



(12)发明专利申请

(10)申请公布号 CN 107003965 A

(43)申请公布日 2017.08.01

(21)申请号 201580063904.1

(22)申请日 2015.11.12

(30)优先权数据

14/581677 2014.12.23 US

(85)PCT国际申请进入国家阶段日

2017.05.24

(86)PCT国际申请的申请数据

PCT/US2015/060424 2015.11.12

(87)PCT国际申请的公布数据

W02016/105684 EN 2016.06.30

(71)申请人 英特尔公司

地址 美国加利福尼亚州

(72)发明人 G.奈格 R.M.桑卡兰

(74)专利代理机构 中国专利代理(香港)有限公司 72001

代理人 姜冰 付曼

(51)Int.Cl.

G06F 13/34(2006.01)

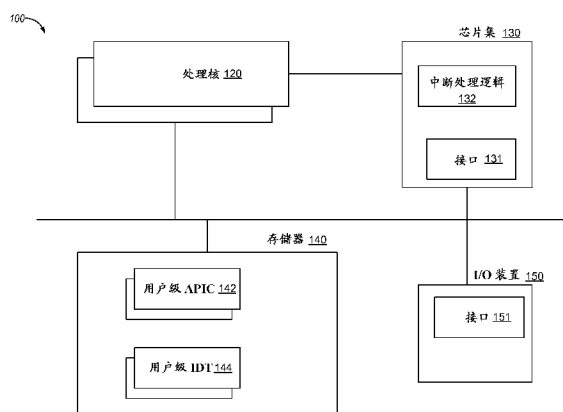
权利要求书3页 说明书20页 附图14页
按照条约第19条修改的权利要求书3页

(54)发明名称

向用户级应用传递中断

(57)摘要

提供用于向用户级应用传递中断的系统和方 法。示例处理系统包括：存储器，配置成存储与处理系统所执行的多个用户级应用对应的多个用户级中断处理程序地址数据结构和多个用户级APIC数据结构；以及处理核，响应接收到用户级中断的通知而配置成：设置与当前由处理核执行的用户级应用关联的用户级APIC数据结构中具有该用户级中断的标识符所定义的位置的挂起的中断比特标志，并且为该用户级APIC数据结构所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断来调用与该用户级应用关联的用户级中断处理程序地址数据结构所标识的用户级中断处理程序。



1. 一种处理系统,包括:

存储器,配置成存储与所述处理系统所执行的多个用户级应用对应的多个用户级中断处理程序地址数据结构和多个用户级高级可编程中断控制器 (APIC) 数据结构;以及

处理核,响应接收到用户级中断的通知而配置成:

设置与当前由所述处理核执行的用户级应用关联的用户级APIC数据结构中具有所述用户级中断的标识符所定义的位置的挂起的中断比特标志,以及

对于该用户级APIC数据结构所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断,调用与所述用户级应用关联的用户级中断处理程序地址数据结构所标识的用户级中断处理程序。

2. 如权利要求1所述的处理系统,其中,所述处理核还响应设置所述挂起的中断比特标志而配置成触发指示存在挂起的用户级中断的微架构事件。

3. 如权利要求2所述的处理系统,其中,所述处理核响应检测到指示存在所述挂起的用户级中断的所述微架构事件而配置成调用所述用户中断处理程序。

4. 如权利要求1-3中的任一项所述的处理系统,其中,所述多个用户级APIC数据结构的每个用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中所述比特图中的每个比特的位置对应于中断标识符。

5. 如权利要求1-3中的任一项所述的处理系统,其中,每个用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

6. 如权利要求1-3中的任一项所述的处理系统,其中,每个用户级中断处理程序地址数据结构包括与该用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

7. 如权利要求1-3中的任一项所述的处理系统,还包括处理逻辑,所述处理逻辑配置成:

响应将进入的中断标识为用户级中断而标识与当前由所述处理核执行的所述用户级应用关联的发布的中断描述符的地址;

设置与所述发布的中断描述符关联的比特图中所述中断的标识符对应的比特;以及
传送具有所述发布的中断描述符所标识的中断号的通知中断。

8. 如权利要求7所述的处理系统,其中,所述处理核还配置成将进入的中断的标识符与为所述通知中断所保留的标识符进行比较。

9. 如权利要求1-3中的任一项所述的处理系统,其中,调用所述用户级中断处理程序还包括:

将指令指针的当前值存储在栈上;以及

将所述用户级中断处理程序的地址加载到所述指令指针中。

10. 如权利要求1-3中的任一项所述的处理系统,其中,所述处理系统通过芯片上系统 (SoC) 来表示。

11. 一种方法,包括:

接收用户级中断的通知;

设置与当前由处理核执行的用户级应用关联的用户级APIC数据结构中与所述用户级中断的标识符相对应的挂起的中断比特;

使用所述用户级APIC数据结构来标识一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断;以及

使用与所述用户级应用关联的用户级中断处理程序地址数据结构来标识对于所标识的用户级中断的中断处理程序。

12. 如权利要求11所述的方法,还包括:

响应设置所述挂起的中断比特标志而触发指示存在挂起的用户级中断的微架构事件。

13. 如权利要求12所述的方法,还包括:

响应检测到指示存在所述挂起的用户级中断的所述微架构事件而调用所标识的用户中断处理程序。

14. 如权利要求13所述的方法,其中,调用所述用户级中断处理程序还包括:

将指令指针的当前值存储在栈上;以及

将所述用户级中断处理程序的地址加载到所述指令指针中。

15. 如权利要求11-14中的任一项所述的方法,其中,所述多个用户级APIC数据结构的每个用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中所述比特图中的每个比特的位置对应于中断标识符。

16. 如权利要求11-14中的任一项所述的方法,其中,每个用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

17. 如权利要求11-14中的任一项所述的方法,其中,每个用户级中断处理程序地址数据结构包括与所述用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

18. 如权利要求11-14中的任一项所述的方法,还包括:

响应将进入的中断标识为用户级中断而标识与当前由所述处理核执行的所述用户级应用关联的发布的中断描述符的地址;

设置与所述发布的中断描述符关联的比特图中所述中断的标识符对应的比特;以及
传送具有所述发布的中断描述符所标识的中断号的通知中断。

19. 如权利要求18所述的方法,还包括:

将进入的中断的标识符与为所述通知中断所保留的标识符进行比较。

20. 一种设备,包括:

存储器;以及

处理系统,耦合到所述存储器,所述处理系统用于执行如权利要求11-19中的任一项所述的方法。

21. 一种包括可执行指令的计算机可读非暂态存储媒体,所述可执行指令在由处理系统执行时使所述处理系统执行操作,包括:

接收用户级中断的通知;

设置与当前由所述处理核执行的用户级应用关联的用户级APIC数据结构中与所述用户级中断的标识符相对应的挂起的中断比特;

使用所述用户级APIC数据结构来标识一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断;以及

使用与所述用户级应用关联的用户级中断处理程序地址数据结构来标识对于所标识

的用户级中断的中断处理程序。

22. 如权利要求21所述的计算机可读非暂态存储媒体,还包括使所述处理系统响应设置所述挂起的中断比特标志而触发指示存在挂起的用户级中断的微架构事件的可执行指令。

23. 如权利要求22所述的计算机可读非暂态存储媒体,还包括使所述处理系统响应检测到指示存在所述挂起的用户级中断的所述微架构事件而调用所标识的用户中断处理程序的可执行指令。

24. 如权利要求23所述的计算机可读非暂态存储媒体,其中,调用所述用户级中断处理程序还包括:

将指令指针的当前值存储在栈上;以及

将所述用户级中断处理程序的地址加载到所述指令指针中。

25. 如权利要求21-24中的任一项所述的计算机可读非暂态存储媒体,其中,所述多个用户级APIC数据结构的每个用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中所述比特图中的每个比特的位置对应于中断标识符。

向用户级应用传递中断

技术领域

[0001] 本公开一般涉及处理系统,以及具体来说涉及用于向用户级应用传递中断的系统和方法。

背景技术

[0002] 中央处理器 (CPU) 外部的输入/输出 (I/O) 装置可通过工作在内核特权级的软件模块 (驱动程序) 来管理。来自 I/O 装置的通知可采取中断形式来传递给对应驱动程序。中断通常首先传递给操作系统 (OS) 内核,其然后可将控制传到适当的驱动程序。

附图说明

[0003] 本公开作为举例而不是限制来示出,并且可结合附图参照以下详细描述更全面了解,附图中:

图1示出按照本公开的一个或更多方面的示例处理系统的高级组件图;

图2示意示出按照本公开的一个或更多方面的与用户级线程关联的用户级中断处理程序 (handler) 地址数据结构的示例和示例用户级APIC数据结构;

图3示意示出按照本公开的一个或更多方面、通过由平台硬件发布用户级中断的示例方法所采用的示例数据结构;

图4示出按照本公开的一个或更多方面、用于由平台硬件发布用户级中断的示例方法的流程图;

图5A-5B示出按照本公开的一个或更多方面、用于向用户级应用传递中断的示例方法的流程图;

图6示出按照本公开的一个或更多方面的示例计算机系统的高级组件图;

图7示出按照本公开的一个或更多方面的处理器的框图;

图8a-8b示意示出按照本公开的一个或更多方面的处理器微架构的元件;

图9示出按照本公开的一个或更多方面的示例计算机系统的框图;

图10示出按照本公开的一个或更多方面的示例芯片上系统 (SoC) 的框图;

图11示出按照本公开的一个或更多方面的示例计算机系统的框图;以及

图12示出按照本公开的一个或更多方面的示例芯片上系统 (SoC) 的框图。

具体实施方式

[0004] 本文所述的是用于向用户级应用传递中断的处理系统及相关方法。

[0005] 在通用实现中,用户级应用可经由执行于内核模式中的对应驱动程序与I/O装置进行交互。在说明性示例中,应用可执行系统调用,其引起到内核模式的转变。OS内核可通过将控制传到装置驱动程序来处理系统调用。驱动程序然后可与I/O装置进行交互,以执行应用所请求的操作。当I/O装置完成操作时,它可经由中断 (其可由OS内核和装置驱动程序来处置) 来通知应用。后者然后可通知应用 (例如通过调用先前由应用所注册的回调处理程

序)。

[0006] 因此,应用与I/O装置的交互可引起多个特权级转变,包括从应用到驱动程序的系统调用、在已经发起I/O装置操作之后从内核模式回到应用的转变、因中断而到内核级的转变以及回到应用以调用应用回调处理程序的转变。那些转变可对用户级应用与I/O装置之间的每一个交互引起附加等待时间。这种等待时间在某些状况中可超过I/O装置的操作等待时间。因此,减少或消除特权级转变所引起的软件等待时间可显著改进与I/O装置进行交互的各种用户级应用的操作方面。

[0007] 按照本公开的一个或更多方面,处理系统(例如单核或多核处理器)可包括设计成支持新事件(其在本文中称作“用户级中断”)的某个处理逻辑。与普通中断不同,用户级中断在没有特权级转变的情况下传递给对应用户级应用,并且可以仅当这种应用被执行时才传递。

[0008] 用户级中断事件可调用CPU控制流修改机制,其在本文中称作“用户级中断传递”。在某些实现中,用户级中断传递可基于处理器状态(例如当某个用户级应用被执行时)并且通过某些存储器数据结构(其可由处理器和操作系统协作管理)的状态来触发。这些和其它软件可配置机制及数据结构可使处理器能够直接向用户级应用传递某些I/O中断,而无需将控制流重定向到内核级软件,如下文更详细描述。

[0009] 在某些实现中,存储器数据结构(其照高级编程中断控制器(APIC)来推论,在本文中称作“用户级APIC”)可由处理器和操作系统协作管理。可为处理器所执行的用户级应用的各线程来创建用户级APIC的分离实例。用户级APIC可包括比特图,其包括多个比特标志。每个比特标志可指示与比特图中的比特标志的位置对应的向量所标识的用户级中断的状态。在说明性示例中,处理器可设置比特,以指示与比特图中的比特标志的位置对应的向量所标识的用户级中断当前是挂起的(pending)。

[0010] 操作系统还可维护另一个存储器数据结构,其在本文中称作用户级中断处理程序地址数据结构。可为处理器所执行的用户级应用的各线程来创建用户级中断处理程序地址数据结构的分离实例。在某些实现中,用户级中断处理程序地址数据结构可通过表来表示,该表包括用户级中断处理程序的多个地址,其通过中断向量(其照中断描述符表(IDT)来推论,称作“用户级IDT”)来索引。备选地,单个用户级中断处理程序地址可存储在处理器寄存器中,以标识负责所有中断向量的用户级中断处理程序。在后一种情形中,用户级中断向量可推送到中断处理程序栈。在以下描述和权利要求书中,“用户级中断处理程序地址数据结构”将表示包括用户级中断处理程序的多个地址的上述表或者表示上述单个用户级中断处理程序地址,其可存储在一个或更多处理器寄存器中或者系统存储器中。

[0011] 响应在对应用户级应用被执行的同时接收用户级中断的通知,处理器可通过调用通过用户级中断处理程序地址数据结构所标识的用户级中断处理程序来通知应用。当用户级中断处理程序完成执行时,控制流可返回到软件模块,其在传递用户级中断时正被执行。因此,用户级中断可由用户级应用来传递和处理,而没有引起特权级转变,如下文更详细描述。

[0012] 在某些实现中,处理器可例如基于中断向量号来优先化(prioritize)用户级中断。在说明性示例中,响应接收到用户级中断的通知,处理器可设置与当前由处理器执行的用户级应用关联的用户级APIC中对应用户级中断向量的比特。处理器然后可对于用户级

APIC所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断来调用通过与用户级应用关联的用户级中断处理程序地址数据结构所标识的用户级中断处理程序,如下文更详细描述。

[0013] 作为举例而不是限制,下文更详细描述上述方法和系统的各个方面。

[0014] 在以下描述中,提出许多具体细节、例如具体类型的处理器和系统配置、具体硬件结构、具体架构和微架构细节、具体寄存器配置、具体指令类型、具体系统组件、具体测量/高度、具体处理器流水线级和操作的示例,以便提供对本公开的透彻了解。但是,本领域的技术人员将会清楚地知道,这些具体细节不一定要用来实施本文所公开方法。在其它情况下,没有详细描述众所周知的组件或方法、例如具体和备选处理器架构、用于所述算法的具体逻辑电路/代码、具体固件代码、具体互连操作、具体逻辑配置、具体制造技术和材料、具体编译器实现、通过代码对算法的具体表达、具体断电和光栅技术/逻辑以及计算机系统的其它具体操作细节,以便避免不必要地使本公开难以理解。

[0015] 虽然参照处理器来描述以下示例,但是其它实现可适用于其它类型的集成电路和逻辑装置。本文所述示例的类似技术和理论能够适用于可获益于更高管线吞吐量和改进性能的其它类型的电路或半导体装置。本文所述示例的理论可适用于执行数据操纵的任何处理器或机器。但是,本公开并不局限于执行512比特、256比特、128比特、64比特、32比特或16比特数据操作的处理器或机器,而是能够适用于其中执行数据的操纵或管理的任何处理器和机器。

[0016] 示出本公开和附图的示例不应当在限制的意义中被解释,因为它们只是要提供本文所述实施例的示例,而不是提供本文所述实施例的所有可能实现的详尽列表。虽然以下示例在执行单元和逻辑电路的上下文中描述指令处置和分发,但是本文所述系统和方法的其它实现能够通过机器可读有形媒体上存储的数据或指令(其在被机器执行时使该机器执行与本文所述的至少一个实施例一致的功能)来实现。在某些实现中,与本文所述实施例关联的功能通过机器可执行指令来体现。指令能够用来使采用指令所编程的通用或专用处理器来执行本文所述的方法。本文所述的实现可作为计算机程序产品或软件来提供,其可包括其上存储了指令的机器或计算机可读媒体,指令可用来对计算机(或其它电子装置)进行编程,以执行按照本文所述实施例的一个或更多操作。备选地,本文所述系统和方法的操作可由包含用于执行操作的固定功能逻辑的特定硬件组件或者由编程计算机组件和固定功能硬件组件的任何组合来执行。

