



(12)发明专利

(10)授权公告号 CN 107533348 B

(45)授权公告日 2020.06.26

(21)申请号 201680022929.1

(22)申请日 2016.05.05

(65)同一申请的已公布的文献号
申请公布号 CN 107533348 A

(43)申请公布日 2018.01.02

(30)优先权数据
14/709,201 2015.05.11 US
62/169,058 2015.06.01 US

(85)PCT国际申请进入国家阶段日
2017.10.19

(86)PCT国际申请的申请数据
PCT/US2016/031001 2016.05.05

(87)PCT国际申请的公布数据
W02016/182851 EN 2016.11.17

(73)专利权人 慧与发展有限责任合伙企业
地址 美国德克萨斯州

(72)发明人 英格·林·果戈 帕特里克·唐伦
安德鲁·韦尔纳

(74)专利代理机构 北京市汉坤律师事务所
11602
代理人 张涛 吴丽丽

(51)Int.Cl.
G06F 1/20(2006.01)
G06F 1/26(2006.01)
G06F 1/30(2006.01)

(56)对比文件
US 2013139896 A1,2013.06.06,
US 2010235003 A1,2010.09.16,
US 2004268166 A1,2004.12.30,
CN 104122910 A,2014.10.29,
CN 104024975 A,2014.09.03,
US 2015070836 A1,2015.03.12,

审查员 徐书芳

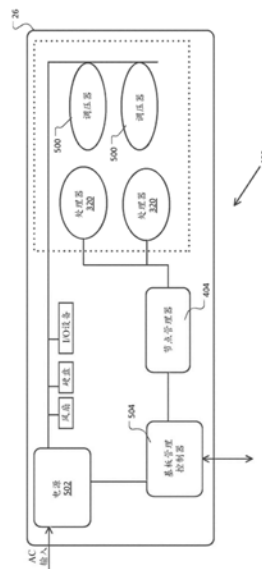
权利要求书2页 说明书20页 附图9页

(54)发明名称

热管理高性能计算系统的方法和装置及计算机可读介质

(57)摘要

一种装置和方法热管理具有多个具有微处理器的节点的高性能计算系统。为此,装置和方法监测a)高性能计算系统的环境和b)高性能计算系统的至少一部分中的至少一个的温度。作为响应,该装置和方法根据所监测的温度中的至少一个控制多个节点中的至少一个节点上的至少一个微处理器的处理速度。



1. 一种热管理高性能计算系统的方法,所述方法包括:

监测以下中的至少一个的温度:a)所述高性能计算系统的环境和b)所述高性能计算系统的至少一部分,所述高性能计算系统具有多个节点,所述多个节点具有微处理器;以及

根据所监测的温度中的至少一个,控制所述多个节点中的至少一个节点上的所述微处理器中的至少一个微处理器的处理速度,

所述方法还包括:

响应于所监测的温度超过高温阈值,减小所述处理速度至较低速度,并且在减小所述处理速度至较低速度的步骤之后,使用在包含在所述高性能计算系统中的盘管上吹送空气的干式冷却系统来对流地冷却所述盘管;以及

在减小所述处理速度和使用所述干式冷却系统之后,在所监测的温度超过规定量之后,使用在所述盘管上喷射水的湿式冷却系统传导地冷却所述盘管,在使用所述湿式冷却系统时,所述处理速度不大于所述较低速度,由此所述盘管从所述高性能计算系统的发热元件抽取能量。

2. 根据权利要求1所述的方法,其中所述高性能计算系统在具有大气温度的空间内,所述环境包括所述空间的区域,所述温度是所述空间的所述区域处的大气温度。

3. 根据权利要求1所述的方法,其中所述节点中的至少一个节点还包括存储器和电源,所述高性能计算系统的所述部分包括微处理器、存储器和电源中的至少一个。

4. 根据权利要求1所述的方法,其中监测所述温度包括监测a)所述高性能计算系统的环境的温度和b)所述高性能计算系统的至少所述部分的温度。

5. 根据权利要求4所述的方法,其中所述控制包括根据所监测的温度a)和b)两者来控制所述多个节点中至少一个节点上的所述微处理器中的至少一个微处理器的处理速度。

6. 根据权利要求1所述的方法,其中所述控制包括在规定的时段内将所述至少一个微处理器的所述处理速度减小,然后在所述规定的时段过去之后增加所述至少一个微处理器的速度。

