BAR133)中存储数据地址范围指示器(例如，引用图1的RAM164的地址)，所述数据地址范围指示器定义存储器(例如，图1的RAM164)中在管理者代码区域和用户代码区域之间的分隔。更具体地，以及如前面所讨论的，数据地址范围指示器定义被分配用于存储与管理者状态(例如，管理者状态410)相关联的数据的存储器区域，并且定义被分配用于存储与用户状态(例如，用户状态420)相关联的数据的存储器区域。最后，初始化代码可以将外围地址范围指示器(例如，外围地址)写入到外围代码基址寄存器(例如，图1的用户外围BAR134)，所述外围地址范围指示器定义了对于管理者代码可以访问的一组外围装置和对于用户代码可以访问的一组外围装置。

[0056] 在一个实施例中，系统被配置为使得管理者代码可以访问管理者和用户数据区域及外围装置，而用户代码仅可以访问用户数据区域及外围装置。一旦定义了管理者和用户代码区域、数据区域、以及外围装置的保护寄存器被初始化，初始化代码可以设置相应的寄存器锁定标志(一个或多个)(例如，在图1的控制寄存器131中)，以使得保护寄存器中的值不被无意地重写。在一个实施例中，保护寄存器处于被指定为仅仅对管理者代码可访问的外围装置中(例如，外围地址范围指示器使得用于保护寄存器的外围装置在仅仅对于管理者代码可访问的外围装置的组中)。因此，保护寄存器对于用户代码不是可访问的。

[0057] 在块304，初始化代码还通过将管理者堆栈指针写入到堆栈指针寄存器(例如，图1的SP寄存器120)中来建立管理者堆栈(例如，图2的管理者堆栈227)。如前面所讨论的，初始的管理者堆栈指针可以指定存储器中(例如，在RAM164中)与管理者堆栈的初始对应的地址(例如，图2的管理者堆栈的基址227)。在一个实施例中，初始化代码也可以在该点建立用户堆栈(例如，图2的用户堆栈229)，然而用户堆栈也可以稍后(例如，在初始化代码已经执行之后，诸如就在用户代码的第一实例执行之前)建立。无论哪种方式，在核心上执行的管理

者代码可以通过将与用户堆栈的初始对应的地址(例如,图2的用户堆栈229的基址)存储在替代堆栈指针寄存器(例如,图1的替代SP寄存器135)中来建立用户堆栈。如在下面的讨论中将要阐明的,在核心堆栈指针寄存器(例如,图1的SP寄存器120)中的堆栈指针对应于用于当前权限状态的堆栈,而替代堆栈指针寄存器(例如,图1的替代SP寄存器135)中的堆栈指针对应于当前权限状态以外的权限状态的堆栈。换句话说,例如,在当前权限状态是管理者状态时,核心的堆栈指针寄存器包含用于管理者堆栈的堆栈指针,而替代堆栈指针寄存器可以包含用于用户堆栈的堆栈指针。相反,在当前权限状态是用户状态时,核心的堆栈指针寄存器包含用于用户堆栈的堆栈指针,而替代堆栈指针寄存器包含用于管理者堆栈的堆栈指针。

[0058] 一旦系统初始化完成,核心可以进入正常操作模式,在该模式下,核心根据程序流提取、译码、以及执行指令。更具体地,在块306,当指令进行执行流水线时,在每个指令提取之前,核心更新IFAR(例如,图1的IFAR117)中的地址以对应于下一个指令提取地址,并且稍后更新PC寄存器(例如,图1的PC寄存器118)中的地址。如果先前指令中没有指定分支或跳转,则在适当时间,核心分别以指令提取宽度和指令长度递增IFAR和PC寄存器中的地址。否则,如果先前指令指定了分支或跳转到存储器中的处于非连续地址的位置,则核心更新IFAR中的地址以对应于指定的地址。一旦IFAR中的地址已被更新,核心估算下一个指令提取地址。