[0017] 用来对逻辑进行编程以执行本文所述方法的指令能够存储在系统的存储器中,例如DRAM、高速缓存、闪速存储器或其它存储装置。此外,指令可经由网络或者通过其它计算机可读媒体来分配。因此,机器可读媒体可包括用于存储或传送机器(例如计算机)可读形式的信息的任何机构,包括但不限于软盘、光盘、致密光盘、只读存储器(CD-ROM)以及磁光盘、只读存储器(ROM)、随机访问存储器(RAM)、可擦可编程只读存储器(EPROM)、电可擦可编程只读存储器(EEPROM)、磁或光卡、闪速存储器,或者通过因特网、经由电、光、声或其它形式的传播信号(例如载波、红外信号、数字信号等)来传送信息中使用的有形机器可读存储装置。相应地,计算机可读媒体包括适合于存储或传送采取机器(例如计算机)可读形式的电子指令或信息的任何类型的有形机器可读媒体。

[0018] 本文中的“处理器”将表示能够执行对算术、逻辑或I/O操作进行编码的指令的装

置。在一个说明性示例中,处理器可沿用冯·诺伊曼架构模型,并且可包括算术逻辑单元(ALU)、控制单元和多个寄存器。在另一方面,处理器可包括一个或更多处理核,并且因此可以是单核处理器(其通常能够处理单个指令管线)或者多核处理器(其可同时处理多个指令管线)。在另一方面,处理器可实现为单个集成电路、两个或更多集成电路,或者可以是多芯片模块的组件(例如,其中各个微处理器管芯包含在单个集成电路封装中,并且因此共享单个插座)。

[0019] 现在参照图1,所示的是按照本公开的一个或更多方面的示例处理系统100的框图。如图1所示,处理系统100可包括一个或更多处理核120,其可经由共享互连来耦合到系统存储器140。

[0020] 处理系统100还可包括芯片集130,其支持存储器操作、输入/输出操作、配置、控制、内部或外部接口、连接或通信功能和/或处理系统100的处理核120和/或其它组件的其它类似功能。芯片集130的各个元件可一起被编组于单个芯片、在多个芯片之间散布和/或部分、完全、冗余地或者按照分布式方式集成到一个或更多处理器(包括处理核120)中。

[0021] 在某些实现中,芯片集130可包括中断处理逻辑132,其可按照本公开的一个或更多方面实现用户级中断发布,如下文更详细描述。备选地,中断处理逻辑132可驻留在处理系统100的其它组件中。本文所述系统和方法的各个实现可使用未改变或者修改芯片集元件(其当前用于普通中断处理)来执行用户级中断传递。

[0022] 系统存储器140可包括一个或更多媒体,例如静态或动态随机访问存储器、基于半导体的只读或闪存存储器、磁或光盘存储器或者处理核120可读的任何其它类型的媒体,其上可存储诸如数据和/或程序代码之类的信息。

[0023] 装置150可包括任何类型的I/O装置、外设装置或者另一装置,其可发起中断请求,例如键盘、鼠标、轨迹球、指针装置、监视器、打印机、媒体卡、网络接口、信息存储装置等。装置150可包含在分立组件中,或者可与其它装置集成。在某些实现中,装置150可表示多功能I/O、外设或另一装置中的功能。

[0024] 处理核120、芯片集130、系统存储器140和装置150可直接地或者经过一个或更多并行、依次、管线化、异步、同步、有线、无线和/或另一总线或点对点连接或者通信部件间接地相互耦合。在图1的说明性示例中,芯片集130包括接口131,以便经过任何这种连接或者其它通信部件从装置150接收信号、消息和/或事务(例如中断请求),或者向装置150和/或系统100中的任何其它代理器或组件传送信号、消息和/或事务。类似地,装置150包括接口151,以针对芯片集130和/或处理系统100的任何其它代理器或组件传送和/或接收信号、消息和/或事务。在某些实现中,处理系统100还可包括图1未示出的各种其它组件。

[0025] 在某些实现中,存储器140可用来存储与处理系统100所执行的多个用户级线程关联的多个用户级APIC数据结构142和多个用户级中断处理程序地址数据结构144,如下文更详细描述。备选地,用户级APIC数据结构142和/或用户级中断处理程序地址数据结构144可存储在处理核120的寄存器中。与给定用户线程关联的用户级APIC数据结构142和/或用户级中断处理程序地址数据结构144的基址可存储在由操作系统用来在上下文切换时加载状态的XSAVE区域中。在某些实现中,处理器在上下文切换时可将XSAVE区域中存储的用户级APIC数据结构142和/或用户级中断处理程序地址数据结构144的基址加载到某些处理器寄存器中。

[0026] 普通中断通过其相应唯一中断类型或中断号(常常称作“中断向量”)相互加以区分。用户级中断可与普通中断共享向量空间(使得每一个中断向量标识普通中断或用户级中断),或者它们可被分配专用用户级中断向量空间。分离向量空间允许由各用户级应用对用户级中断的独立优先化,因而促进跨处理核的用户线程的动态迁移。

[0027] 处理器可配置成使用与处理器所执行的多个用户级线程关联的多个用户级APIC数据结构142来跟踪挂起的用户级中断,如图2示意所示。各用户级APIC数据结构142可包括比特图,其包含多个比特标志210。比特图中的每个比特标志的位置可对应于标识与用户级APIC数据结构关联的用户级应用要处理的与用户级中断类型的中断号。响应接收一个或更多用户级中断的通知(例如采取发布的中断描述符中设置的一个或更多比特的形式,如下文更详细描述),处理器可设置与当前由处理器执行的用户级应用关联的用户级APIC数据结构中对应于用户级中断向量的一个或更多比特。用户线程的用户级APIC数据结构的基址可存储在那个线程的XSAVE区域中。

[0028] 在某些实现中,处理器可维护用户级中断的软件控制掩蔽比特。这个比特的某个值可防止处理器传递任何用户级中断,直到比特值被改变。在说明性示例中,处理器可设置掩蔽用户级中断的掩蔽比特,作为下文更详细描述的用户级中断传递过程的组成部分。相反,用户级中断处理程序(或者从用户级中断处理程序返回流程控制的指令)可清除该比特,以允许用户级中断传递。掩蔽比特可驻留在现有处理器寄存器中或者重新定义的处理器的寄存器中。处理器可支持新指令,以允许单独地或者作为该比特所在的寄存器的一部分来读和写该比特。

[0029] 在某些实现中,处理器可例如基于中断向量号来优先化尚未被传递的用户级中断。在说明性示例中,优先级可按照升序与中断向量号关联,使得最低优先级被给予最低向量号,第二优先级被给予第二最低向量号,以及最高优先级被给予最高向量号。处理器可选择传递用户级APIC比特所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断。

[0030] 除了跟踪挂起的用户级中断之外,处理器在某些实现中还可跟踪已经传递但是仍然由软件所服务的中断。在这种情况下,处理器可以仅当其向量号超过当前被服务的中断的最高向量号时才传递用户级中断。为了促进这个用户级中断传递模式,处理器可支持一种机制(例如通过新指令和/或新寄存器所实现),软件通过其可指示用户级中断服务的完成。另外,处理器可支持一种机制,软件通过其可限制待传递用户级中断的类型(例如通过指示应当传递的最低向量)。

[0031] 处理器然后可通过查找与用户级应用关联的用户级中断处理程序地址数据结构中的用户级中断处理程序地址,来标识与选择的最高优先级中断关联的中断处理程序。用户线程的用户级中断处理程序地址数据结构的基址可存储在那个线程的XSAVE区域中。在某些实现中,用户级中断处理程序地址数据结构可通过表144来表示,表144包含用户级中断处理程序220的多个地址,其通过中断向量来索引,如图2示意所示。备选地,用户级中断处理程序地址数据结构可包括负责所有中断向量的单个用户级中断处理程序的地址。

[0032] 处理器然后可调用所标识用户级中断处理程序。在说明性示例中,处理器然后可在栈上存储指令指针(IP)的当前值,并且将所标识用户级中断处理程序地址加载到IP中。当调用所标识用户级中断处理程序时,处理器还可设置用户级中断的掩蔽比特,以掩蔽用

户级中断；包含这个掩蔽比特的处理器寄存器可处于处理器先前存储在栈上的那些之中。用户级中断处理程序可在存储器中或栈上保存CPU状态的一个或更多组件（例如处理器寄存器）的值。用户级中断处理程序可包括多个指令，其设计成处理挂起的中断（例如，将某个数据从与I/O装置关联的存储缓冲器复制到应用所分配的存储缓冲器中或者反之）。用户级中断处理程序可通过恢复所保存CPU状态来完成执行返回（例如RET）指令，其为IP加载存储于栈顶的地址，因而将控制流传到用户级中断被传递之前已经被执行指令的那一个之后的指令。在某些实现中，从用户级中断处理程序的流程控制返回可通过使用现有返回指令来执行。备选地，新指令可添加到指令集，以用于实行从用户级中断处理程序的流程控制返回。这种新指令可清除用户级中断的掩蔽比特以显露用户级中断；备选地，它可从处理器寄存器包含这个掩蔽比特的栈中加载。在各个说明性示例中，可采用调用所标识中断处理程序的其它机制。

[0033] 在某些实现中，来自I/O装置的通知可被显式标记为用户级中断（在这种通知经过平台硬件（例如芯片集130或处理器非核）被传送给CPU时）。备选地，通知可经过具有普通中断向量的普通中断消息来实现。

[0034] 处理器可基于向量号或另外某个性质来确定哪些中断应当被看作是用用户级中断。在说明性示例中，操作系统可维护表，其具有与多个普通中断向量对应的多个条目。中断映射表的各条目可包括比特标志，其指示处理器应当将对应中断转换为用用户级中断。在某些实现中，中断映射表的各条目还可包括用户级中断向量，其应当用于对应中断传递。备选地，处理器可在响应接收普通中断消息而传递用户级中断时保持普通中断的向量。在某些实现中，中断请求可被标识为内核中断或用户级中断，从而允许它经过分离的中断映射表被独立处理。在另一个实现中，中断请求可经过通用中断映射表来处理，其中用来映射中断请求的条目被编程为指定中断请求是作为内核中断还是用用户级中断来处理。

[0035] 在某些实现中，平台硬件（例如芯片集130或处理器非核）可通过在与用户级应用（其与进入的中断向量关联）关联的存储器数据结构中记录与中断有关的信息，来处理来自I/O装置的中断。如图3示意所示，对于可接收用户级中断的各用户线程302，操作系统可创建和维护存储器数据结构310，其在本文中称作“发布的中断描述符”。发布的中断描述符310可包括比特图，其包含多个比特标志314。每个比特标志314可指示与比特图中的比特标志的位置对应的向量所标识的用户级中断的状态。在说明性示例中，平台硬件可设置与挂起的用户级中断向量对应的比特。

[0036] 发布的中断描述符310还可包括通知中断向量316，其标识平台硬件可用来通知处理器关于挂起的用户级中断的普通中断，如下文所述。挂起的中断描述符310还可包括一个或更多控制比特318，其可由处理系统的软件和/或其它代理器使用。

[0037] 操作系统还可创建和维护中断映射表320，其包括包含多个比特标志324的比特图322。每个比特标志324可指示处理器是否应当把来自通过比特图322中的比特标志的位置所标识的中断源的中断转换为用用户级中断。中断映射表320还可包括发布的中断描述符的标识符（例如基址）328的列表326。各发布的中断描述符可与用户线程（其与通过列表326中的发布的中断描述符的位置所标识的普通中断的源关联）关联。

[0038] 在某些实现中，中断映射表320还可包括中断向量329的列表327，其应当用于对应用户级中断传递。备选地，处理器可在响应接收普通中断消息而传递用户级中断时保持普

通中断的向量。

[0039] 响应接收来自I/O装置的中断,平台硬件可查找中断映射表320中的中断源标识符(例如通过装置编号所表示)。如果与中断源关联的比特标志324指示中断应当作为用户级中断来处理,则平台硬件可从中断映射表320中检索与中断源关联的发布的中断描述符310的基址328。平台硬件然后可在发布的中断描述符中设置指示对应用户级中断挂起的比特标志314。比特标志314然后将被处理器清除,以确认对发布的中断的接收。在设置比特标志314时,平台硬件可向处理器传送具有通过发布的中断描述符310的通知中断向量字段316所标识的向量的通知中断。

[0040] 响应接收中断,处理器可将中断向量与当前由处理器所执行的用户线程的通知中断向量进行比较。在说明性示例中,当前用户线程的通知中断向量可通过发布的中断描述符310来标识。备选地,当前用户线程的通知中断向量可由控制寄存器(其是操作系统可编程的)来标识。响应确定进入的中断不是通知中断,处理器可使用通用中断传递机制(例如将控制流传到通过IDT查找所标识的中断处理程序)来传递进入的中断。备选地,响应确定进入的中断向量匹配通知中断向量,处理器可处理发布的用户级中断。

[0041] 发布的用户级中断的处理可包括由处理器读与当前用户线程关联的发布的中断描述符310的比特标志314。处理器可在与当前用户线程关联的用户级APIC数据结构142中的发布的中断描述符310的设置比特标志的位置来设置比特标志210。处理器然后可触发微架构事件,以指示存在挂起的用户级中断。响应检测微架构事件,处理器可选择用于传递的最高优先级用户级中断,通过查找用户级中断处理程序地址数据结构来标识对于选择的用户级中断的中断处理程序,并且调用所标识中断处理程序,如下文参照图5B更详细描述。

[0042] 图4示出按照本公开的一个或更多方面、用于由平台硬件发布用户级中断的示例方法的流程图。方法400可由计算机系统来执行,计算机系统可包括硬件(例如电路、专用逻辑和/或可编程逻辑)、软件(例如计算机系统上可执行以执行硬件模拟的指令)或者其组合。方法400和/或其功能、例程、子例程或操作的每个可由执行该方法的计算机系统的一个或更多物理处理器和/或其它组件来执行。在一个示例中,如图4所示,方法400可由下文所述并且通过图6-12所示的处理系统来执行。

[0043] 参照图4,在框410,实现该方法的平台硬件可接收来自I/O装置的中断。

[0044] 在框420,平台硬件可查找中断映射表中的中断源标识符(例如通过装置编号所表示),其由操作系统来管理,以便向平台硬件指示哪些中断应当被看作是用户级中断,如上文更详细描述。

[0045] 响应在框430确定与中断源关联的比特标志指示中断应当作为用户级中断来处理,处理可在框440继续进行;否则该方法可分支到框470,以执行通用硬件中断传递机制。

[0046] 在框440,平台硬件可从中断映射表中检索与中断源关联的发布的中断描述符的基址,如上文更详细描述。

[0047] 在框450,平台硬件可在发布的中断描述符中设置指示对应用户级中断挂起的比特标志。

[0048] 在框460,平台硬件可向处理器传送具有通过发布的中断描述符的通知中断向量字段所标识的向量的通知中断,如上文更详细描述,以及该方法可结束。

[0049] 图5A-5B示出按照本公开的一个或更多方面、用于向用户级应用传递中断的示例

方法的流程图。具体来说,方法500A示出通知中断的处理,以及方法500B示出用户级中断传递。方法500A-500B可由计算机系统来执行,计算机系统可包括硬件(例如电路、专用逻辑和/或可编程逻辑)、软件(例如计算机系统上可执行以执行硬件模拟的指令)或者其组合。方法500A-500B和/或其功能、例程、子例程或操作的每个可由执行该方法的计算机系统的一个或更多物理处理器和/或其它组件来执行。在一个示例中,如图5A-5B所示,方法500A-500B可由下文所述并且通过图6-12所示的处理系统来执行。

[0050] 参照图5A,在框510,实现该方法的处理器可接收来自平台的中断。

[0051] 响应在框520确定进入的中断向量匹配当前用户线程的通知中断向量,该处理可在框530继续进行;否则该方法可分支到框550,以执行通用中断传递机制。

[0052] 在框530,处理器可在与当前用户线程关联的用户级APIC数据结构中与当前用户线程关联的发布的中断描述符的设置比特标志的位置来设置比特标志,如上文更详细描述。

[0053] 在框540,处理器可基于用户级APIC的状态来触发微架构事件,以指示存在挂起的用户级中断,以便触发用于传递挂起的用户级中断的方法500B。

[0054] 方法500B示出用户级中断传递。该方法可通过方法500A所引起的微架构事件来触发,以指示存在挂起的用户级中断。因此,方法500B可在断言微架构事件之后的某个时间点来调用,这取决于若干条件,包括例如是否掩蔽用户级中断以及当前特权级是否允许用户级中断处理。