7. 根据权利要求1所述的方法,其中所述控制包括:

减小所述至少一个微处理器的所述处理速度,至少直到所监测的温度降低至规定温度;并且

在所监测的温度降低至所述规定温度之后,增加所述至少一个微处理器的所述处理速度。

8. 一种用于热管理高性能计算系统的装置,所述装置包括:

高性能计算系统,所述高性能计算系统具有多个节点,所述多个节点具有微处理器;

温度传感器,被配置为监测以下中的至少一个的温度:a)所述高性能计算系统的环境和b)所述高性能计算系统的至少一部分;

盘管,被选择性地配置为使用在所述盘管上吹送空气的干式冷却系统来对流地冷却,并且被选择性地配置为使用在所述盘管上喷射水的湿式冷却系统来传导地冷却,由此所述盘管从所述高性能计算系统的发热元件抽取能量;以及

处理器控制器,与所述温度传感器可操作地联接,所述处理器控制器被配置为根据所监测的温度中的至少一个来控制所述多个节点中的至少一个节点上的所述微处理器中的至少一个微处理器的处理速度。

9. 根据权利要求8所述的装置,其中所述高性能计算系统在具有大气温度的空间内,所述环境包括所述空间的区域,所述温度是所述空间的所述区域处的大气温度。

10. 根据权利要求8所述的装置,其中所述多个节点中的至少一个还包括存储器和电源,所述高性能计算系统的所述部分包括所述微处理器、存储器和电源中的至少一个。

11. 根据权利要求8所述的装置,其中所述温度传感器被配置为监测a) 所述高性能计算系统的环境的温度和b) 所述高性能计算系统的至少所述部分的温度,

所述处理器控制器被配置为根据所监测的温度a) 和b) 两者来控制所述多个节点中的至少一个节点上的所述微处理器中的至少一个的处理速度。

12. 一种计算机可读介质,其中固定有一系列计算机指令,并且该一系列计算机指令用于实现根据权利要求1~7中的任一项所述的方法。

热管理高性能计算系统的方法和装置及计算机可读介质

[0001] 优先权

[0002] 本专利申请要求2015年6月1日递交的、律师参考号为3549/163和1741、申请号为62/169,058、Eng Lim Goh作为发明人的、题目为“用于管理高性能计算系统中的节点功率的方法和装置(METHOD AND APPARATUS FOR MANAGING NODAL POWER IN A HIGH PERFORMANCE COMPUTER SYSTEM)”的美国临时申请的优先权,其公开通过引用整体合并于此。

[0003] 本专利申请还要求2015年5月11日递交的、律师参考号为3549/159和1732、申请号为62/169,058、Patrick Donlin和Andrew Warner作为发明人的、题目为“用于管理高性能计算系统中的节点功率的方法和装置(METHOD AND APPARATUS FOR MANAGING NODAL POWER IN A HIGH PERFORMANCE COMPUTER SYSTEM)”的美国临时申请的优先权,其公开通过引用整体合并于此。

技术领域

[0004] 本发明通常涉及高性能计算系统,并且更具体地,本发明涉及管理高性能计算系统中的功率。

背景技术

[0005] 在分布式处理系统中,多个处理器彼此通信并与存储设备通信以执行共享计算。由于所涉及的计算的类型通常非常复杂或需要大量的处理能力,因此这种通信速度通常必须非常高。

[0006] 高性能计算(“HPC”)系统通过使用专用硬件来进一步提高速度,这些专用硬件通常不是商业上现成可用的,例如用于桌面或服务器计算机。该专用硬件通常包括多个计算节点,该多个计算节点具有用于与其他节点上的其它专用集成电路(“ASIC”) (以及相同节点上的部件)通信的定制ASIC和多个通信信道。这样的硬件还包括处理器、存储器和其他专用于实施紧密耦合(tightly-coupled)的HPC系统的专用硬件。因此,HPC系统经常横跨多个这种互连节点划分复杂计算的执行。