[0059] 在块308,作为所述估算的一部分,核心确定下一个指令提取地址是否指示应当从与当前权限状态对应的代码存储器的区域,还是从与不同的权限状态对应的代码存储器的区域,来访问下一个指令。这可以通过如下(例如,通过图1的提取模块124)来实现:将下一个指令提取地址与指令地址范围指示器(例如,在图1的用户代码BAR132中)进行比较以确定要提取的下一个指令被存储的区域,并然后确定该区域是否与当前权限状态相关联。例如,在当前权限状态是管理者状态410,并且下一个指令提取地址处于分配给用户代码的代码存储器的区域(例如,图2的区域218)中时,核心确定下一个指令提取地址不对应于与当前权限状态相关联的代码存储器的区域。同样地,在当前权限状态是用户状态420,并且下一个指令提取地址处于分配给管理者代码的代码存储器的区域(例如,图2的区域216)中时,核心确定下一个指令提取地址不对应于与当前权限状态相关联的代码存储器的区域。

[0060] 当下一个指令提取地址指示下一个指令应该从与不同的权限状态对应的代码存储器的区域取得时,在块310中,进一步确定(例如,通过图1的提取模块124)状态转换是否是合法的状态转换。如前面所讨论的,例如,当合法的管理者到用户状态转换事件已经发生(例如,核心执行了第一组指令中的一个,包括从中断返回指令)时,核心可以确定从管理者状态410到用户状态420的转换被允许。同样地,当合法的用户到管理者状态转换事件已经发生(例如,核心执行了第二组指令中的一个,包括软件中断指令、陷阱指令等等),和/或检测到导致核心放弃其当前执行线程并且将控制传回到管理者代码(例如,I/O中断信号的断言)的异常情况时,核心可以确定从用户状态420到管理者状态410的转换被允许。在图4中通过箭头430、432指示出合法的管理者到用户以及用户到管理者的状态转换。

[0061] 当核心确定状态转换是合法的(如在块310确定的)时,在块312中,通过(例如,由图1的状态模块122)将当前权限状态改变为与要提取的下一个指令(例如,结合块310估算的指令)相关联的权限状态,来执行状态转换。此外,存储在核心的堆栈指针寄存器(例如,

图1的SP寄存器120)中的堆栈指针与存储在替代堆栈指针寄存器(例如,图1的替代SP寄存器135)中的堆栈指针进行交换(例如,通过图1的SP交换模块128)。

[0062] 当在块310确定状态转换是不合法的时,核心迫使当前权限状态变为管理者状态(如果它不是已经是的话),并且在块313,故障处理管理者代码产生异常故障。例如,在当前权限状态是管理者状态410并且管理者代码试图使用非法指令(例如,指令不是从中断返回类型的指令)跳转到用户代码时,核心保持处于管理者状态410(如图4中箭头434所示),并且产生执行故障。相反,例如,在当前权限状态是用户状态420并且用户代码试图使用非法指令(例如,指令不是软件中断类型的指令)跳转到管理者代码时,核心执行从用户状态420到管理者状态410的转换(如图4中箭头436所示),并且产生执行故障。结合状态转换,核心的堆栈指针寄存器(例如,图1的SP寄存器120)中的堆栈指针与替代堆栈指针寄存器(例如,图1的替代SP寄存器135)中的堆栈指针交换。

[0063] 在一个实施例中,执行故障的产生包括执行存储与非法指令相关联的地址(例如,PC寄存器118中与非法指令对应的地址)的管理者代码(例如,内核故障处理例程)。例如,所述地址,连同可能与故障报告或处理相关的其它信息,可以存储在保护寄存器(例如,图1的非法PC寄存器136)中。假定故障可以被处理而没有显著地中断程序流,则方法可以返回到块306以更新IFAR中的地址,并且继续执行与所述程序流相关联的指令。否则,方法可以结束。

[0064] 再次参照块308,在当前权限状态是管理者状态410,并且下一个指令地址处于分配给管理者代码的核心存储器的区域(例如,图2的区域216)中时,核心确定下一个指令地址对应于与当前权限状态相关联的代码存储器的区域,并且方法进行到块314。同样,在当前权限状态是用户状态420,并且下一个指令地址处于分配给用户代码的核心存储器的区域(例如,图2的区域218)中时,核心确定下一个指令地址对应于与当前权限状态相关联的代码存储器的区域,并且方法进行到块314。在无论哪种情况下,如图4的箭头438、440所示的,当前权限状态没有被改变。