[0055] 在框560,处理器可检测到指示存在挂起的用户级中断的微架构事件。

[0056] 在框570,处理器可选择传递用户级APIC比特所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断,如上文更详细描述。

[0057] 在框580,处理器可通过查找与用户级应用关联的用户级中断处理程序地址数据结构中的中断处理程序地址,来标识与选择的最高优先级中断关联的中断处理程序,如上文更详细描述。

[0058] 在框590,处理器可调用所标识用户级中断处理程序,如上文更详细描述,以及该方法可结束。

[0059] 本文所述的系统和方法执行对用户级应用的中断传递。虽然本文参照特定集成电路、例如处理器来描述各种系统和方法,但是其它实现可以可适用于其它类型的集成电路和逻辑装置。本文所述系统和方法的技术和理论可适用于能够获益于更好的能量效率和能量节约的其它类型的电路或半导体器件。例如,所公开实现并不局限于任何特定类型的计算机系统,而是还可用于其它装置、例如手持装置、芯片上系统(SoC)和嵌入式应用中。手持装置的一些示例包括蜂窝电话、因特网协议装置、数码摄像机、个人数字助理(PDA)和手持PC。嵌入式应用可包括微控制器、数字信号处理器(DSP)、网络计算机(NetPC)、机顶盒、网络集线器、广域网(WAN)交换机或者能够执行以下所述功能和操作的其它任何系统。此外,本文所述的系统和方法并不局限于物理计算装置,而是还可涉及软件实现方法。通过本文所述系统和方法所实现的功率节省可与基于操作系统(OS)的机制(例如高级配置和平台接口(ACPI)标准)无关和互补。

[0060] 上文所述的方法和系统可通过各种架构的计算机系统来实现,用于膝上型、台式、手持PC、个人数字助理、工程工作站、服务器、网络装置、网络集线器、交换机、嵌入式处理

器,数字信号处理器(DSP)、图形装置、视频游戏装置、机顶盒、微控制器,蜂窝电话、便携媒体播放机、手持装置和各种其它电子装置的设计和配置也适合实现本文所述方法。一般来说,能够结合本文所公开的处理器和/或其它执行逻辑的大量各种系统或电子装置一般适合于实现本文所述系统和方法。

[0061] 图6示出按照本公开的一个或更多方面的计算机系统的一个示例的高级组件图。按照本发明,例如在本文所述的实施例中,处理系统100可包括采用包括执行用于处理数据的算法的逻辑的执行单元的处理器110。系统100表示基于从Intel Corporation (Santa Clara, California) 可得到的PENTIUM III™、PENTIUM 4™、Xeon™、Itanium、XScale™和/或StrongARM™微处理器的处理系统,但是也可采用其它系统(包括具有其它微处理器的PC、工程工作站、机顶盒等)。在一个实施例中,示例系统100执行从Microsoft Corporation (Redmond, Washington) 可得到的一种版本的WINDOWS™操作系统,但也可采用其它操作系统(例如UNIX和Linux)、嵌入式软件和/或图形用户界面。因此,本发明的实施例并不局限于硬件电路和软件的任何特定组合。

[0062] 在说明性示例中,处理器102包括一个或更多执行单元108,以实现执行至少一个指令的算法。一个实施例可在单处理器台式或服务器系统的上下文中描述,但是备选实施例可包含在微处理器系统中。处理器102耦合到处理器总线110,其在处理器102与系统100中的其它组件之间传送数据信号。系统100的元件(例如图形加速器112、存储器控制器集线器116、存储器120、I/O控制器集线器124、无线收发器126、闪存BIOS 128、网络控制器134、音频控制器136、串行扩展端口138、I/O控制器140等)执行本领域的技术人员众所周知的常规功能。

[0063] 在某些实现中,处理器102包括第1级(L1)内部高速缓冲存储器104。取决于架构,处理器102可具有单个内部高速缓存或者多级内部高速缓存。取决于具体实现和需要,其它实施例包括内部和外部两种高速缓存的组合。寄存器文件(register file) 106在包括整数寄存器、浮点寄存器、向量寄存器、分组寄存器、影子寄存器、校验点寄存器、状态寄存器和指令指针寄存器的各种寄存器中存储不同类型的数据。

[0064] 包括执行整数和浮点操作的逻辑的执行单元108也驻留在处理器102中。在某些实现中,处理器102包括存储微码的微码ROM,微码在被执行时执行某些宏指令的算法或者处置复杂情形。在这里,微码潜在地是可更新的,以处置处理器102的逻辑缺陷/修复。对于一个实施例,执行单元108包括处置紧缩指令集109的逻辑。通过在通用处理器102的指令集中包含紧缩指令集109,连同执行指令的关联电路一起,由许多多媒体应用所使用的操作可使用通用处理器102中的紧缩数据来执行。因此,通过将处理器的数据总线的全宽度用于对紧缩数据执行操作,来加速并且更有效地执行许多多媒体应用。这潜在地消除了需要跨处理器的数据总线传递数据的较小单元以便每次一个数据元素来执行一个或更多操作。

[0065] 执行单元108的备选实现也可用于微控制器、嵌入式处理器、图形装置、DSP和其它类型的逻辑电路。系统100包括存储器120。存储器120包括动态随机访问存储器(DRAM)装置、静态随机访问存储器(SRAM)装置、闪存存储器装置或其它存储器装置。存储器120存储通过将要由处理器102所执行的数据信号所表示的指令和/或数据。

[0066] 系统逻辑芯片116耦合到处理器总线110和存储器120。所述实施例中的系统逻辑芯片116是存储器控制器集线器(MCH)。处理器102能够经由处理器总线110与MCH 116进行

通信。MCH 116为指令和数据存储以及为图形命令、数据和纹理的存储提供到存储器120的高带宽存储器通路118。MCH 116引导处理器102、存储器120和系统100中的其它组件之间的数据信号,并且桥接处理器总线110、存储器120和系统I/O 122之间的数据信号。在一些实施例,系统逻辑芯片116可提供用于耦合到图形控制器112的图形端口。MCH 116经过存储器接口118耦合到存储器120。图形卡112经过加速图形端口 (AGP) 互连114耦合到MCH 116。

[0067] 系统100采用专有集线器接口总线122将MCH 116耦合到I/O控制器集线器 (ICH) 130。ICH 130经由本地I/O总线来提供到一些I/O装置的直接连接。本地I/O总线是用于将外设连接到存储器120、芯片集和处理器102的高速I/O总线。一些示例是音频控制器、固件集线器 (闪存BIOS) 128、无线收发器126、数据存储装置124、包含用户输入和键盘接口的遗留I/O控制器、串行扩展端口 (例如通用串行总线 (USB)) 和网络控制器134。数据存储装置124能够包括硬盘驱动器、软盘驱动器、CD-ROM装置、闪速存储器装置或者其它大容量存储装置。

[0068] 图7是处理器200的微架构的框图,其包括执行按照本公开的一个或更多方面的指令的逻辑电路。在一些实施例中,按照一个实施例的指令能够实现为对于具有字节、字、双字、四字等的大小以及诸如单和双精度整数及浮点数据类型之类的数据类型的数据元素进行操作。在一个实施例中,有序前端201是处理器200的组成部分,其取出待执行指令,并使它们准备以后在处理器管线中使用。前端201可包括若干单元。在一个实施例中,指令预取器226从存储器中取出指令,并且将它们馈送到指令解码器228,其又对它们进行解码或解释。例如,在某些实现中,解码器将所接收指令解码为机器能够执行的一个或更多操作,称作“微指令”或“微操作”(又称作 μop)。在其它实施例中,解码器将指令解析为操作码以及对应数数据和控制字段,其由微架构用来执行按照一个实施例的操作。在一个实施例中,踪迹高速缓存230获得解码 μop ,并且将它们汇编成 μop 队列234中的程序有序序列或踪迹以供执行。当踪迹高速缓存230遇到复杂指令时,微码ROM 232提供完成操作所需的 μop 。

[0069] 一些指令被转换为单个微操作,而其它指令需要若干微操作来完成全面操作。在一个实施例中,如果需要多于四个微操作来完成指令,则解码器228访问微码ROM 232以进行指令。对于一个实施例,能够将指令解码为少量微操作,以供在指令解码器228进行处理。在另一个实施例中,如果需要多个微操作来完成操作,则指令能够存储在微码ROM 232中。踪迹高速缓存230参照入口点可编程逻辑阵列 (PLA) 来确定用于从微码ROM 232中读微码序列的正确微指令指针以完成按照一个实施例的一个或更多指令。在微码ROM 232完成指令的定序微操作之后,机器的前端201继续从踪迹高速缓存230中取出微操作。

[0070] 乱序执行引擎203是其中准备指令以供执行之处。乱序执行逻辑具有多个缓冲器,以便在指令沿管线下行并且被调度以供执行时对其流程进行平整和重新排序,以优化性能。分配器逻辑分配各 μop 所需以便执行的机器缓冲器和资源。寄存器别名化逻辑将逻辑寄存器映射到寄存器文件的条目上。在下列指令调度器之前,分配器还分配两个 μop 队列其中之一各 μop 的条目,一个用于存储器操作并且一个用于非存储器操作:存储器调度器,快速调度器202,慢速/一般浮点调度器204,以及简单浮点调度器206。 μop 调度器202、204、206基于其相关输入寄存器操作数源的准备就绪和 μop 所需以完成操作的执行资源的可用性来确定 μop 准备好执行的时间。一个实施例的快速调度器202能够在主时钟周期的每一半进行调度,而其它调度器在每个主处理器时钟周期能够调度一次。调度器仲裁分发端口,以调度

用于执行的 μop 。

[0071] 物理寄存器文件208、210位于调度器202、204、206与执行块211的执行单元212、214、216、218、220、222、224之间。存在分别用于整数和浮点操作的分离的寄存器文件208、210。一个实施例的各寄存器文件208、210还包括旁路网络,其能够对新的相关 μop 绕过或转发刚完成的结果(其尚未写入寄存器文件中)。整数寄存器文件208和浮点寄存器文件210还能够与另一个传递数据。对于一个实施例,整数寄存器文件208划分为两个分离的寄存器文件,一个寄存器文件用于数据的低阶32比特,并且第二寄存器文件用于数据的高阶32比特。一个实施例的浮点寄存器文件210具有128比特宽的条目,因为浮点指令通常具有从64到128比特宽的操作数。

[0072] 执行块211包含执行单元212、214、216、218、220、222、224,其中实际执行指令。这个部分包括寄存器文件208、210,其存储微指令所需以执行的整数和浮点数据操作数值。一个实施例的处理器200由多个执行单元组成:地址生成单元(AGU) 212,AGU 214,快速ALU 216,快速ALU 218,慢速ALU 220,浮点ALU 222,浮点移动单元224。对于一个实施例,浮点执行块222、224执行浮点、MMX、SIMD和SSE或其它操作。一个实施例的浮点ALU 222包括64比特 $\times$ 64比特浮点除法器,以执行除法、平方根及其余微操作。对于本文所述的系统和方法,涉及浮点值的指令可采用浮点硬件来处置。在一个实施例中,ALU操作转到高速ALU执行单元216、218。一个实施例的快速ALU 216、218能够采用半个时钟周期的有效等待时间来执行快速操作。对于一个实施例,大多数复杂整数操作转到慢速ALU 220,因为慢速ALU 220包括长等待时间类型的操作的整数执行硬件,例如乘法器、移位、标志逻辑和分支处理。存储器加载/存储操作由AGU 212、214执行。对于一个实施例,在对64比特数据操作数执行整数操作的上下文中描述整数ALU 216、218、220。在备选实施例中,ALU 216、218、220能够实现成支持包括16、32、128、256等的多种数据比特。类似地,浮点单元222、224能够实现成支持具有各种宽度的比特的一系列操作数。对于一个实施例,浮点单元222、224能够结合SIMD和多媒体指令对128比特宽的紧缩数据操作数进行操作。

[0073] 在一个实施例中, μop 调度器202、204、206在父加载已经完成执行之前分发相关操作。由于 μop 在处理器200中推测地调度和执行,所以处理器200还包括处置存储器未命中的逻辑。如果数据加载不在数据高速缓存中,则在管线中能够存在即时相关操作,其令调度器具有暂时不正确数据。重放机构跟踪并且重新执行使用不正确数据的指令。应当重放相关操作,而允许不相关操作完成。处理器的一个实施例的调度器和重放机构还设计成捕捉文本串比较操作的指令序列。

[0074] 术语“寄存器”可表示板载处理器存储位置,其用作标识操作数的指令的组成部分。换言之,寄存器可以是从处理器外部(从程序员的角度)可使用的那些寄存器。但是,一实施例的寄存器在含义上不应当局限于特定类型的电路。相反,一实施例的寄存器能够存储和提供数据并且执行本文所述功能。本文所述的寄存器能够通过处理器中的电路使用任何数量的不同技术来实现,例如专用物理寄存器、使用寄存器别名化的动态分配物理寄存器、专用和动态分配物理寄存器的组合等。在一个实施例中,整数寄存器存储32比特整数数据。一个实施例的寄存器文件还包含用于紧缩数据的8个多媒体SIMD寄存器。对于以下论述,寄存器被理解为设计成保存紧缩数据的数据寄存器,例如采用来自Intel Corporation (Santa Clara, California)的MMX™技术所实现的微处理器中的64比特宽MMX寄存器(在一

些情况下又称作‘mm’寄存器)。采取整数和浮点两种形式可用的这些MMX寄存器能够与伴随SIMD和SSE指令的紧缩数据元素配合操作。类似地,与SSE2、SSE3、SSE4或以上(一般称作“SSEx”)的技术相关的128比特宽XMM寄存器也能够用来保存这类紧缩数据操作数。在一个实施例中,在存储紧缩数据和整数数据时,寄存器无需区分两种数据类型。在一个实施例中,整数和浮点包含在相同寄存器文件或不同寄存器文件中。此外,在一个实施例中,浮点和整数数据可存储在不同寄存器或相同寄存器中。

[0075] 图8a-8b示意示出按照本公开的一个或更多方面的处理器微架构的元件。图8a中,处理器管线400包括取级402、长度解码级404、解码级406、分配级408、重命名级410、调度(又称作分发或发出)级412、寄存器读/存储器读级414、执行级416、回写/存储器写级418、异常处置级422和提交级424。

[0076] 图8b中,箭头表示两个或更多单元之间的耦合,以及箭头的方向指示那些单元之间的数据流的方向。图8b示出包括耦合到执行引擎单元450并且均耦合到存储器单元470的前端单元430的处理器核111。

[0077] 核111可以是简化指令集计算(RISC)核、复杂指令集计算(CISC)核、超长指令字(VLIW)核或者混合或备选核类型。作为又一个选项,核111可以是专用核(例如网络或通信核)、压缩引擎、图形核等。

[0078] 前端单元430包括耦合到指令高速缓存单元434的分支预测单元432,指令高速缓存单元434耦合到指令转换后备缓冲器(TLB)436,指令转换后备缓冲器(TLB)436耦合到指令取单元438,指令取单元438耦合到解码单元440。解码单元或解码器可对指令进行解码,并且生成作为输出的一个或更多微操作、微码入口点、微指令、其它指令或其它控制信号,其从原始指令来解码或得出或者以其它方式反映原始指令。解码器可使用各种不同的机制来实现。适当机制的示例包括但不限于查找表、硬件实现、可编程逻辑阵列(PLA)、微代码只读存储器(ROM)等。指令高速缓存单元434还耦合到存储器单元470中的第2级(L2)高速缓存单元476。解码单元440耦合到执行引擎单元450中的重命名/分配器单元452。

[0079] 执行引擎单元450包括重命名/分配器单元452,其耦合到引退单元454和一个或更多调度器单元456的集合。调度器单元456表示任何数量的不同调度器,包括保留站、中心指令窗口等。调度器单元456耦合到物理寄存器文件单元458。物理寄存器文件单元458的每个表示一个或更多物理寄存器文件,其中不同的寄存器文件存储一个或更多不同的数据类型(例如标量整数、标量浮点、紧缩整数、紧缩浮点、向量整数、向量浮点等)、状态(例如,作为要执行的下一个指令的地址的指令指针)等。物理寄存器文件单元458被引退单元454重叠,以示出可实现寄存器别名化和乱序执行的各种方式(例如使用重排序缓冲器和引退寄存器文件,使用将来文件、历史缓冲器和引退寄存器文件;使用寄存器映射和寄存器池等)。一般来说,架构寄存器从处理器外部或者从编程人员的角度来看是可见的。寄存器并不局限于任何已知特定类型的电路。各种不同类型的寄存器是适合的,只要它们能够如本文所述来存储和提供数据。适当寄存器的示例包括但不限于专用物理寄存器、使用寄存器别名化的动态分配物理寄存器、专用和动态分配物理寄存器的组合等。引退单元454和物理寄存器文件单元458耦合到执行群460。执行群460包括一个或更多执行单元162的集合和一个或更多存储器访问单元464的集合。执行单元462可执行各种操作(例如移位、加法、减法、乘法)并且对各种类型的数据(例如,标量浮点、紧缩整数、紧缩浮点、向量整数、向量浮点)来执行。