[0007] HPC系统产生大量的热量。因此,适当的冷却对HPC系统的有效运作很重要。HPC系统也容易出现可能损害HPC系统完成任务的能力的错误条件。

发明内容

[0008] 根据本发明的一个实施例,装置和方法热管理具有多个具有微处理器的节点的高性能计算系统。为此,装置和方法监测以下中的至少一个的温度a)高性能计算系统的环境和b)高性能计算系统的至少一部分。作为响应,装置和方法根据所监测的温度中的至少一个控制多个节点中的至少一个上的至少一个微处理器的处理速度。

[0009] 高性能计算系统可以位于具有大气温度的空间内。在那种情况下,所述环境可以包括空间的区域(例如,所述计算机系统附近的区域),并且所述监测的温度可以包括在所

述空间的所述区域处的大气温度。

[0010] 可替代地或另外地,所述方法可以监控所述高性能计算系统的所述节点中的至少一个的微处理器、存储器和功率源的至少一个。

[0011] 所述温度可以通过监测a)高性能计算系统的环境的温度和b)至少部分高性能计算系统的温度来监测。在这种情况下,所述方法和装置可以根据所监测的温度(a)和(b)两者来控制所述多个节点中的至少一个上的微处理器的至少一个的处理速度。

[0012] 各个实施例基于多个变量控制处理器速度。例如,所述方法和装置可以在规定的时间段内将所述至少一个微处理器的处理速度减小,并且然后在所述规定的时间段过去之后增加所述至少一个微处理器的速度。作为另一个示例,该方法和装置可以减小所述处理速度,至少直到所述监测温度降低至规定温度,并且然后在所监测的温度降低到所述规定温度之后增加所述处理速度。

[0013] 为了冷却节点,所述高性能计算系统可以包括盘管,所述盘管被选择性地配置成用干式冷却系统进行对流冷却和/或用湿式冷却系统进行传导冷却。利用这样的系统,该方法和装置可以响应于所监测温度将处理速度降低到较低的速度,并且在首次减小之后用干式冷却系统冷却盘管。接下来,在减小所述处理速度和使用所述干式冷却系统之后,所述方法可以在所监测的温度超过规定量之后使用湿式冷却系统冷却盘管。此时的处理速度优选地不大于使用湿式冷却系统时的较低速度。因此,在该示例中,所述干式冷却系统可能不能够充分地冷却盘管到给定的某些性能参数,这导致切换到湿式冷却系统。在一些情况下,只要所监视的温度保持在规定的水平,减少所述处理速度可以消除对所述湿式冷却系统的需要。

[0014] 因此,所述的冷却系统可被认为以多种模式冷却,即低冷却模式和高冷却模式。为了节约用水,所述方法和装置可以确定所述低冷却模式的冷却能力,并且控制/改变至少一个微处理器的处理速度以将所述冷却系统维持在低冷却模式。除了其他方式,所述方法和装置可以通过监测所述高性能计算系统的至少部分的温度来确定所述冷却能力。

[0015] 根据另一个实施例,用于热管理高性能计算系统的装置具有温度传感器包括监测a)高性能计算系统的环境,以及b)至少一部分高性能计算系统中的至少一个的温度的温度传感器。以类似于上述先前实施例的方式,所述高性能计算系统包含具有微处理器的多个节点。所述装置还具有与所述温度传感器可操作地联接的处理器控制器。所述处理器控制器被配置为根据所监测的温度中的至少一个控制所述多个节点中的至少一个上的微处理器中的一个的处理速度。

[0016] 根据其他实施例,所述方法和装置管理包含具有微处理器的多个节点的高性能计算系统中的错误。为此,所述方法和装置检测所述高性能计算系统的规定错误条件,并且在检测到规定错误条件之后,减小所述多个节点中的至少一个上的微处理器中的至少一个的处理速度。

[0017] 规定的错误条件可以包括可校正错误和所述节点中的至少一个的温度读数中的至少一个。例如,所述可校正错误可以包括存储器可校正错误和网络可校正错误中的至少一个。此外,所述方法和装置可以允许所述处理速度维持当前水平(即,由所述系统和所述微处理器指定的正常处理速度),然后在检测到规定的多个规定错误条件之后将当前水平的处理速度减小。