[0065] 当核心已确定(在块308)下一个指令地址对应于与当前权限状态相关联的代码区域时,或者,当已经进行了合法的状态转换(在块312)时,在块314,核心可以提取与IFAR中的下一个指令提取地址对应的指令,并且可以译码所述指令。

[0066] 在块316,核心然后可以确定该指令的执行是否将产生数据访问。例如,指令可以包括对数据在数据存储器中(例如,在图1的RAM164中)存储的位置的引用。当指令的执行将不产生数据访问时,在块318,核心执行指令,并且方法如所示地重复(例如,以继续执行与程序流相关联的指令)。

[0067] 当指令的执行产生数据访问时,核心(例如,图1的访问模块126)确定数据访问地址是否处于在当前权限状态下允许访问的数据存储器中(例如,图1的RAM164)的区域。例如,如前面所讨论的,管理者代码可以被允许访问管理者和用户数据区域(例如,图2的区域226、228)两者中的数据,而用户代码可以被允许仅仅访问在用户数据区域(例如,图2的区域228)中的数据。因此,在当前权限状态是管理者状态时,核心可以允许任何数据访问。相反,在当前权限状态是用户状态时,核心可以仅在数据访问地址处于用户数据区域中而不是在管理者数据区域中(例如,图2的区域228)时,允许数据访问。

[0068] 在一个实施例,在数据访问地址和数据地址范围指示器(例如,在图1的用户数据

BAR133中)之间进行比较(例如,通过图1的访问模块126),以确定数据访问地址所处的区域。然后,确定当核心处于当前权限状态(例如,处于与译码的指令对应的权限状态)时,是否允许该区域中的数据访问。当数据访问被允许时,在块318,核心执行指令(包括执行数据访问),并且方法如所示地迭代重复(例如,程序计数器在块306被更新,并且方法继续进行)。

[0069] 在块320确定当数据访问地址处于在当前权限状态下不允许访问的数据存储器(例如,图1的RAM164)中的区域时,在块322,核心迫使当前权限状态变为管理者状态(如果它不是已经是的话),并且故障处理管理者代码产生访问故障。例如,在当前权限状态是用户状态420,并且用户代码试图访问管理者数据区域(例如,图2的区域226)中的数据时,核心执行从用户状态420到管理者状态410的转换(如图4中箭头436所示的),数据访问被禁止,并且产生访问故障。结合状态转换,核心的堆栈指针寄存器(例如,图1的SP寄存器120)中的堆栈指针与替代堆栈指针寄存器(例如,图1的替代SP寄存器135)中的堆栈指针交换。

[0070] 在一个实施例中,访问故障的产生包括执行存储与非法指令相关联的地址的管理者代码(例如,内核故障处理例程)。例如,所述地址,连同可能与故障报告或处理相关的其它信息,可以被存储在保护寄存器(例如,图1的故障PC寄存器137)中。在一个替代实施例中,故障数据地址可以被存储在寄存器137(例如,寄存器137可以是“非法数据地址”寄存器)中。假定故障可以被处理而没有显著地中断程序流,则方法可以返回到块306以继续执行与所述程序流相关联的指令。否则,方法可以结束。

[0071] 应理解,图3中描述的过程块中的一些可以彼此并行地或者与其它处理过程的执行并行地被进行。此外,应理解,图3中描述的处理块的特定顺序可以被修改,同时基本上实现相同的结果。因此,这样的修改应被包括在本发明主题的范围内。