虽然一些实施例可包括专用于特定功能或功能集合的多个执行单元,但是其它实施例可包括一个执行单元或多个执行单元,其全部执行全部功能。调度器单元456、物理寄存器文件单元458和执行群460示为可能是多个的,因为某些实施例创建某些类型的数据/操作的分离管线(例如标量整数管线、标量浮点/紧缩整数/紧缩浮点/向量整数/向量浮点管线和/或存储器访问管线,其各具有其自己的调度器单元、物理寄存器文件单元和/或执行群一以及在分离的存储器访问管线的情况下,实现这个管线的执行群具有存储器访问单元464的某些实施例)。还应当理解,在使用分离管线的情况下,这些管线的一个或更多可以是乱序发出/执行,而其余的是有序的。

[0080] 存储器访问单元464的集合耦合到存储器单元470,其包括耦合到数据高速缓存单元474(其耦合到第2级(L2)高速缓存单元476)的数据TLB单元472。在一个示范实施例中,存储器访问单元464可包括加载单元、存储地址单元和存储数据单元,其各耦合到存储器单元470中的数据TLB单元472。L2高速缓存单元476耦合到一个或更多其它等级的高速缓存,并且最终耦合到主存储器。

[0081] 作为举例,示范寄存器别名化、乱序发出/执行核架构可按如下所述实现管线400:指令取438执行取和长度解码级402、404;解码单元440执行解码级406;重命名/分配器单元452执行分配级408和重命名级410;调度器单元456执行调度级412;物理寄存器文件单元458和存储器单元470执行寄存器读/存储器读级414;执行群460执行执行级416;存储器单元470和物理寄存器文件单元458执行回写/存储器写级418;各种单元可包含在异常处置级422中;以及引退单元454和物理寄存器文件单元458执行提交级424。

[0082] 核111可支持一个或更多指令集(例如x86指令集(其中具有随较新版本已经添加的一些扩展);MIPS Technologies(Sunnyvale,CA)的MIPS指令集;ARM Holdings(Sunnyvale,CA)的ARM指令集(其中具有附加扩展、例如NEON))。

[0083] 在某些实现中,核可支持多线程执行(执行操作或线程的两个或更多并行集合),并且可按照多种方式这样做,包括时间切片多线程执行、同时多线程执行(其中单个物理核为物理核同时多线程执行的线程的每个提供逻辑核)或者其组合(例如,例如在Intel®超线程执行技术中的时间切片取和解码以及此后的同时多线程执行)。

[0084] 虽然在乱序执行的上下文中描述寄存器别名化,但是应当理解,寄存器别名化可用于有序架构中。虽然处理器的所示实施例还包括分离指令和数据高速缓存单元434/474和共享L2高速缓存单元476,但是备选实施例可具有用于指令和数据的单个内部高速缓存,例如第1级(L1)内部高速缓存或者多级内部高速缓存。在一些实施例中,系统可包括内部高速缓存以及核和/或处理器外部的的外部高速缓存的组合。备选地,高速缓存全部可以是核和/或处理器外部的。

[0085] 在某些实现中,处理器核111可设计为乱序(000)核,以便通过在其操作数变成可用时立即执行指令(而不是按照程序)来改进性能。但是,性能有益效果可被功率消耗的大量增加所抵消。当多个执行线程可用于操作系统以进行调度时,采用多个有序核而不是大000核可改进处理器的能量消耗分布,而没有损害总体性能。因此,要改进处理器的性能和能量消耗可缩放性,后者可设计成根据性能需要和操作系统可用于调度的线程的数量来支持可变数量的核。

[0086] 图9示出按照本公开的一个或更多方面的示例计算机系统700的框图。如图9所示,

多处理器系统700是点对点互连系统,并且包括经由点对点互连750所耦合的第一处理器770和第二处理器780。处理器770和780的每个可以是能够实现基于虚拟机的保护视频通路的处理系统100的某个版本,如上文更详细描述。虽然示为仅具有两个处理器770、780,但是要理解,本公开的范围并不局限于此。在其它实施例中,一个或更多附加处理器可存在于示例计算机系统中。

[0087] 示出处理器770和780,其分别包括集成存储器控制器单元772和782。处理器770还包括作为其总线控制器单元的部分的点对点(P-P)接口776和778;类似地,第二处理器780包括P-P接口786和788。处理器770、780可使用点对点(P-P)接口电路778、788经由P-P接口750来交换信息。如图9所示,IMC 772和782将处理器耦合到相应存储器、即存储器732和存储器734,其可以是本地附连到相应处理器的主存储器的部分。

[0088] 处理器770、780均可使用点对点接口电路776、794、786、798经由各个P-P接口752、754与芯片集790交换信息。芯片集790还可经由高性能图形接口739与高性能图形电路738交换信息。

[0089] 共享高速缓存(未示出)可包含处理器中或者两个处理器外部,但是仍然经由P-P互连与处理器连接,使得如果使处理器进入低功耗模式,则任一个或两个处理器的本地高速缓存信息可存储在共享高速缓存中。

[0090] 芯片集790可经由接口796耦合到第一总线716。在一个实施例中,第一总线716可以是外设组件互连(PCI)总线或者例如PCI高速总线等总线或另一种第三代I/O互连总线,但是本公开的范围并不局限于此。

[0091] 如图9所示,各种I/O装置714可连同总线桥718一起耦合到第一总线716,总线桥718将第一总线716耦合到第二总线720。在一个实施例中,第二总线720可以是低引脚数(LPC)总线。在一个实施例中,各种装置可耦合到第二总线720,包括例如键盘和/或鼠标722、通信装置727以及可包括指令/代码和数据730的存储单元728、例如磁盘驱动器或其它大容量存储装置。此外,音频I/O 724可耦合到第二总线720。

[0092] 图10示出按照本公开的一个或更多方面的示例芯片上系统(SoC)900的框图。应用处理器910由能够实现基于虚拟机的保护视频通路的处理系统100的某个版本来提供,如上文更详细描述。如图10示意所示,互连单元902可耦合到:应用处理器910,其包括一个或更多核902A-N的集合和共享高速缓存单元906;系统代理器单元910;总线控制器单元916;集成存储器控制器单元914;一个或更多媒体处理器920的集合,其可包括集成图形逻辑908、用于提供静止和/或视频摄像机功能性的图像处理器924、用于提供硬件音频加速的音频处理器926和用于提供视频编码/解码加速的视频处理器928;静态随机访问存储器(SRAM)单元930;直接存储器访问(DMA)单元932;以及显示器单元940,用于耦合到一个或更多外部显示器。

[0093] 图11示出按照本公开的一个或更多方面的示例计算机系统的框图。处理器1610可由能够实现基于虚拟机的保护视频通路的处理系统100的某个版本来提供,如上文更详细描述。

[0094] 图11示意所示的系统1600可包括组件的任何组合,其中组件实现为在计算机系统中适配的IC、其部分、分立电子装置或其它模块、逻辑、硬件、软件、固件或者其组合,或者实现为以其它方式结合在计算机系统的机箱(chassis)中的组件。图11的框图意在示出计算

机系统的许多组件的高级视图。但是要理解,所示组件的一部分可省略,附加组件可存在,以及所示组件的不同布置可在其它实现中发生。

[0095] 处理器1610可由微处理器、多核处理器、多线程处理器、超低压处理器、嵌入式处理器或者其它已知处理元件来提供。在所示实现中,处理器1610充当主处理单元和中央集线器,以用于与系统1600的许多各种组件的通信。作为一个示例,处理器1600可实现为芯片上系统 (SoC)。作为具体说明性示例,处理器1610包括基于Intel® 架构核™ 的处理器(例如 i3、i5、i7)或者从Intel Corporation (Santa Clara, CA) 可得到的另一个这种处理器。

[0096] 处理器1610可与系统存储器1615进行通信。在各个实现中,各个存储器装置可具有不同封装类型,例如单管芯封装 (SDP)、双管芯封装 (DDP) 或四管芯封装 (IP)。在一些实现中,这些装置可直接焊接到主板上,以提供更低剖面解决方案,而在其它实现中,装置可配置为一个或更多存储器模块,其又通过给定连接器来耦合到主板。其它存储器实现是可能的,例如其它类型的存储器模块,例如不同种类的双列直插存储器模块 (DIMM),包括但不限于microDIMM、MiniDIMM。在一个说明性示例中,存储器可确定大小在2GB与16GB之间,并且可配置为DDR3LM封装或LPDDR2或LPDDR3存储器,其经由球栅阵列 (BGA) 来焊接到主板上。

[0097] 要提供诸如数据、应用、一个或更多操作系统等的信息的永久存储,大容量存储装置1620也可耦合到处理器1610。在某些实现中,要实现更薄和更轻的系统设计以及改进系统响应性,大容量存储装置1620可经由SSD来实现。在其它实现中,大容量存储装置主要可由硬盘驱动器 (HDD) 来提供,其中具有少量SSD存储装置来充当SSD高速缓存,以便实现断电事件期间的上下文状态和其它这种信息的非易失性存储,使得快速加电在系统活动的重新开始能够发生。

[0098] 又如图11所示,闪存装置1622可例如经由串行外设接口 (SPI) 来耦合到处理器1610。闪存装置1622可提供系统软件(包括基本输入/输出软件 (BIOS) 以及系统的其它固件)的非易失性存储。

[0099] 在各个实现中,系统的大容量存储装置可由SSD单独来提供或者作为具有SSD高速缓存的磁盘、光或其它驱动器来提供。在一些实现中,大容量存储装置能够由SSD来提供或者作为HDD连同恢复 (RST) 高速缓存模块一起来提供。SSD高速缓存可配置为单级高速缓存 (SLC) 或者多级高速缓存 (MLC) 选项,以提供适当等级的响应性。

[0100] 各种输入/输出 (IO) 装置可存在于系统1600中,包括例如显示器1624,其可由配置在机箱的盖子部分中的高清晰度LCD或LED面板来提供。这个显示器面板还可提供触摸屏1625,其在显示器面板之上外部被适配,使得经由用户与这个触摸屏的交互,用户输入能够提供给系统,以实现例如针对信息的显示、信息的访问等的预期操作。在某些实现中,显示器1624可经由显示器互连(其能够实现为高性能图形互连)来耦合到处理器1610。触摸屏1625可经由另一个互连(其在一实施例中能够是I2C互连)来耦合到处理器1610。除了触摸屏1625之外,通过触摸进行的用户输入还能够经由触摸板1630发生,触摸板1630可配置在机箱中,并且还可耦合到与触摸屏1625相同的I2C互连。

[0101] 各种传感器可存在于系统中,并且可按照不同方式耦合到处理器1610。某些惯性和环境传感器可经过传感器集线器1640、例如经由I2C互连来耦合到处理器1610。这些传感器可包括加速计1641、环境光传感器 (ALS) 1642、罗盘1643和陀螺仪1644。其它环境传感器可包括一个或更多热传感器1646,其在一实施例中经由系统管理总线 (SMBus) 总线来耦

合到处理器1610。在某些实现中,一个或更多红外或其它热感测元件或者用于感测用户的存在或移动的任何其它元件可存在。

[0102] 各种外设装置可经由低引脚数(LPC)互连来耦合到处理器1610。在某些实现中,各种组件能够经由嵌入式控制器1635来耦合。这类组件能够包括键盘1636(例如经由PS2接口所耦合)、风扇1637和热传感器1639。在一些实施例中,触摸板1630还可经由PS2接口来耦合到EC 1635。另外,安全处理器、例如按照可信计算组织(TCG)TPM规范版本1.2(2003年10月2日)的可信平台模块(TPM)1638也可经由这个LPC互连来耦合到处理器1610。

[0103] 在某些实现中,外设端口可包括:高清晰度媒体接口(HDMI)连接器(其能够具有不同形状因数,例如全尺寸、袖珍或微型);一个或更多USB端口,例如按照通用串行总线修订版本3.0规范(2008年11月)的全尺寸外部端口,其中至少一个在系统处于连接待机状态时被加电以用于对USB装置(例如智能电话)进行充电,并且被插入AC壁式电源。另外,能够提供一个或更多Thunderbolt™端口。其它端口可包括外部可访问读卡器、例如用于WWAN的全尺寸SD-XC读卡器和/或SIM读卡器(例如8引脚读卡器)。对于音频,具有立体声和话筒能力(例如组合功能性)的3.5mm插孔能够存在,其中具有对插孔检测的支持(例如耳机仅支持使用盖子中的话筒或者具有缆线中的话筒的耳机)。在一些实施例中,这个插孔在立体声耳机与立体声话筒输入之间能够是可重新分派任务的。还能够提供电源插孔,以用于耦合到AC砖(brick)。

[0104] 系统1600能够按照多种方式(包括无线)与外部装置进行通信。在图16所示的实施例中,各种无线模块(其各能够对应于配置用于特定无线通信协议的无线电单元)存在。短程、例如近场的无线通信的一种方式可以是经由近场通信(NFC)单元1645,其在一个实施例中可经由SMBus与处理器1610进行通信。

[0105] 附加无线单元能够包括其它短程无线引擎,包括WLAN单元1650和蓝牙单元1652。使用WLAN单元1650,能够实现按照给定电气和电子工程师协会(IEEE)802.11标准的Wi-Fi™通信,而经由蓝牙单元1652,经由蓝牙协议的短程通信能够发生。这些单元可经由例如USB链路或通用异步接收器发射器(UART)链路 with 处理器1610进行通信。或者这些单元可经由按照外设组件互连高速™(PCIe™)协议、例如按照PCI高速™规范基本规范版本3.0(2007年1月17日发表)或者另一个这种协议、例如串行数据输入/输出(SDIO)标准的互连来耦合到处理器1610。当然,这些外设装置(其可配置在一个或更多插入卡上)之间的实际物理连接能够通过适合主板的NGFF连接器进行。

[0106] 另外,例如按照蜂窝或另一无线广域协议的无线广域通信能够经由WWAN单元1656(其又可耦合到订户身份模块(SIM)1657)发生。另外,要实现位置信息的接收和使用,GPS模块1655也可存在。

[0107] 要提供音频输入和输出,音频处理器能够经由数字信号处理器(DSP)1660(其可经由高清晰度音频(HAD)链路来耦合到处理器1610)来实现。类似地,DSP 1660可与集成编码器/解码器(CODEC)和放大器1662(其又可耦合到可在机箱中实现的输出扬声器1663)进行通信。类似地,放大器和CODEC 1662能够经耦合以接收来自话筒1665的音频输入。

[0108] 图12示出按照本公开的一个或更多方面的示例芯片上系统(SoC)的框图。作为具体说明性示例,SOC 1700可包含在用户设备(UE)中。在一个实施例中,UE表示将要由最终用户用来进行通信的任何装置,例如手持电话、智能电话、平板、超薄笔记本、具有宽带适配器

的笔记本或者任何其它类似通信装置。UE通常连接到基站或节点,其潜在地实际上对应于GSM网络中的移动站(MS)。

[0109] 如图12示意所示,SOC 1700可包括两个核。核1706和1707可耦合到高速缓存控制1708,其与总线接口单元1709和L2高速缓存1710关联,以便与系统1700的其它部分进行通信。互连1710可包括芯片上互连,例如IOSF、AMBA或另一互连。

[0110] 接口1710可提供到其它组件(例如:订户身份模块(SIM) 1730,以便与SIM卡进行接口;引导ROM 1735,以便保存引导代码,供核1706和1707执行,以初始化和引导SOC 1700;SDRAM控制器1740,以便与外部存储器(例如DRAM 1760)进行接口;闪存控制器1745,以便与非易失性存储器(例如闪存1765)进行接口;外设控制1550(例如串行外设接口),以便与外设进行接口;视频编解码器1720和视频接口1725,以便显示和接收输入(例如触摸使能的输入);GPU 1715,以便执行图形相关计算;等等)的通信信道。另外,系统可包括用于通信的外设,例如蓝牙模块1770、3G调制解调器1775、GPS 1785和WiFi 1785。