[0018] 在一些实施例中,所述方法和装置可以检测所述高性能计算系统的多个错误条件,然后根据错误条件的数量来降低所述处理速度。为了纠正错误,各个实施例可以热交换所述高性能计算系统的至少一部分,或停止所述高性能计算系统的至少一部分的执行。在一些情况下,当给定节点执行任务时,所述方法和装置可以检测所述给定节点上的错误条件,并推迟减小处理速度,直到完成任务之后。

[0019] 根据又一个实施例,用于管理具有多个具有微处理器的节点的高性能计算系统中的错误的装置具有配置为检测所述高性能计算系统的规定错误条件的错误检查器。此外,所述装置还具有与所述错误检查器可操作地联接的处理器控制器。所述处理器控制器被配置为在检测到所述规定错误条件之后减小所述多个节点中的至少一个上的微处理器的至少一个的处理速度。

[0020] 根据其他实施例,所述方法和装置热管理具有多个具有微处理器的节点的高性能计算系统和用于冷却所述多个节点的冷却系统。

[0021] 为此,所述方法和装置首先检测冷却系统中的故障,并且在检测到所述故障之后响应地将至少一个微处理器的处理速度降低到多个节点中的至少一个节点上的非零速率。

[0022] 本发明的说明性实施例被实施为具有其上具有计算机可读程序代码的计算机可用介质的计算机程序产品。根据常规过程,所述计算机可读代码可以被计算机系统读取和使用。

附图说明

[0023] 参考下面紧跟着总结的附图,本领域技术人员应该从下面的“具体实施方式”中更充分地理解本发明的各个实施例的优点。

[0024] 图1示意性示出了根据本发明的一个实施例的HPC系统的逻辑视图。

[0025] 图2示意性示出了图1的HPC系统的物理视图。

[0026] 图3示意性示出了图1的HPC系统的刀片机箱的细节。

[0027] 图4示意性示出了图1中的高性能计算系统中的多个节点的另一个表示。

[0028] 图5示意性示出了图4的节点中一个节点的另一个视图。

[0029] 图6示出了根据本发明的说明性实施例的管理高性能计算系统中的功耗的过程。

[0030] 图7示意性地示出了根据本发明的各个实施例的具有高性能计算系统以及管理高性能计算系统中的功耗的装置的空间的视图。

[0031] 图8示出了根据本发明的说明性实施例的管理高性能计算系统中的热曲线的过程。

[0032] 图9示出了根据本发明的说明性实施例的管理高性能计算系统中的错误的过程。

具体实施方式

[0033] 说明性实施例根据各种环境和/或性能标准管理高性能计算系统中的节点的功率。为此,在一个实施例中,处理器控制器选择性地减少一个或多个节点的功率使用,以减少湿式冷却系统或混合冷却系统所需的耗水量。当适当地控制时,这样的实施例可以减少或基本上消除对湿式冷却系统的需要,并且在一些情况下是对昂贵的冷却装置的需要。事实上,如果相关实施例检测到冷却系统本身的故障,则可以减少功率使用。因此,如果冷却

系统不能充分地对其进行冷却,则处理器控制器可以减小功耗以减少系统过热的可能性。

[0034] 在另一个实施例中,在检测到高性能计算系统中的错误条件之后,处理器控制器减小其一个或多个节点的处理速度。例如,控制器可以允许一些或全部当前执行的进程在关闭用于服务的系统中的一些或全部或者执行热插拔操作之前结束(或到达适当的停止点)。虽然使进程能够结束或达到适当的停止点,但是减小处理速度应当增加系统能被维修之前不会失效的可能性。这种受控制的速度减小到最终服务(ultimate service)有助于保持系统的弹性和效率。

[0035] 在其他实施例中,在高性能计算系统上执行的应用程序动态地改变其允许的功率范围。为此,使用与系统中的多个节点相关的信息,中央控制器设置系统的初始功率范围。配置为在多个节点的子集上执行的应用(即,该应用在系统的一个或多个节点上执行)最初也旨在遵循该初始功率范围。然而,响应于改变的或独特的要求,应用程序可以为其节点请求新的/不同的功率范围。事实上,即使在接收到初始功率范围之前,应用程序也可以请求这个不同的功率范围。

[0036] 因此,这个后面的实施例的中央控制器可以改变执行应用程序的一个或多个节点的功率范围。因此,一个或多个节点随后使用该新的功率范围来执行应用程序所要求的进程。这种动态功率控制可以更有效地协调系统中的多个节点之间的功率使用和处理,从而有效地提高整体系统性能。