[0072] 虽然上面讨论的实施例主要集中于其中实现两个权限级别(例如,管理者级别和用户级别)的系统,但是应理解,这些实施例也可以在其中实现多于两个权限级别的系统(例如,实现了具有三个或更多个权限环的基于环的安全方案的系统)中实现。基于这里的描述,本领域技术人员将理解如何修改各种描述的实施例以在支持多于两个的权限级别的系统中实现实施例。此外,虽然上面结合图1描述了特定的系统配置,然而实施例也可以在具有其它架构的系统中实现。例如,虽然系统100包括用于存储结合资源和执行保护使用的值的各种保护寄存器130,并且保护寄存器130是与核心110分离并且是对于核心110可通过总线150和/或其它连接152访问的模块140的一部分,但是其它实施例可以被配置为在其它地方存储这些值,包括核心110本身中。此外,实施例可以在多处理器系统和/或具有与图1的架构明显不同的架构的系统中实现。这些和其它变体被包括在本发明主题的范围中。

[0073] 如此,已经描述了资源和执行保护方法及装置的多种实施例。处理系统的一个实施例包括第一模块和第二模块。所述第一模块被配置为:基于取自存储器的指令的地址保持当前权限状态是第一权限状态还是第二权限状态的知识。所述第二模块被配置为:通过将下一个指令提取地址和至少一个指令地址范围指示器进行比较确定所述下一个指令提取地址是否对应于位于与所述当前权限状态相关联的第一存储器区域内的位置或与不同的权限状态相关联的第二存储器区域内的位置。当所述下一个指令提取地址不位于与所述当前权限状态相关联的所述第一存储器区域内时,仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时允许提取下一个计算机指令。

[0074] 在另一个实施例中，所述处理系统包括第三模块，被配置为：确定为计算机指令生成数据访问地址，通过将所述数据访问地址与至少一个数据地址范围指示器进行比较确定所述数据访问地址是与所述第一权限状态相关联还是与所述第二权限状态相关联，以及当所述当前权限状态是允许访问所述存储器位置的权限状态时允许访问与所述数据访问地址对应的存储器位置。

[0075] 处理方法的一个实施例包括：存储至少一个指令地址范围指示器，所述至少一个指令地址范围指示器定义一个或多个被分配用于存储与第一权限状态相关联的计算机指令的第一存储器区域，并且定义一个或多个被分配用于存储与第二权限状态相关联的计算机指令的第二存储器区域。所述方法还包括：基于取自存储器的指令的地址将当前权限状态定义为所述第一权限状态或所述第二权限状态；以及通过将下一个指令提取地址与所述至少一个指令地址范围指示器进行比较，确定所述下一个指令提取地址是对应于位于与所述当前权限状态相关联的存储器区域内的位置还是位于与不同的权限状态相关联的存储器区域内的位置。当所述下一个指令提取地址不对应于位于与所述当前权限状态相关联的存储器区域内的位置时，仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时，允许提取所述下一个计算机指令。

[0076] 在另一个实施例中，所述处理方法包括存储定义一个或多个被分配用于存储与所述第一权限状态相关联的数据的第三存储器区域并且定义一个或多个被分配用于存储与所述第二权限状态相关联的数据的第四存储器区域的至少一个数据地址范围指示器。所述方法还包括确定确定计算机指令生成数据访问地址。当生成所述数据访问时，通过将所述数据访问地址与所述至少一个数据地址范围指示器进行比较，确定所述数据访问地址是用于与所述第一权限状态相关联的存储器位置还是用于与所述第二权限状态相关联的存储器位置。当所述当前权限状态是允许访问与所述数据访问地址对应的存储器位置的权限状态时，允许访问所述存储器位置。

[0077] 一种处理方法的另一个实施例包括：将当前权限状态定义为管理者状态或用户状态；以及通过将下一个指令提取地址与指令地址范围指示器进行比较，确定所述下一个指令提取地址是对应于位于与所述当前权限状态相关联的存储器区域内的位置还是位于与不同的权限状态相关联的存储器区域内的位置。所述指令地址范围指示器定义被分配用于存储管理者代码的管理者代码存储器区域和被分配用于存储用户代码的用户代码存储器区域。当所述下一个指令提取地址不对应于位于与所述当前权限状态相关联的存储器区域内的位置时，仅仅在从所述当前权限状态到所述不同的权限状态的转换是合法时，允许提取所述下一个计算机指令。