[0111] 其它计算机系统设计和配置也可适合实现本文所述系统和方法。下列示例示出按照本公开的一个或更多方面的各个实现。

[0112] 示例1是一种处理系统,包括:存储器,配置成存储与处理系统所执行的多个用户级应用对应的多个用户级中断处理程序地址数据结构和多个用户级高级可编程中断控制器(APIC)数据结构;以及处理核,响应接收到用户级中断的通知而配置成:设置与当前由处理核执行的用户级应用关联的用户级APIC数据结构中具有该用户级中断的标识符所定义的位置的挂起的中断比特标志,并且为该用户级APIC数据结构所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断来调用与该用户级应用关联的用户级中断处理程序地址数据结构所标识的用户级中断处理程序。

[0113] 在示例2中,示例1的处理系统的处理核还响应设置挂起的中断比特标志而配置成触发指示存在挂起的用户级中断的微架构事件。

[0114] 在示例3中,示例2的处理系统的处理核响应检测到指示存在挂起的用户级中断的微架构事件而配置成调用该用户中断处理程序。

[0115] 示例4是示例1-3中的任一个的处理系统,其中所述多个用户级APIC数据结构的各用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中比特图中的每个比特的位置对应于中断标识符。

[0116] 示例5是示例1-3中的任一个的处理系统,其中,各用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

[0117] 示例6是示例1-3中的任一个的处理系统,其中各用户级中断处理程序地址数据结构包括与该用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

[0118] 示例6是示例1-3中的任一个的处理系统,还包括处理逻辑,其配置成:响应将进入的中断标识为用户级中断而标识与当前由处理核执行的用户级应用关联的发布的中断描述符的地址;设置与该发布的中断描述符关联的比特图中与所述中断的标识符相对应的比特;以及传送具有该发布的中断描述符所标识的中断号的通知中断。

[0119] 在示例8中,示例7的处理系统的处理核还配置成将进入的中断的标识符与为通知中断所保留的标识符进行比较。

[0120] 示例9是示例1-3中的任一个的处理系统,其中调用用户级中断处理程序还包括:将指令指针的当前值存储在栈上;以及将用户级中断处理程序的地址加载到指令指针中。

[0121] 示例9是示例1-3中的任一个的处理系统,其中处理系统通过芯片上系统 (SoC) 来表示。

[0122] 示例11是一种方法,包括:接收用户级中断的通知;设置与其当前由处理核执行的用户级应用关联的用户级APIC数据结构中与该用户级中断的标识符相对应的挂起的中断比特;使用该用户级APIC数据结构来标识一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断;以及使用与该用户级应用关联的用户级中断处理程序地址数据结构来标识对于所标识的用户级中断的中断处理程序。

[0123] 示例12是示例11的方法,还包括:响应设置该挂起的中断比特标志而触发指示存在挂起的用户级中断的微架构事件。

[0124] 示例13是示例12的方法,还包括:响应检测到指示存在挂起的用户级中断的微架构事件而调用所标识的用户中断处理程序。

[0125] 示例14是示例13的方法,其中调用用户级中断处理程序还包括:将指令指针的当前值存储在栈上;以及将用户级中断处理程序的地址加载到指令指针中。

[0126] 示例15是示例11-14中的任一个的方法,其中所述多个用户级APIC数据结构的各用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中比特图中的每个比特的位置对应于中断标识符。

[0127] 示例16是示例11-14中的任一个的方法,其中,各用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

[0128] 示例17是示例11-14中的任一个的方法,其中各用户级中断处理程序地址数据结构包括与该用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

[0129] 示例18是示例11-14中的任一个的方法,还包括:响应将进入的中断标识为用户级中断而标识与当前由处理核执行的用户级应用关联的发布的中断描述符的地址;设置与该发布的中断描述符关联的比特图中与该中断的标识符相对应的比特;以及传送具有该发布的中断描述符所标识的中断号的通知中断。

[0130] 示例16是示例18的方法,还包括:将进入的中断的标识符与为通知中断所保留的标识符进行比较。

[0131] 示例20是一种设备,包括:存储器;以及处理系统,耦合到存储器,所述处理系统用于执行示例11-19中的任一个的方法。

[0132] 示例21是一种包括可执行指令的计算机可读非暂态存储媒体,所述可执行指令在由处理系统执行时使该处理系统执行操作,包括:接收用户级中断的通知;设置与当前由处理核执行的用户级应用关联的用户级APIC数据结构中与所述用户级中断的标识符相对应的挂起的中断比特;使用该用户级APIC数据结构来标识一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断;以及使用与该用户级应用关联的用户级中断处理程序地址数据结构来标识所标识的用户级中断的中断处理程序。

[0133] 示例22是示例21的计算机可读非暂态存储媒体,还包括使处理系统响应设置挂起的中断比特标志而触发指示存在挂起的用户级中断的微架构事件的可执行指令。

[0134] 示例23是示例22的计算机可读非暂态存储媒体,还包括使处理系统响应检测到指示存在挂起的用户级中断的微架构事件而调用所标识的用户中断处理程序的可执行指令。

[0135] 示例24是示例23的计算机可读非暂态存储媒体,其中调用用户级中断处理程序还包括:将指令指针的当前值存储在栈上;以及将用户级中断处理程序的地址加载到指令指针中。

[0136] 示例25是示例21-24中的任一个的计算机可读非暂态存储媒体,其中所述多个用户级APIC数据结构的各用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中比特图中的每个比特的位置对应于中断标识符。

[0137] 示例26是示例21-24中的任一个的计算机可读非暂态存储媒体,其中,各用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

[0138] 示例27是示例21-24中的任一个的计算机可读非暂态存储媒体,其中各用户级中断处理程序地址数据结构包括与该用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

[0139] 示例28是示例21-24中的任一个的计算机可读非暂态存储媒体,还包括使处理系统响应将进入的中断标识为用户级中断而执行以下操作的可执行指令:标识与当前由处理核执行的用户级应用关联的发布的中断描述符的地址;设置与该发布的中断描述符关联的比特图中与该中断的标识符相对应的比特;以及传送具有该发布的中断描述符所标识的中断号的通知中断。

[0140] 示例29是示例28的计算机可读非暂态存储媒体,还包括使处理系统将进入的中断的标识符与为通知中断所保留的标识符进行比较的可执行指令。

[0141] 详细描述的一些内容部分根据对计算机存储器中的数据比特的操作的算法和符号表示来提供。这些算法描述和表示是由数据处理领域的技术人员用来向本领域的其它技术人员最有效地传达其工作主旨的方式。算法在此以及一般被认为是产生预期结果的自我一致的操作序列。操作是要求物理量的物理操纵的那些操作。这些数量通常但不一定采取电或磁信号的形式,它们能够被存储、传送、组合、比较以及以其它方式处理。主要为了一般使用的原因,将这些信号称作比特、值、元素、符号、字符、项、编号等,已经证明有时非常便利。

[0142] 但是应当记住,所有这些及类似项均与适当的物理量关联,并且只是应用于这些量的便捷标签。除非另加具体说明,否则如从以上论述显而易见的,领会到,在本描述中,利用诸如“加密”、“解密”、“存储”、“提供”、“得出”、“得到”、“接收”、“认证”、“删除”、“执行”、“请求”、“通信”等的术语的论述通篇指的是计算系统或类似电子计算装置的动作和过程,其中计算系统或类似电子计算装置操纵表示为计算系统的寄存器和存储器中的物理(例如电子)量的数据并将其变换为计算系统存储器或寄存器或者其它这种信息存储、传输或显示装置中类似地表示为物理量的其它数据。

[0143] 词语“示例”或“示范”在本文中用来表示“用作示例、实例或说明”。本文中描述为“示例”或“示范”的任何方面或设计不一定被解释为相对其它方面或设计是优选或有利的。相反,词语“示例”或“示范”的使用意在以具体方式呈现概念。如本申请所使用的术语“或者”意在表示包含“或”而不是互斥“或”。也就是说,除非另加说明或者从上下文清楚地知道,否则“X包括A或B”意在表示自然包含置换的任一个。也就是说,如果X包括A、X包括B或者

X包括A和B,则在上述情况的任一种下均满足“X包括A或B”。另外,如本申请和所附权利要求书中所使用的冠词“一”和“一个”应当一般地被解释为表示“一个或更多”,除非另加说明或者从上下文清楚地知道针对单数形式。此外,术语“一实施例”或“一个实施例”或者“一实现”或“一个实现”的通篇使用不是意在表示相同实施例或实现,除非另加说明。另外,如本文所使用的术语“第一”、“第二”、“第三”、“第四”等也表示区分不同元件的标记,而可能不一定具有按照其数字标号的序数含意。

[0144] 本文所述的实施例还可涉及用于执行本文的操作的设备。这个设备可专门构造用于所需目的,或者可包括通过计算机中存储的计算机程序有选择地激活或重新配置的通用计算机。这种计算机程序可存储在非暂态计算机可读存储媒体中,例如但不限于:任何类型的磁盘,包括软盘、光盘、CD-ROM和磁光盘;只读存储器(ROM)、随机访问存储器(RAM), EPROM、EEPROM、NVRAM、磁或光卡、闪速存储器;或者适合于存储电子指令的任何类型的媒体。术语“计算机可读存储媒体”应当被认为包括单个媒体或者多个媒体(例如集中或分布式数据库和/或关联高速缓存和服务器的),其存储一个或更多指令集。术语“计算机可读媒体”也将被认为包括能够对于供机器执行的指令集进行存储、编码或携带并且使机器执行提出的实施例的方法中的任何一个或更多方法的任何媒体。术语“计算机可读存储媒体”相应地将被认为包括但不限于固态存储器、光媒体、磁媒体、能够存储供机器执行的指令集并且使机器执行提出的实施例的方法的任何或更多方法的任何媒体。

[0145] 本文所示的算法和显示并不是固有地与任何特定计算机或其它设备相关。各种通用系统可与按照本文理论的程序配合使用,或者构建更专门的设备来执行所需方法操作可证明是便利的。多种这类系统的所需结构将通过以下描述来呈现。另外,没有参照任何具体编程语言来描述提出的实施例。将领会到,各种编程语言可用来实现本文所述的实施例的理论。

[0146] 以上描述提出例如具体系统、组件、方法的示例等许多具体细节,以便提供对若干实施例的透彻了解。然而,本领域的技术人员清楚地知道,即使没有这些具体细节也可实施至少部分实施例。在其它情况下,没有详细描述众所周知的材料或方法,或者以简单框图来提供众所周知的组件或方法,以便避免不必要地使提出的实施例难以理解。因此,上述具体细节只是示范性的。具体实现可不同于这些示范细节,但仍然预期落入提出的实施例的范围之内。

[0147] 要理解,预计以上描述是说明性而不是限制性的。通过阅读和理解以上描述,许多其它实施例将对于本领域的技术人员是显而易见的。因此,提出的实施例的范围应当参照所附权利要求连同这类权利要求被授权的完整等效范围被确定。

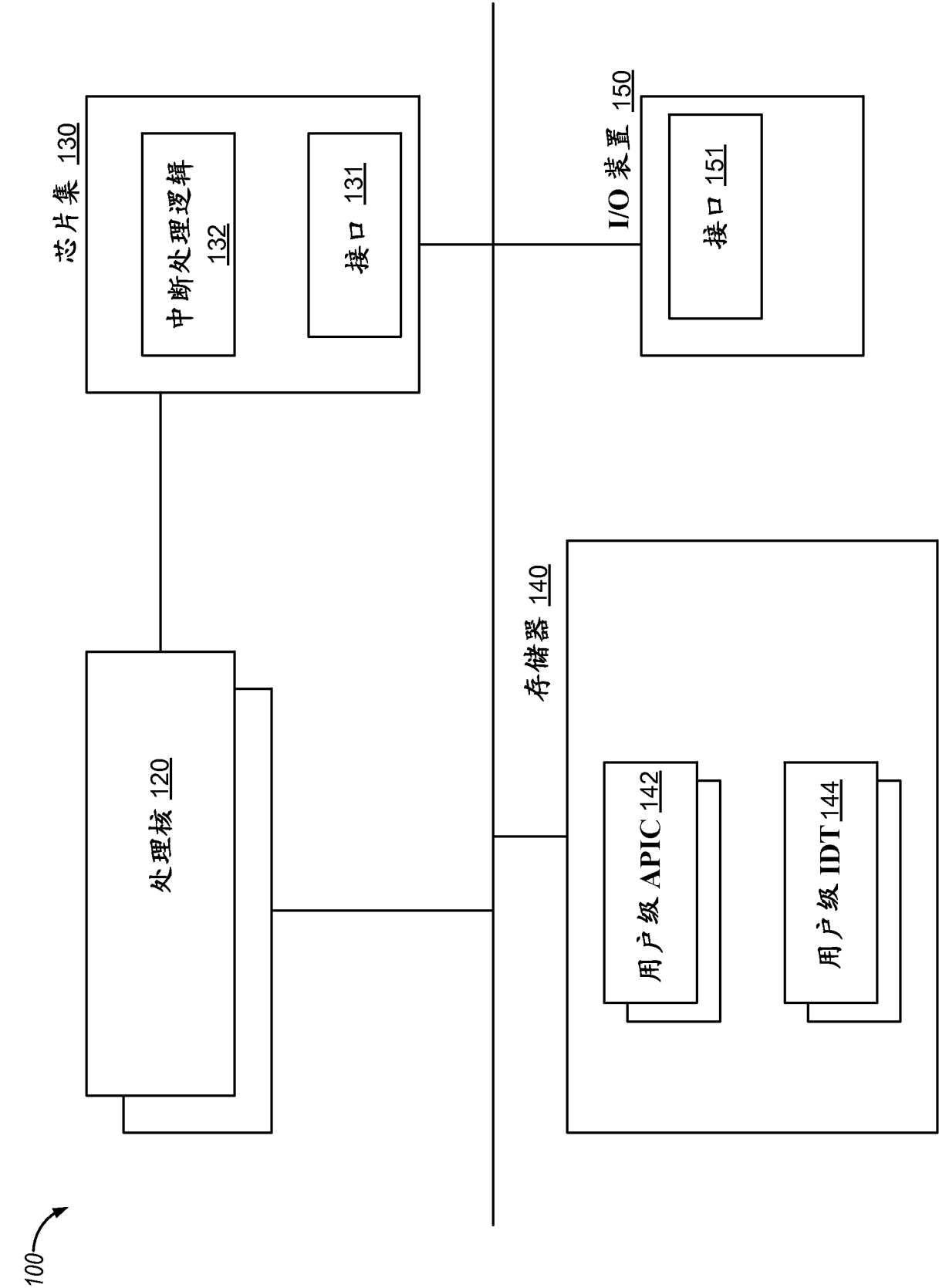
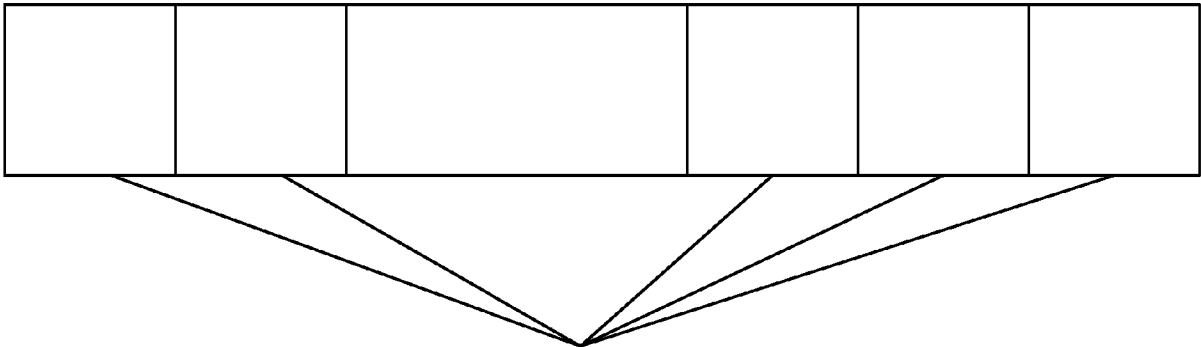