[0037] 下面讨论这些和其他实施例的细节。

[0038] 系统结构

[0039] 图1示意性示出了可以与本发明的说明性实施例一起使用的示例性高性能计算系统100的逻辑视图。具体地,如本领域技术人员已知的,“高性能计算系统”或“HPC系统”是具有使用硬件互连紧密耦合的多个模块化计算资源的计算系统,以使处理器可以使用公共存储器地址空间直接访问远程数据。

[0040] HPC系统100包括用于提供计算资源的多个逻辑计算分区120、130、140、150、160、170、以及用于管理多个分区120-170的系统控制台110。HPC系统中的“计算分区”(或“分区”)是运行单个操作系统实例并具有公共存储器地址空间的计算资源的管理性分配。分区120-170可以使用逻辑通信网络180与系统控制台110通信。希望执行计算的诸如科学家或工程师的系统用户可以从使用系统控制台110的系统操作者请求计算资源,以分配并管理这些资源。以下描述计算资源分配给分区的分配。HPC系统100可以具有如下面更详细描述的管理性地分配的任何数量的计算分区,并且通常仅具有一个包含所有可用计算资源的分区。因此,该图不应被视为限制本发明的范围。

[0041] 诸如分区160的每个计算分区可以在逻辑上看起来好像是单个计算设备,类似于台式计算机。因此,分区160可以执行软件,该软件包括使用基本输入/输出系统(“BIOS”)192的单个操作系统(“OS”)实例191(如同BIOS和OS在本领域中一起使用),以及用于一个或多个系统用户的应用软件193。

[0042] 因此,如图1所示,计算分区具有由系统操作者分配给计算分区的各种硬件,该各种硬件包括一个或多个处理器194、易失性存储器195、非易失性存储器196以及输入和输出(“I/O”)设备197(例如,网络端口、视频显示设备、键盘等)。然而,在像图1中的实施例的

HPC系统中,每个计算分区具有比一般台式计算机大的多的处理能力和存储器。OS软件可以包括例如华盛顿州雷德蒙德的微软公司的**Windows®**操作系统或Linux操作系统。此外,尽管BIOS可能由硬件制造商(如加利福尼亚州圣克拉拉市的英特尔公司)作为固件提供,但BIOS通常根据HPC系统设计人员的需求进行定制,以支持高性能计算,如下文详细所述。

[0043] 作为系统控制台110的系统管理角色的一部分,系统控制台110充当计算分区120-170的计算能力与系统操作或其他计算系统之间的接口。为此,系统控制台110代表系统操作员向HPC系统硬件和软件发出命令,：1) 启动硬件,2) 将系统计算资源划分为计算分区,3) 初始化分区,4) 监控每个分区的运行状况以及其中产生的任何硬件或软件错误,5) 将操作系统和应用软件分布到各个分区,6) 使操作系统和软件执行,7) 备份其中的分区或软件的状态,8) 关闭应用软件,以及9) 关闭计算分区或整个HPC系统100,以及其他命令。这些特定的功能在下面的标题为“系统操作”的部分中有更详细的描述。

[0044] 图2示意性示出了根据图1的实施例的高性能计算系统100的物理视图。包括图1的HPC系统100的硬件被虚线围绕。HPC系统100连接到客户数据网络210 以便于客户访问。

[0045] HPC系统100包括执行系统控制台110的功能的系统管理节点(“SMN”)220。管理节点220可以被实施为由客户或HPC系统设计者提供的台式计算机、服务器计算机或其他类似的计算设备,并且包括控制HPC系统100所需的软件(即,系统控制台软件)。

[0046] 可以使用数据网络210访问HPC系统100,数据网络210可以包括本领域已知的任何数据网络,诸如客户局域网(LAN)、虚拟专用网(“VPN”)、互联网等等或者这些网络的组合。这些网络中的任何一个可以允许多个用户远程和/或同时访问HPC系统资源。例如,管理节点220可以由客户计算机230通过使用本领域已知的工具(诸如**Windows®**远程桌面服务或UNIX安全外壳)远程登录来访问。如果客户愿意,则对HPC系统100的访问可以被提供到远程计算机240。远程计算机240可以如刚刚描述的那样通过登录到管理节点220来访问HPC系统,或者使用本领域技术人员已知的网关或代理系统来访问HPC系统。