[0078] 在另一个实施例中，所述处理方法还包括：确定计算机指令生成数据访问地址，以及当生成所述数据访问时，通过将所述数据访问地址与数据地址范围指示器进行比较，确定所述数据访问地址是用于与所述管理者状态相关联的存储器位置还是用于与所述用户状态相关联的存储器位置。所述数据地址范围指示器定义被分配用于存储与所述管理者状态相关联的数据的第三存储器区域，并且定义被分配用于存储与所述用户状态相关联的数据的第四存储器区域。当所述当前权限状态是所述管理者状态时，允许访问所述存储器位置。以及当所述当前权限状态是所述用户状态时，仅仅当所述数据访问地址是用于位于所述第四存储器区域内的存储器位置时，允许访问所述存储器位置。

[0079] 虽然上面已经结合特定系统、装置、以及方法对本发明主题的原则进行了描述，但应清楚理解，该描述仅仅是通过例子进行的，而不是对本发明主题范围的限制。在此所描述的以及附图中所示出的各种功能或处理块可以以硬件、固件、软件或由其任意组合来实现。此外，这里所采用的措辞或术语用于描述的目的而不是限制性的目的。

[0080] 对特定实施例的上述描述充分揭示了本发明主题的一般特性，其他人可以通过运用现有知识很容易地对其进行修改和/或调整以适合各种应用，而不脱离本发明的一般概念。因此，这些调整和修改是在所公开的实施例的等同物的内涵和外延之内。本发明主题涵盖所有这些落在所附权利要求的精神和宽泛范围内的替代物、修改、等价物、以及变化。