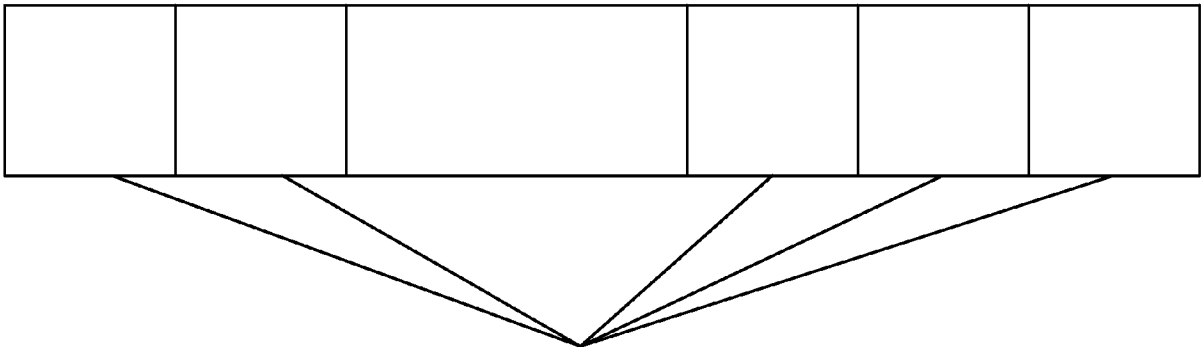
图 1

用户级 APIC 142



挂起的用户级中断比特标志 210

用户级 IDT 144



用户级中断处理程序 220

图 2

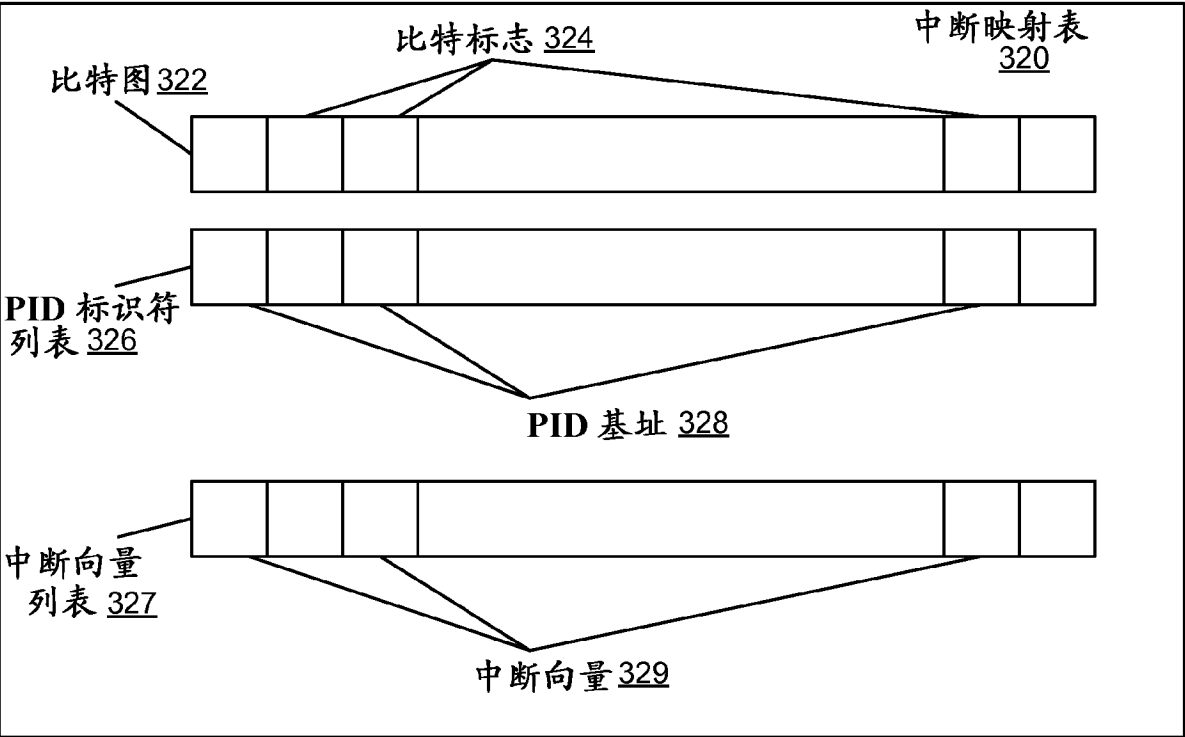
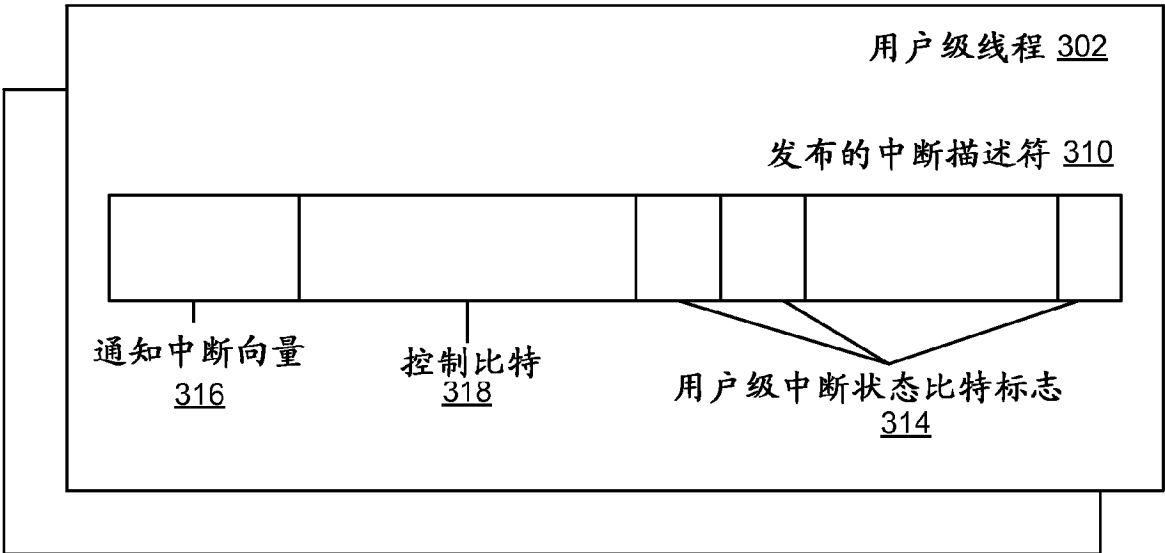