[0047] HPC系统100的硬件计算资源(例如,图1所示的处理器、存储器、非易失性存储和I/O设备)由一个或多个“刀片机箱”集合提供,例如被管理并分配到计算分区中的图2中示出的刀片机箱252、254、256、258。刀片机箱是一种电子机箱,被配置为容纳多个称为“刀片”的可堆叠、模块化电子电路板、为刀片提供功率并提供多个刀片之间的高速数据通信。每个刀片包括足够的计算硬件,以用作独立的计算服务器。刀片机箱的模块化设计允许刀片以最少的布线和垂直空间连接到电源线和数据线。

[0048] 因此,每个刀片机箱(例如刀片机箱252)具有用于管理刀片机箱252中的系统功能的机箱管理控制器260(也称为“机箱控制器”或“CMC”),以及用于提供计算资源的多个刀片262、264、266。每个刀片(例如刀片262)将该刀片的硬件计算资源贡献给HPC系统100的集合总资源。系统管理节点220使用诸如机箱控制器260的机箱控制器来管理整个HPC系统100的硬件计算资源,而每个机箱控制器依次管理仅在其刀片机箱中的刀片的资源。下面将更详细地描述,机箱控制器260通过本地管理总线268物理地和电性地耦合到刀片机箱252内的刀片262-266。另外的刀片机箱254-258中的硬件类似地配置。

[0049] 机箱控制器使用管理连接270彼此通信。管理连接270可以是例如运行以太网通信协议的高速LAN,或其他数据总线。相比之下,刀片使用计算连接280彼此通信。为此,计算连接280说明性地具有高带宽、低延迟的系统互连,诸如由加利福尼亚州米尔皮塔斯的

Silicon Graphics International Corp.开发的NumaLink。

[0050] 机箱控制器260向HPC系统的其余部分提供系统硬件管理功能。例如,机箱控制器260可以从SMN 220接收系统启动命令,并且通过使用本地管理总线268向刀片262-266中的每一个发出启动命令来进行响应。类似地,机箱控制器260可以从一个或多个刀片262-266接收硬件错误数据,并且存储该信息以用于随后与其他机箱控制器存储的错误数据的一起分析。在诸如图2所示的一些实施例中,SMN 220或客户计算机230被提供访问单个主机箱控制器260,单个主机箱控制器260处理系统管理命令以控制HPC系统100,并将这些命令转发到其他机箱控制器。然而,在其他实施例中,SMN 220直接耦合到管理连接270,并且分别向每个机箱控制器发出命令。本领域普通技术人员可以考虑允许相同类型的功能的这些设计的变化,但为了清楚,仅呈现这些设计。

[0051] 可以如本领域已知的那样,提供刀片机箱252、刀片机箱252的刀片262-266和本地管理总线268。然而,机箱控制器260可以使用由HPC系统设计者提供的硬件、固件或软件来实施。每个刀片为HPC系统100提供在独立计算机服务器领域中已知的一些数量的处理器、易失性存储器、非易失性存储器和I/O设备。然而,每个刀片还具有硬件、固件和/或软件,以允许将这些计算资源组合在一起并作为计算分区共同处理,如以下在题为“系统操作”的部分中更详细地描述。

[0052] 虽然图2示出了具有四个机箱的HPC系统以及每个机箱中三个刀片。然而,应当理解,这些图不限制本发明的范围。HPC系统可以具有数十个机箱和数百个刀片;实际上,HPC系统通常是期望的,因为它们提供非常大量的紧密耦合的计算资源。

[0053] 图3更详细地示意性示出了单个刀片机箱252。在该图中,省略了与现在描述无关的部分。机箱控制器260被示出为具有到系统管理节点220和管理连接270的连接。机箱控制器260可以被提供有用于存储机箱管理数据的机箱数据存储器302。在一些实施例中,机箱数据存储器302是易失性随机存取存储器(“RAM”),其中,即使一个或多个计算分区已失效(例如,由于OS崩溃)或刀片故障,只要电力被施加到刀片机箱252,机箱数据存储器302中的案例数据就可由SMN 220访问。在其他实施例中,机箱数据存储器302是诸如硬盘驱动器(“HDD”)或固态驱动器(“SSD”)之类的非易失性存储器。在这些实施例中,机箱数据存储器302中的数据在HPC系统已断电并重新启动之后是可访问的。