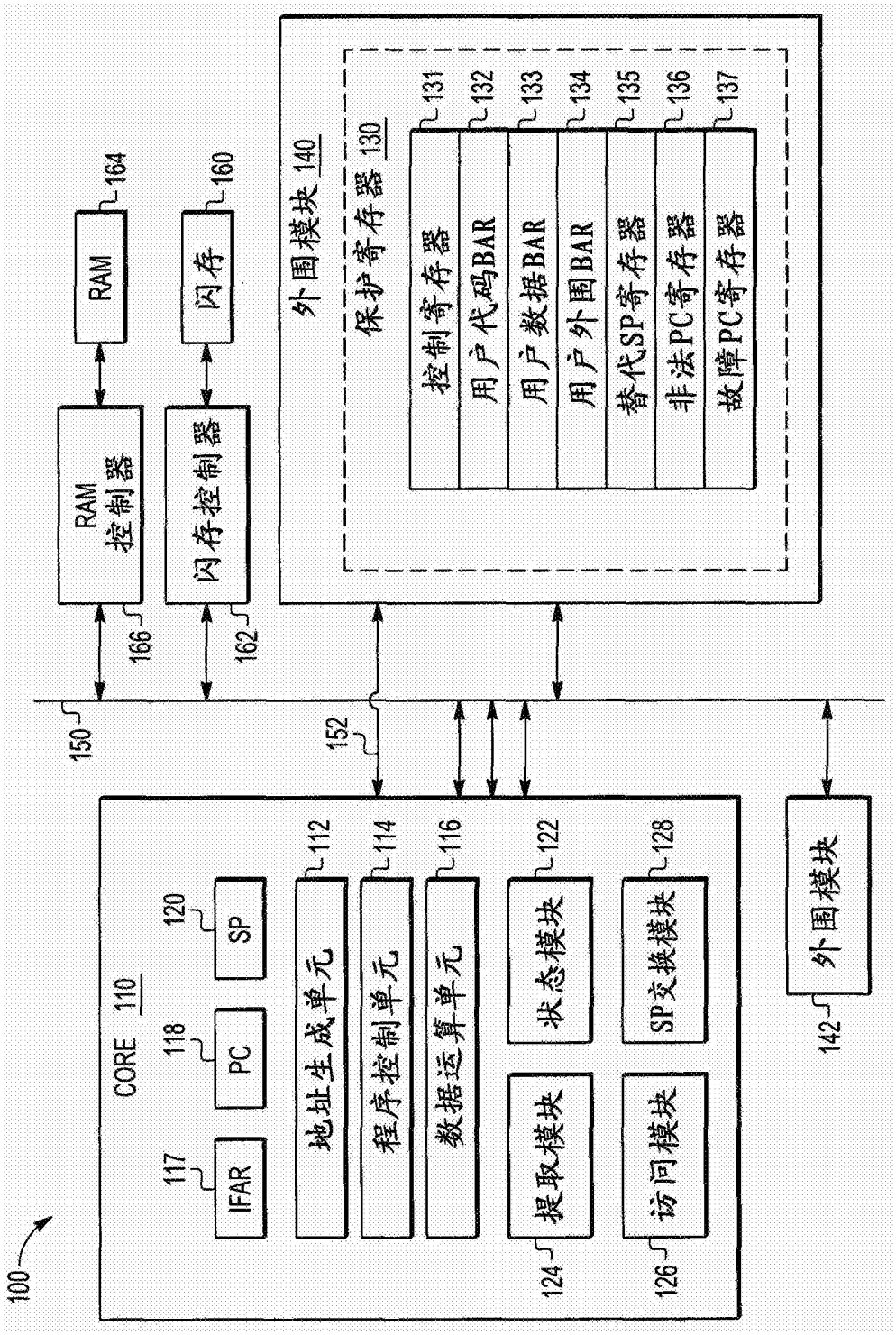

100
CORE 110
117 IFAR
118 PC
120 SP
112 地址生成单元
114 程序控制单元
116 数据运算单元
122 状态模块
124 提取模块
126 访问模块
128 SP交换模块
142 外围模块
150
152
外围模块 140
保护寄存器 130
131 控制寄存器
132 用户代码BAR
133 用户数据BAR
134 用户外围BAR
135 替代SP寄存器
136 非法PC寄存器
137 故障PC寄存器
RAM 控制器
166
闪存控制器
162
RAM
164
闪存
160

图1

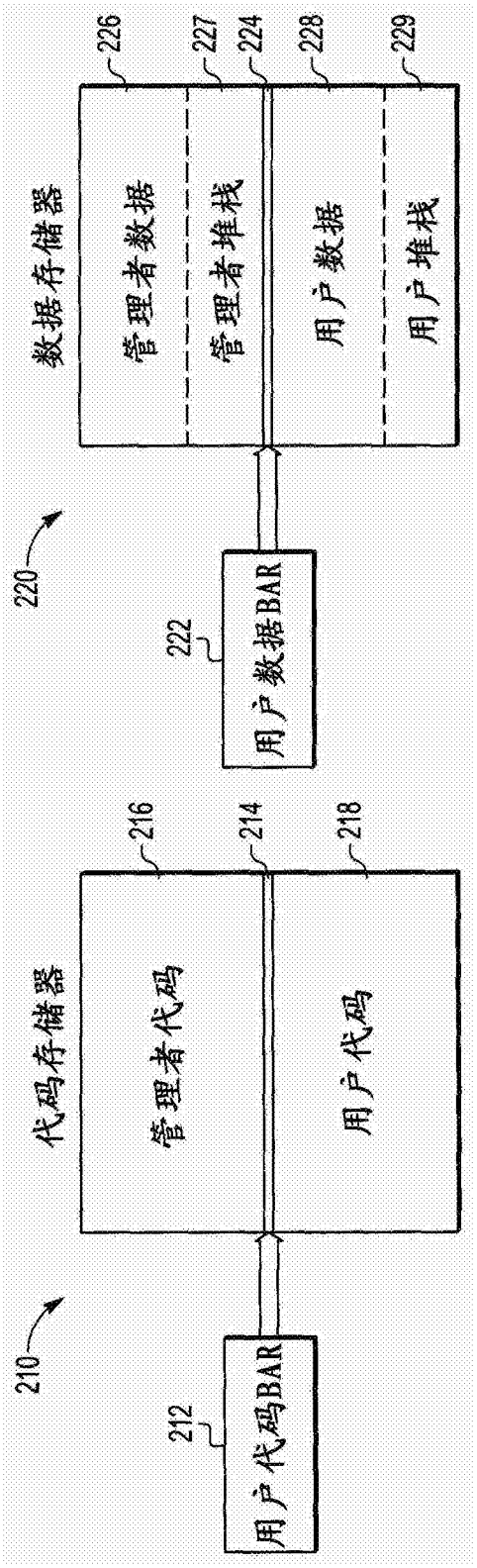

数据存储器
管理者数据
管理者堆栈
用户数据
用户堆栈
226
227
224
228
229
220
222
用户数据BAR
代码存储器
管理者代码
用户代码
216
214
218
210
212
用户代码BAR