图 3

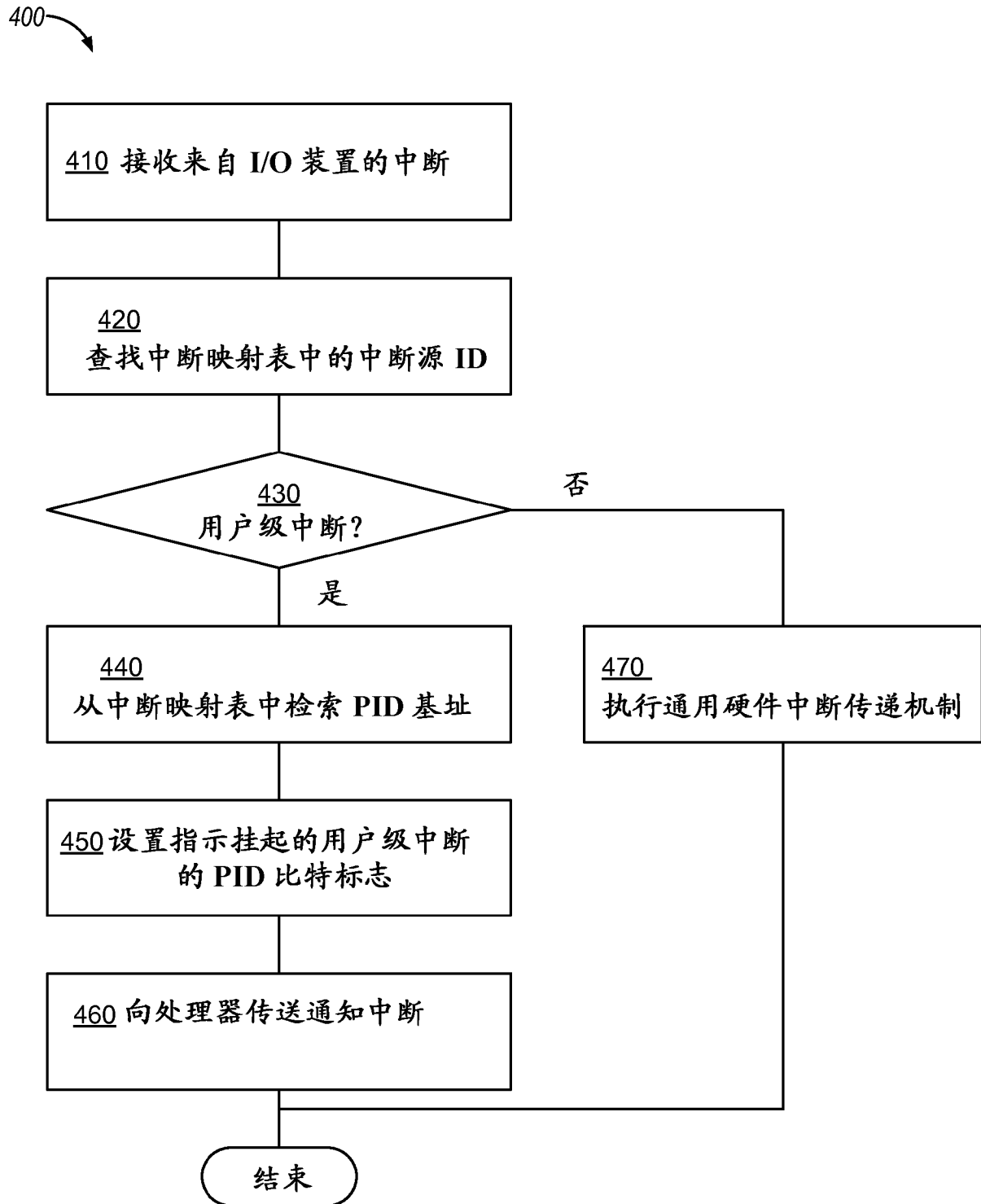


图 4

500A

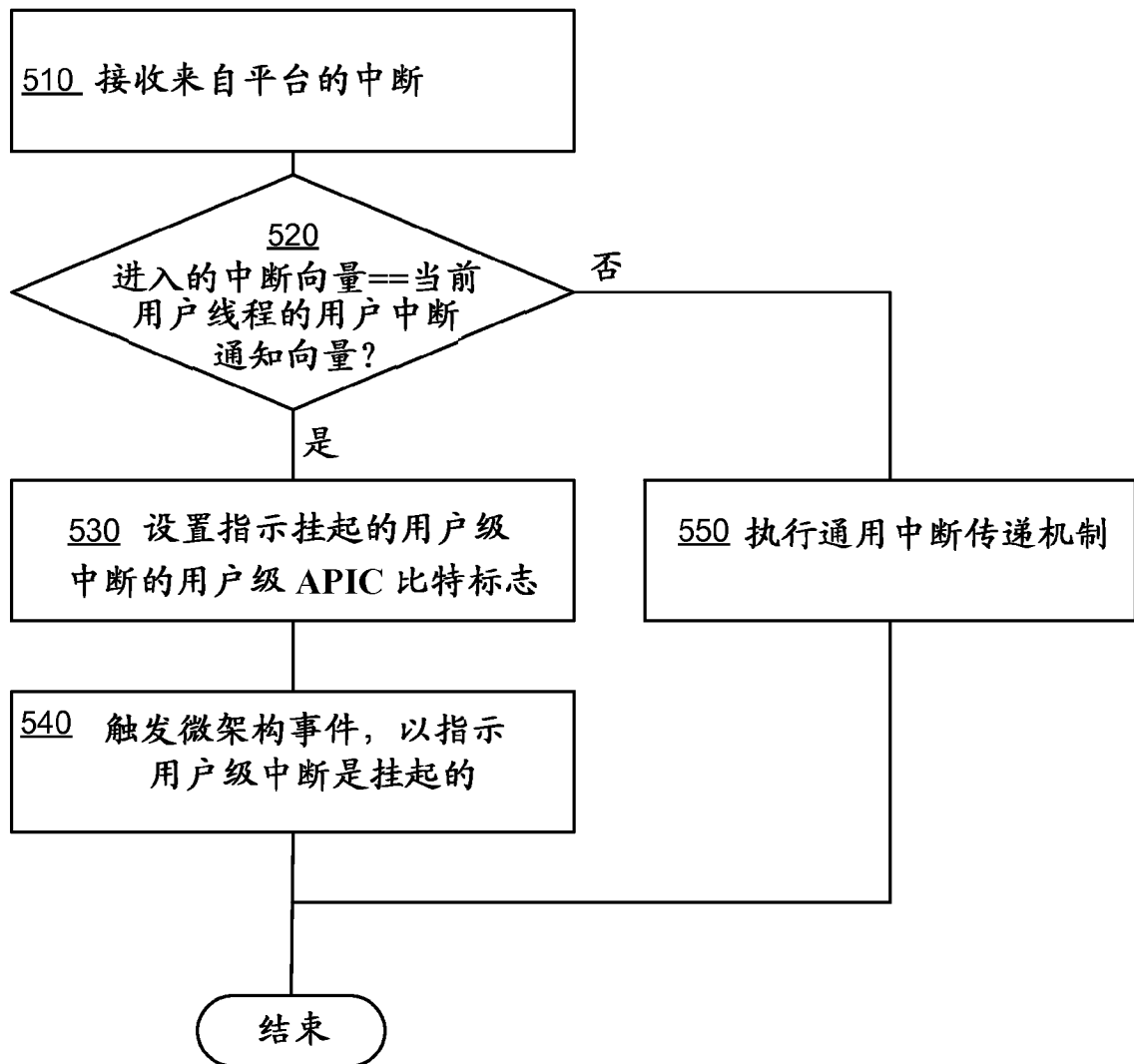


图 5A

500B

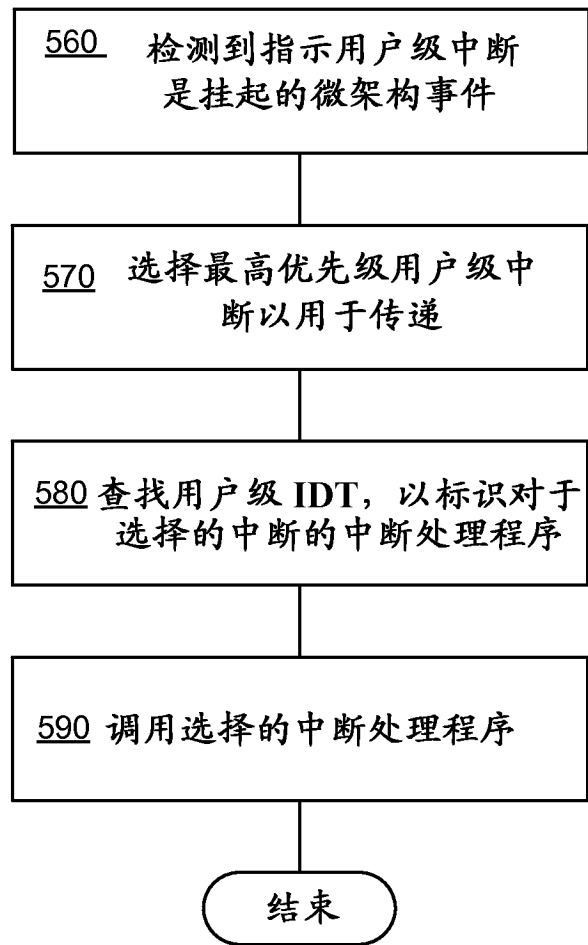


图 5B

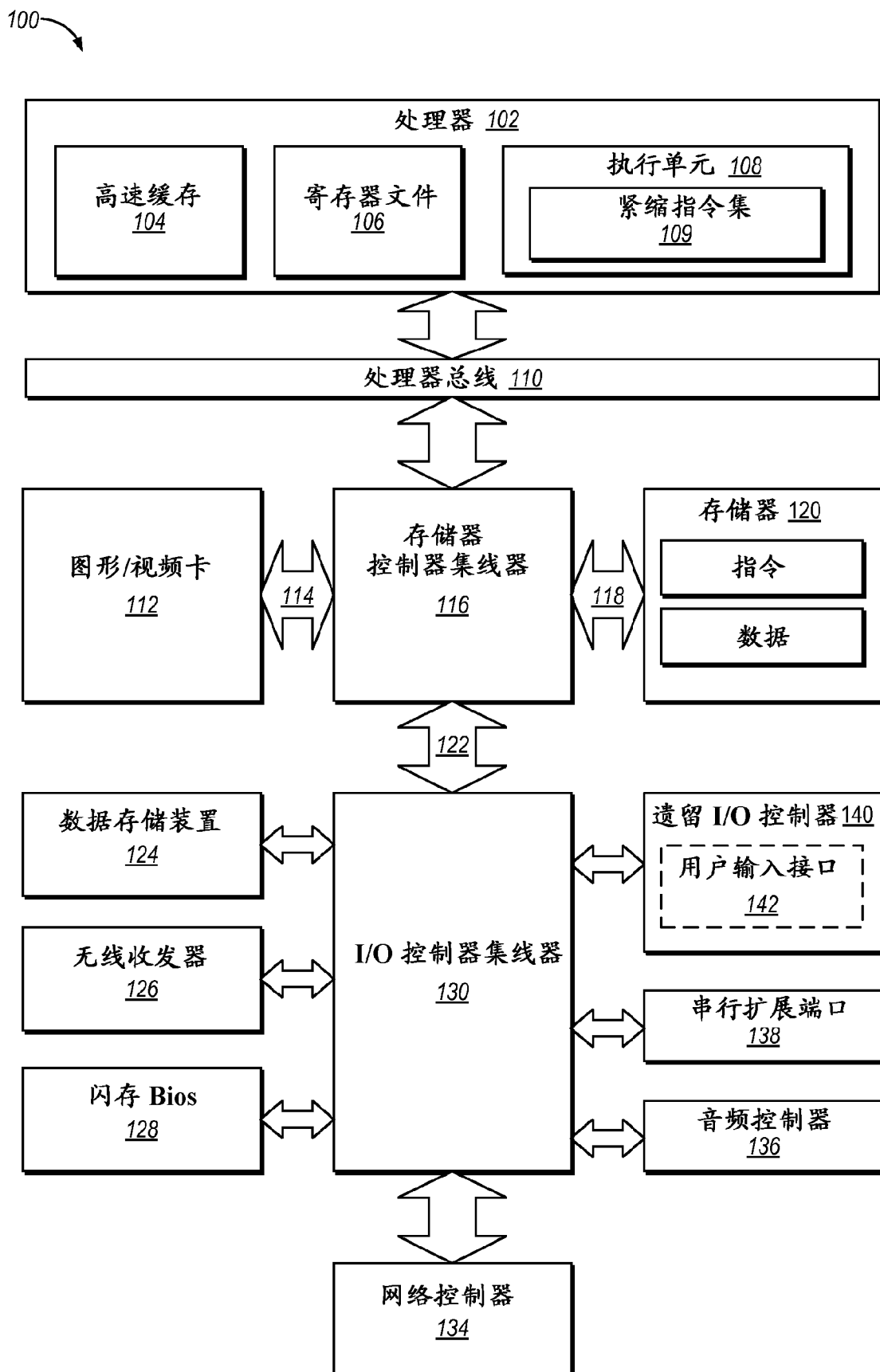


图 6

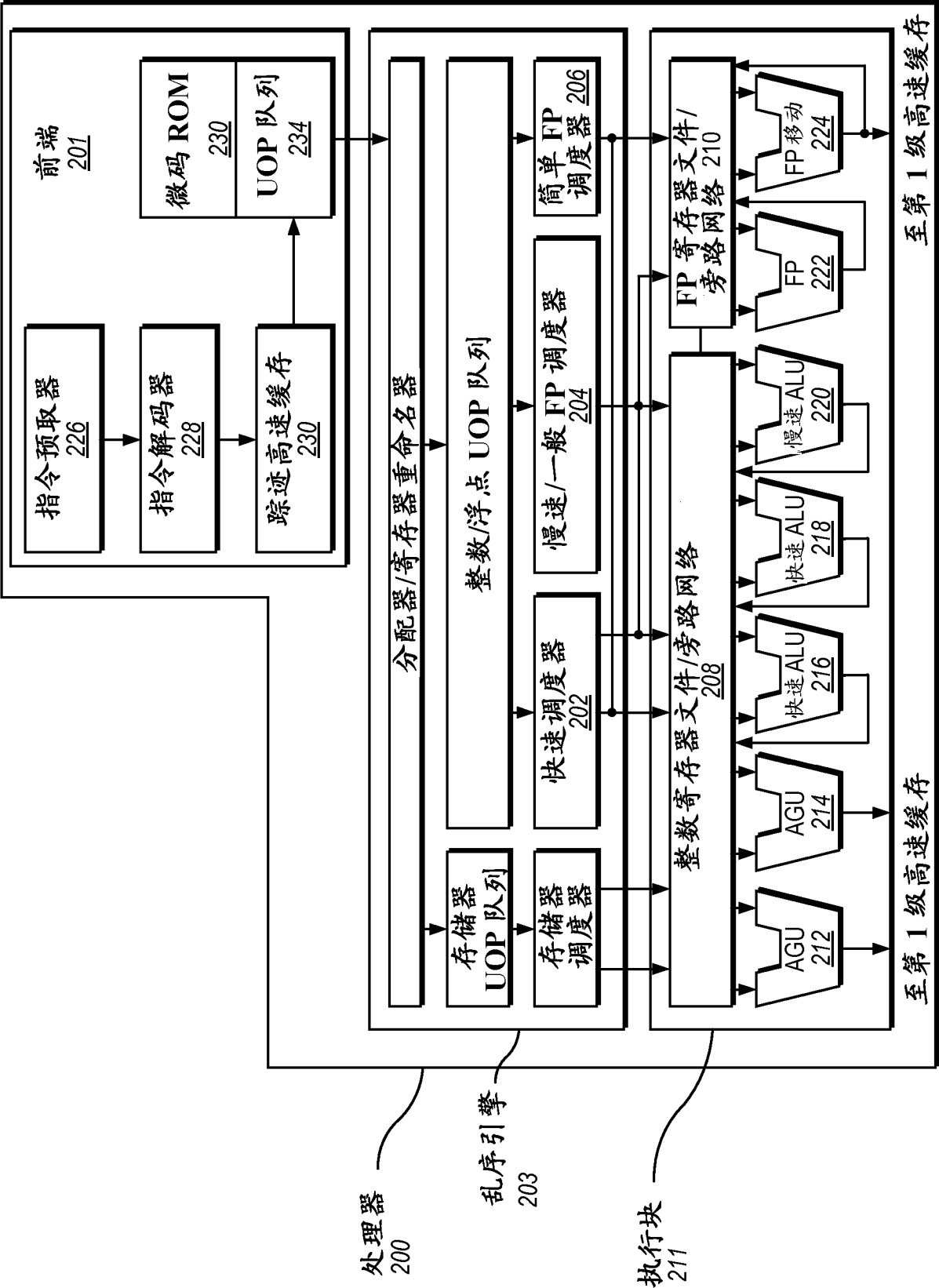


图 7

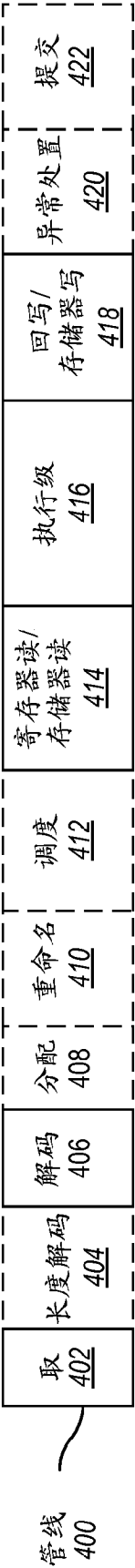


图 8A

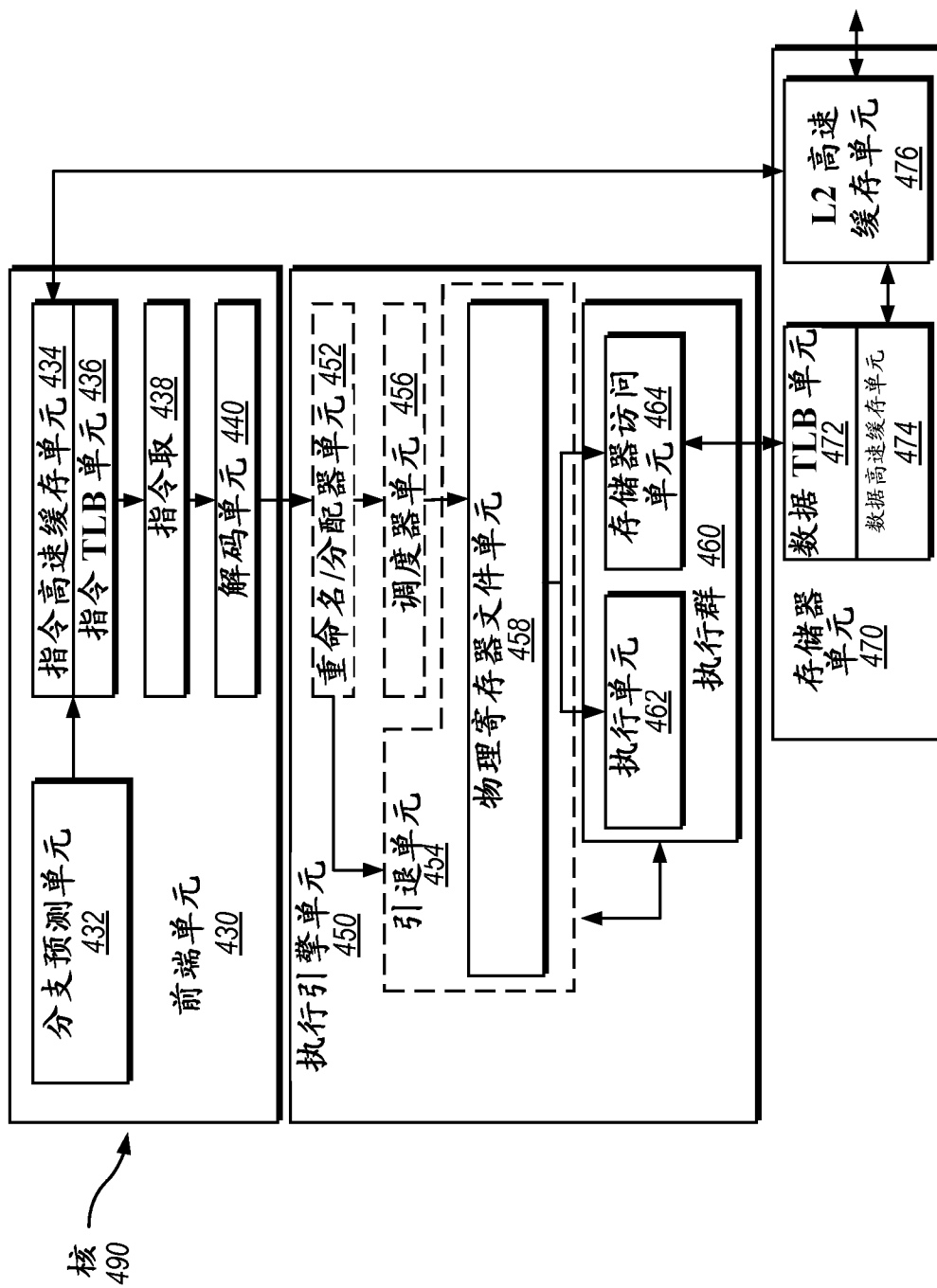


图 8B

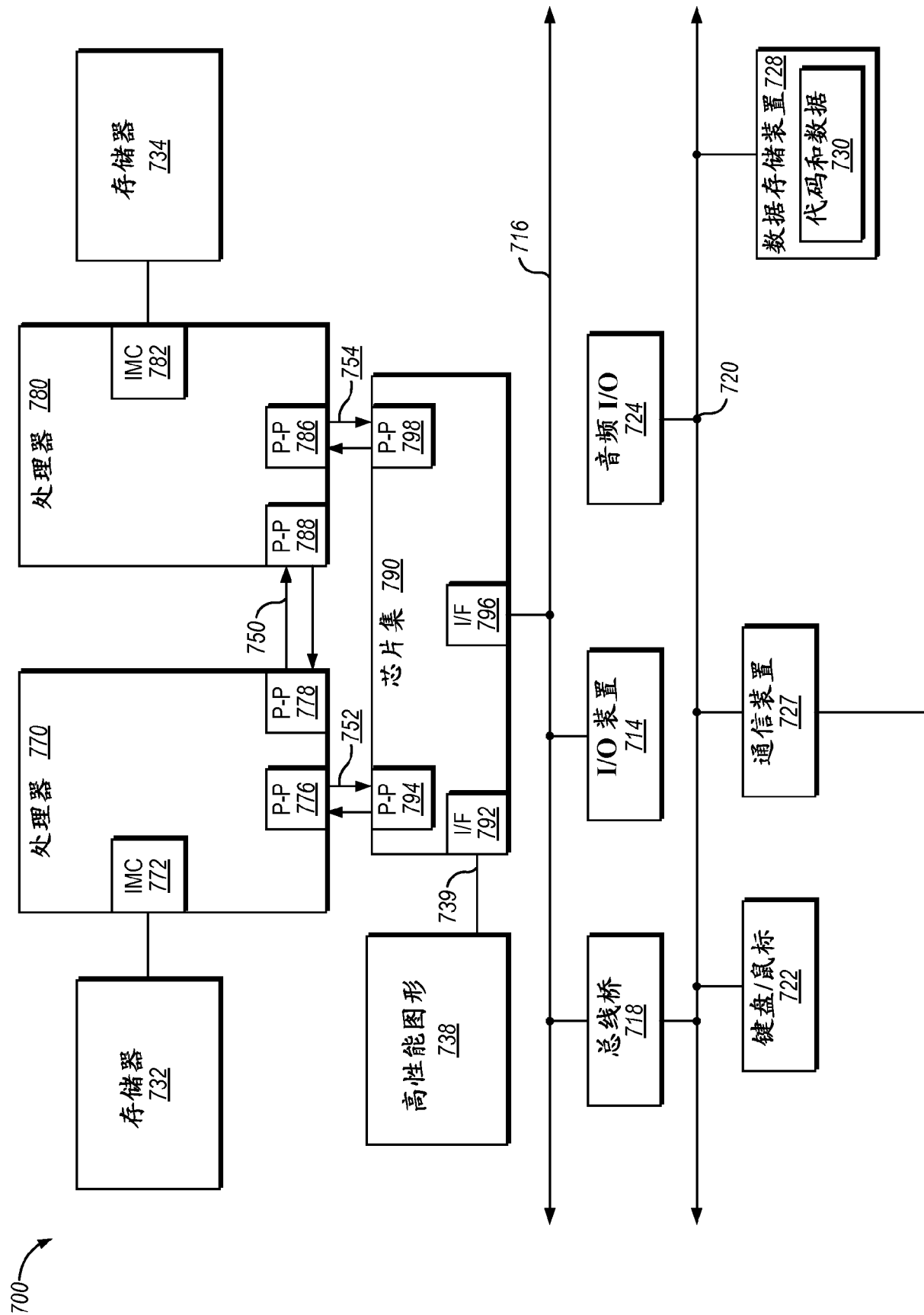


图 9

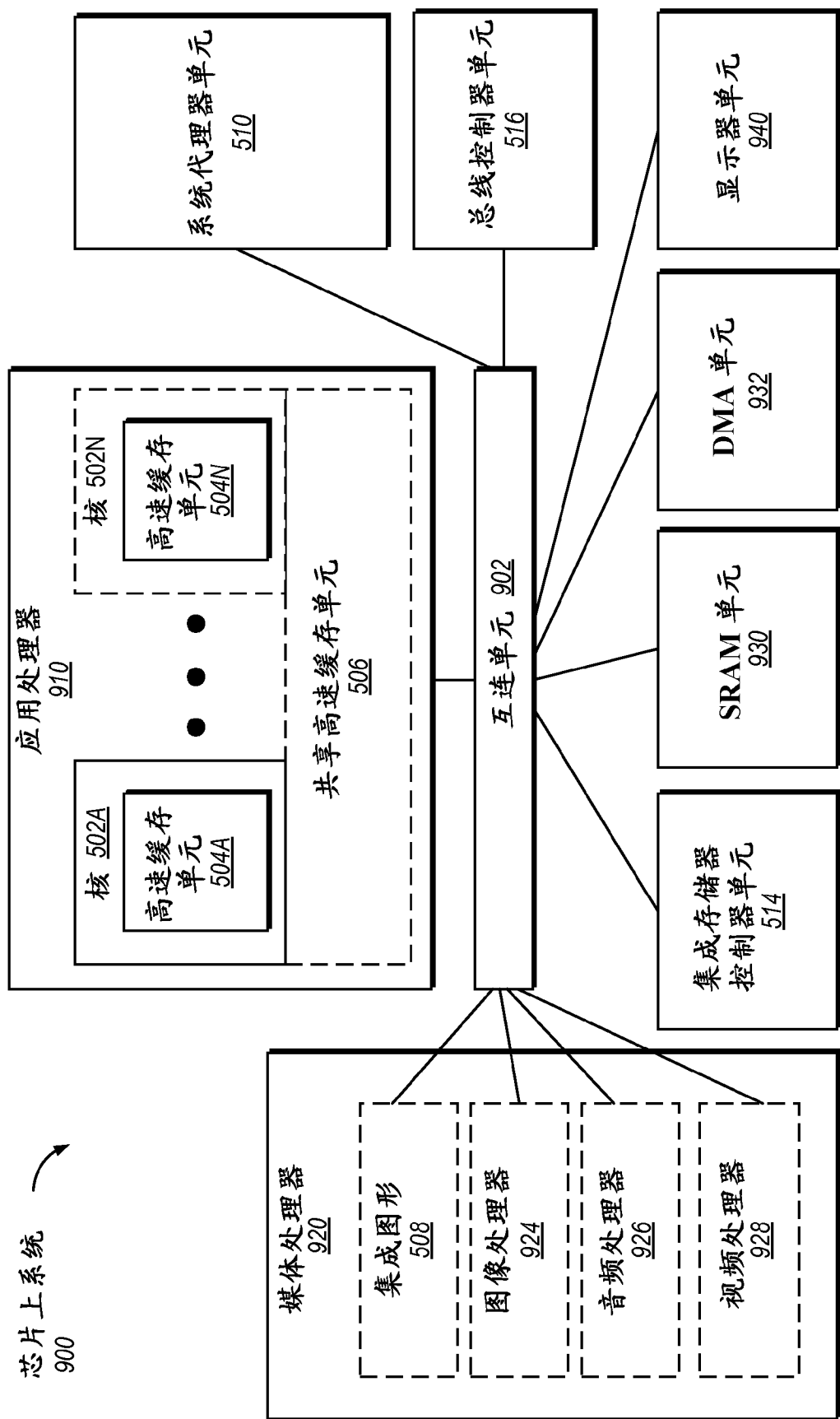


图 10

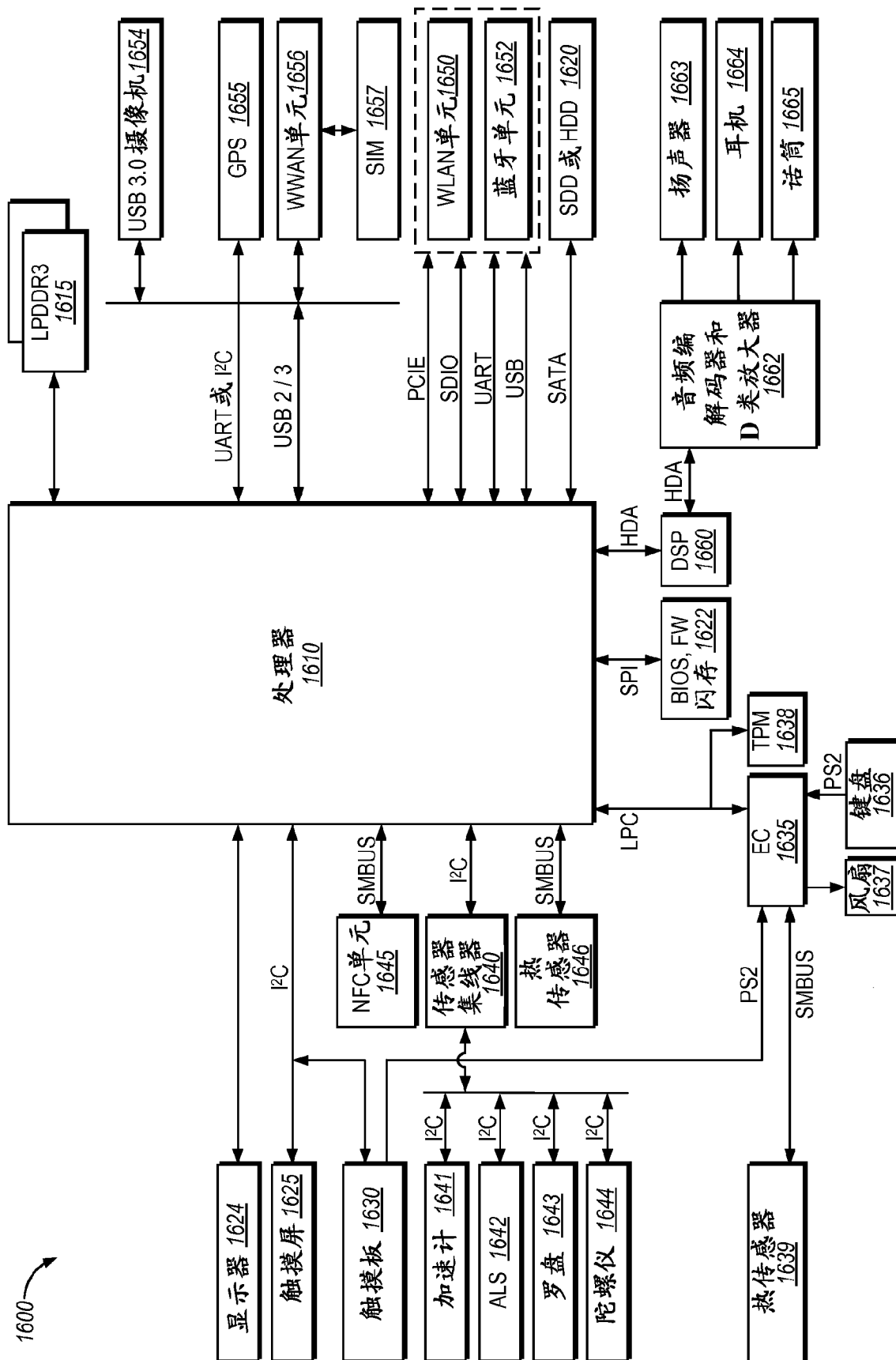


图 11

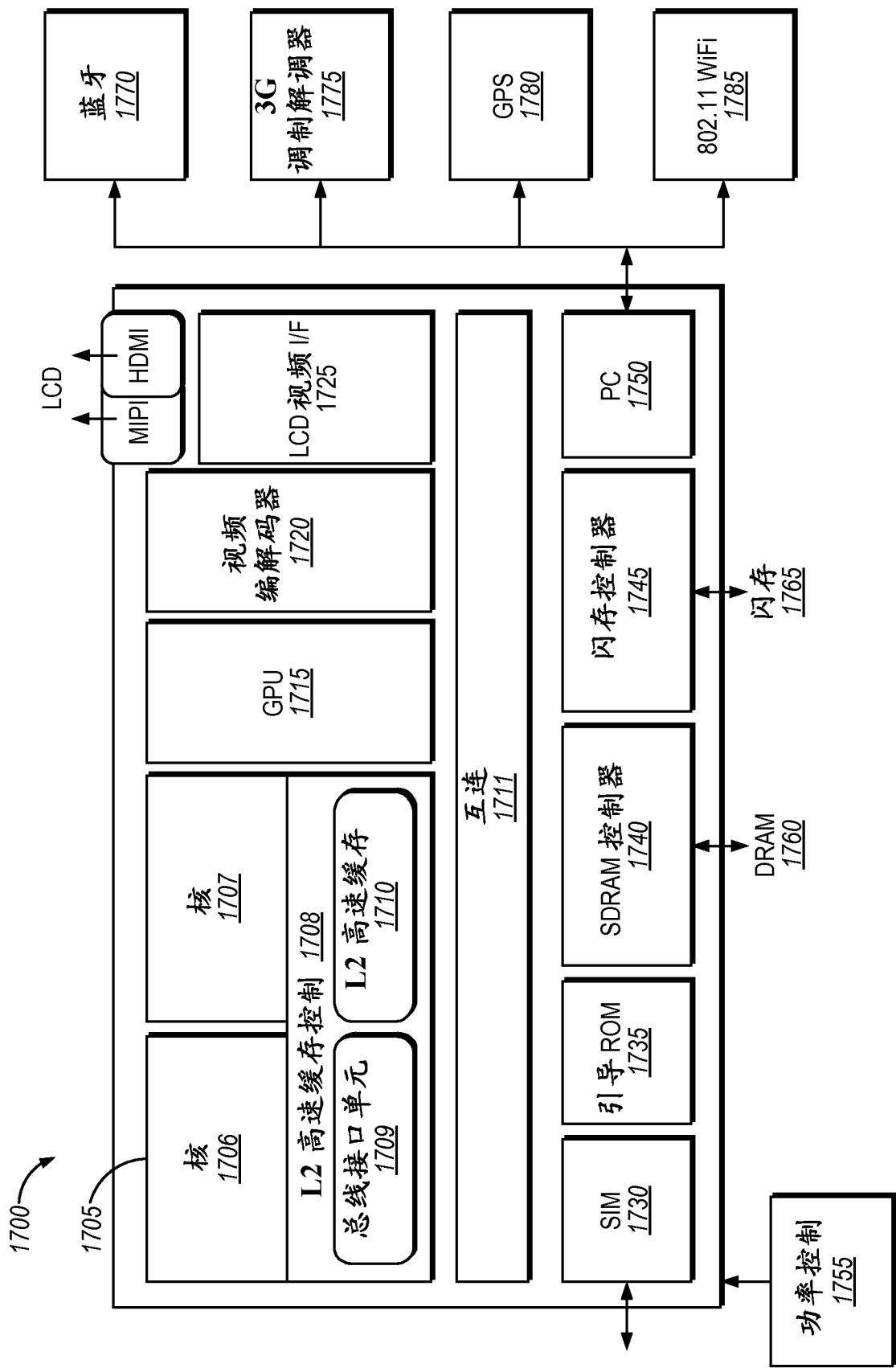


图 12

1. 一种处理系统,包括:

存储器,配置成存储与所述处理系统所执行的多个用户级应用对应的多个用户级中断处理程序地址数据结构和多个用户级高级可编程中断控制器 (APIC) 数据结构;以及

处理核,响应接收到用户级中断的通知而配置成:

设置与当前由所述处理核执行的用户级应用关联的用户级APIC数据结构中具有所述用户级中断的标识符所定义的位置的挂起的中断比特标志,以及

对于该用户级APIC数据结构所标识的一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断,调用与所述用户级应用关联的用户级中断处理程序地址数据结构所标识的用户级中断处理程序。

2. 如权利要求1所述的处理系统,其中,所述处理核还响应设置所述挂起的中断比特标志而配置成触发指示存在挂起的用户级中断的微架构事件。

3. 如权利要求2所述的处理系统,其中,所述处理核响应检测到指示存在所述挂起的用户级中断的所述微架构事件而配置成调用所述用户中断处理程序。

4. 如权利要求1-3中的任一项所述的处理系统,其中,所述多个用户级APIC数据结构的每个用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中所述比特图中的每个比特的位置对应于中断标识符。

5. 如权利要求1-3中的任一项所述的处理系统,其中,每个用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

6. 如权利要求1-3中的任一项所述的处理系统,其中,每个用户级中断处理程序地址数据结构包括与该用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

7. 如权利要求1-3中的任一项所述的处理系统,还包括处理逻辑,所述处理逻辑配置成:

响应将进入的中断标识为用户级中断而标识与当前由所述处理核执行的所述用户级应用关联的发布的中断描述符的地址;

设置与所述发布的中断描述符关联的比特图中所述中断的标识符对应的比特;以及
传送具有所述发布的中断描述符所标识的中断号的通知中断。

8. 如权利要求7所述的处理系统,其中,所述处理核还配置成将进入的中断的标识符与为所述通知中断所保留的标识符进行比较。

9. 如权利要求1-3中的任一项所述的处理系统,其中,调用所述用户级中断处理程序还包括:

将指令指针的当前值存储在栈上;以及

将所述用户级中断处理程序的地址加载到所述指令指针中。