[0054] 图3示出了用于讨论目的的刀片262和264的具体实施方式的相关部分。刀片262包括以类似于机箱控制器在机箱级执行的功能的方式在刀片级执行系统管理功能的刀片管理控制器310(也称为“刀片控制器”或“BMC”)。有关机箱控制器和刀片控制器操作的更多详细信息,请参阅下面的“HPC系统操作”一节。刀片控制器310可以被实施为由HPC系统设计者设计的定制硬件,以用于允许与机箱控制器260通信。此外,刀片控制器310可以具有其自己的RAM 316,以执行其管理功能。机箱控制器260使用本地管理总线268与每个刀片的刀片控制器通信,如图3和之前的图所示。

[0055] 刀片262还包括连接到RAM 324、326的一个或多个微处理器320、322(可替代地称为“处理器320或322”或一般地称为“处理器320”)。刀片262可以被交替配置,使得多个处理器可以在单个总线上访问公共的一组RAM,如本领域已知的。还应当理解,如本领域已知的,处理器320、322可以包括任何数量的中央处理单元(“CPU”)或核。刀片262中的处理器320、322连接到其他项目,诸如与I/O设备332通信的数据总线、与非易失性存储器334通信的数

据总线以及在独立计算系统中普遍地建立的其他总线。(为了清楚起见,图3仅示出了从处理器320到这些其他设备的连接)。处理器320、322可以是例如由英特尔公司制造的英特尔®酷睿™处理器。I/O总线可以是例如PCI或PCI Express(“PCIe”)总线。存储总线可以是例如SATA,SCSI或光纤通道总线。应当理解,根据本发明的说明性实施例,可以使用其他总线标准、处理器类型和处理器制造商。

[0056] 每个刀片(例如,刀片262和264)包括控制该刀片的大部分功能的专用集成电路340(也称为“ASIC”、“集线器芯片”或“集线器ASIC”)。更具体地,为了将处理器320,322、RAM 324,326和其他设备332,334逻辑地连接在一起以形成被管理的多处理器(可选地)相干共享(coherently shared)的分布式存储器HPC系统,处理器320、322电连接到集线器ASIC 340。集线器ASIC 340因此提供由SMN 220、机箱控制器260和刀片控制器310产生的HPC系统管理功能与刀片262的计算资源之间的接口。

[0057] 在这个连接中,集线器ASIC 340可以通过直接连接、或者通过现场可编程门阵列(“FPGA”)342、或用于在集成电路之间传递信号的类似的可编程设备与刀片控制器310连接。本领域技术人员可以选择集线器ASIC 340和刀片控制器310之间的适当连接。直接连接或间接连接的讨论不应限制本发明的各种实施例。

[0058] 具体地,响应于由机箱控制器260发出的命令,在刀片控制器310的输出管脚(pin)上产生信号。在间接连接的情况下,这些信号由FPGA 342转换为集线器ASIC 340的某些输入管脚的命令,反之亦然。例如,由刀片控制器310从机箱控制器260接收的“通电”信号除了别的以外还需要向集线器ASIC 340上的某个管脚提供“通电”电压;FPGA 342有助于实施这一任务。

[0059] FPGA 342的现场可编程性质允许刀片控制器310和ASIC 340之间的接口在制造之后可重新编程。因此,例如,刀片控制器310和ASIC 340可以被设计为具有某些通用功能,并且FPGA 342可以有利地用于以特定于应用的方式对这些功能的使用进行编程。如果在任一模块中发现硬件设计错误,则可以更新刀片控制器310和ASIC 340之间的通信接口,允许快速的系统修复而不需要制造新的硬件。

[0060] 还结合其作为计算资源和系统管理之间的接口的角色,集线器ASIC 340通过高速处理器互连344连接到处理器320、322。在一个实施例中,处理器320、322由Intel公司制造,Intel公司为此提供了英特尔®QuickPath互连(“QPI”),并且集线器ASIC 340包括用于使用QPI与处理器320、322进行通信的模块。其他实施例可以使用其他处理器互连配置。