图2

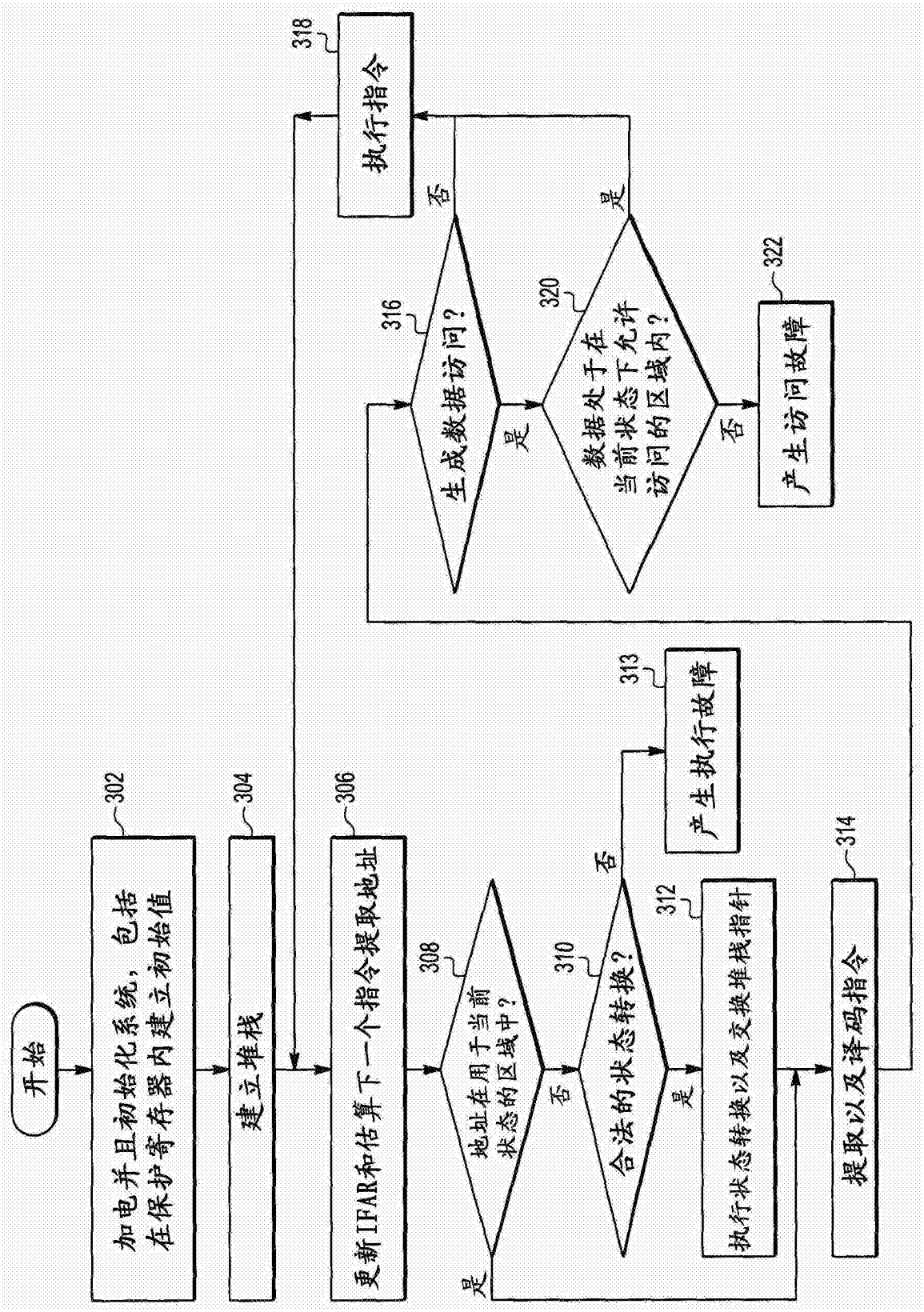

开始
302
加电并且初始化系统，包括在保护寄存器内建立初始值
304
建立堆栈
306
更新IFAR和估算下一个指令提取地址
308
地址在用于当前状态的区域中？
是
否
310
合法的状态转换？
是
否
312
执行状态转换以及交换堆栈指针
313
产生执行故障
314
提取以及译码指令
316
生成数据访问？
是
否