10. 如权利要求1-3中的任一项所述的处理系统,其中,所述处理系统通过芯片上系统 (SoC) 来表示。

11. 一种方法,包括:

接收用户级中断的通知;

设置与当前由处理核执行的用户级应用关联的用户级APIC数据结构中与所述用户级中断的标识符相对应的挂起的中断比特;

使用所述用户级APIC数据结构来标识一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断;以及

使用与所述用户级应用关联的用户级中断处理程序地址数据结构来标识对于所标识的用户级中断的中断处理程序。

12. 如权利要求11所述的方法,还包括:

响应设置所述挂起的中断比特标志而触发指示存在挂起的用户级中断的微架构事件。

13. 如权利要求12所述的方法,还包括:

响应检测到指示存在所述挂起的用户级中断的所述微架构事件而调用所标识的用户中断处理程序。

14. 如权利要求13所述的方法,其中,调用所述用户级中断处理程序还包括:

将指令指针的当前值存储在栈上;以及

将所述用户级中断处理程序的地址加载到所述指令指针中。

15. 如权利要求11-14中的任一项所述的方法,其中,所述多个用户级APIC数据结构的每个用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中所述比特图中的每个比特的位置对应于中断标识符。

16. 如权利要求11-14中的任一项所述的方法,其中,每个用户级中断处理程序地址数据结构包括根据中断标识符来索引的用户级中断处理程序的地址的表。

17. 如权利要求11-14中的任一项所述的方法,其中,每个用户级中断处理程序地址数据结构包括与所述用户级中断处理程序地址数据结构关联的用户级应用要处理的多种类型的用户级中断的单个中断处理程序的地址。

18. 如权利要求11-14中的任一项所述的方法,还包括:

响应将进入的中断标识为用户级中断而标识与当前由所述处理核执行的所述用户级应用关联的发布的中断描述符的地址;

设置与所述发布的中断描述符关联的比特图中所述中断的标识符对应的比特;以及
传送具有所述发布的中断描述符所标识的中断号的通知中断。

19. 如权利要求18所述的方法,还包括:

将进入的中断的标识符与为所述通知中断所保留的标识符进行比较。

20. 一种设备,包括:

存储器;以及

处理系统,耦合到所述存储器,所述处理系统用于执行如权利要求11-14中的任一项所述的方法。

21. 一种包括可执行指令的计算机可读非暂态存储媒体,所述可执行指令在由处理系统执行时使所述处理系统执行操作,包括:

接收用户级中断的通知;

设置与当前由所述处理核执行的用户级应用关联的用户级APIC数据结构中与所述用户级中断的标识符相对应的挂起的中断比特;

使用所述用户级APIC数据结构来标识一个或更多挂起的用户级中断之中具有最高优先级的挂起的用户级中断;以及

使用与所述用户级应用关联的用户级中断处理程序地址数据结构来标识对于所标识

的用户级中断的中断处理程序。

22. 如权利要求21所述的计算机可读非暂态存储媒体,还包括使所述处理系统响应设置所述挂起的中断比特标志而触发指示存在挂起的用户级中断的微架构事件的可执行指令。

23. 如权利要求22所述的计算机可读非暂态存储媒体,还包括使所述处理系统响应检测到指示存在所述挂起的用户级中断的所述微架构事件而调用所标识的用户中断处理程序的可执行指令。

24. 如权利要求23所述的计算机可读非暂态存储媒体,其中,调用所述用户级中断处理程序还包括:

将指令指针的当前值存储在栈上;以及

将所述用户级中断处理程序的地址加载到所述指令指针中。

25. 如权利要求21-24中的任一项所述的计算机可读非暂态存储媒体,其中,所述多个用户级APIC数据结构的每个用户级APIC数据结构包括包含多个挂起的中断比特的比特图,其中所述比特图中的每个比特的位置对应于中断标识符。