318
执行指令
320
数据处于在当前状态下允许访问的区域内？
是
否
322
产生访问故障

图3

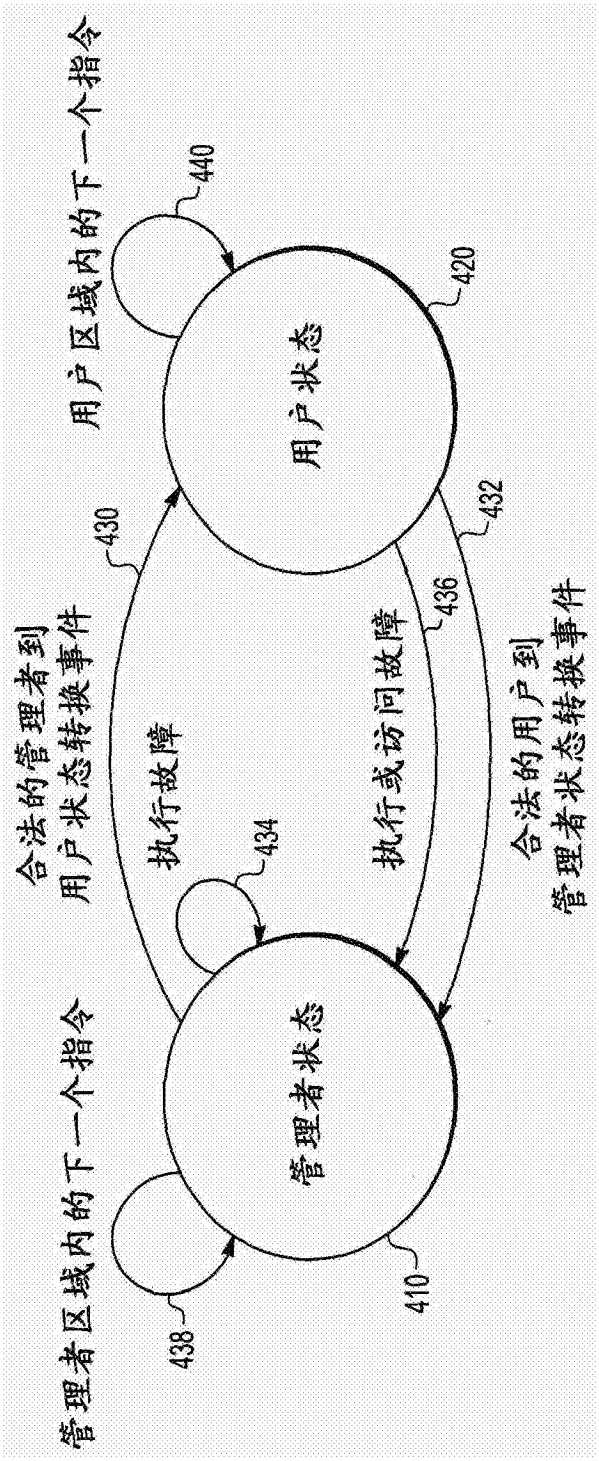

用户区域内的下一个指令
440
420
用户状态
合法的管理者到
用户状态转换事件
430
执行故障
434
执行或访问故障
436
432
合法的用户到
管理者状态转换事件
管理者区域内的下一个指令
管理者状态
438
410

图4