[0061] 每个刀片中的集线器芯片340还提供用于高带宽、低延迟数据通信的与其他刀片的连接。因此,集线器芯片340包括与连接不同刀片机箱的计算连接280连接的链路350。该链路350可以使用例如网络电缆来实施。集线器ASIC 340还包括与同一刀片机箱252中的其他刀片的连接。刀片262的集线器ASIC 340通过机箱计算连接352连接到刀片264的集线器ASIC 340。机箱计算连接352可以被实施为刀片机箱252的背板上的数据总线,而不是使用联网电缆,有利地允许高性能计算任务所需的刀片之间的非常高速的数据通信。可以使用NumaLink协议或类似协议来实施机箱间计算连接280和机箱内计算连接352之间的数据通信。

[0062] HPC系统操作

[0063] 系统管理命令通常从SMN 220通过管理连接270传播到刀片机箱(及其机箱控制器),然后传播到刀片(及其刀片控制器),最后传播到使用系统计算硬件实施命令的集线器ASICS。

[0064] 作为具体的示例,考虑HPC系统通电的过程。根据示例性实施例,当系统操作者从SMN 220发出“通电”命令时,HPC系统100通电。SMN 220通过刀片机箱 252-258各自的机箱控制器(诸如刀片机箱252中的机箱控制器260)将该命令传播到刀片机箱252-258中的每一个。每个机箱控制器依次通过其各自的刀片控制器(例如刀片262的刀片控制器310)向其刀片机箱中的相应刀片中的每一个发出“通电”命令。刀片控制器310使用FPGA 342向其对应的集线器芯片340发出“通电”命令,该FPGA 342在允许集线器芯片340初始化的集线器芯片340的一个管脚上提供信号。其他命令类似地传播。

[0065] 一旦HPC系统通电,HPC系统的计算资源可能会划分为计算分区。分配给每个计算分区的计算资源的数量是一个管理决策。例如,客户可能需要完成一些项目,并且每个项目预计需要一定量的计算资源。不同的项目可能需要不同比例的处理能力,内存和I/O设备使用,而不同的刀片可能具有不同数量的安装资源。HPC系统管理员在对HPC系统100的计算资源进行分区时考虑到这些注意事项。对计算资源的分区可以通过对每个刀片的RAM 316编程来实施。例如,SMN 220可以在读取系统配置文件之后发出适当的刀片编程命令。

[0066] 根据任何管理需要,HPC系统100的集体硬件计算资源可以划分为计算分区。因此,例如,单个计算分区可以包括一个刀片机箱252中的一些刀片或全部刀片的计算资源、多个刀片机箱252和254的全部刀片的计算资源、一个刀片机箱252的一些刀片和刀片机箱254的全部刀片的计算资源,整个HPC系统100的所有计算资源,以及其他类似的组合。硬件计算资源可以被静态地划分,在这种情况下,需要重新启动整个HPC系统100来重新分配硬件。可替代地且优选地,在HPC系统100通电时,硬件计算资源被动态分区。以这种方式,未指定的资源可以被分配给分区,而不中断其他分区的操作。

[0067] 应当注意,一旦HPC系统100被适当地分区,每个分区可以被认为是独立的计算系统。因此,两个或多个分区可以被组合以在HPC系统100内部形成逻辑计算组。例如如果特定的计算任务被分配给比单个操作系统可以控制的多的处理器或存储器,则这种分组可能是必要的。例如,如果单个操作系统只可以控制64个处理器,但是特定的计算任务需要256个处理器的组合能力,则四个分区可以以这种组的形式被分配给任务。可以使用本领域已知的技术来实施该分组,诸如在每个计算分区上安装相同的软件并且向分区提供VPN。

[0068] 一旦创建了至少一个分区,则分区可以被启动并且该分区的计算资源被初始化。诸如分区160的每个计算分区可以被逻辑地看作具有单个OS 191和单个BIOS 192。如本领域已知的那样,BIOS是电探测可用硬件并初始化可用硬件至已知状态的指令集合,以使OS可以启动,并且BIOS通常被提供在每个物理服务器上的固件芯片中。然而,单个逻辑计算分区160可以跨越多个刀片,或甚至跨越多个刀片机箱。刀片可以被称为“计算节点”或简称为“节点”,以强调其分配给特定分区。

[0069] 根据本发明的实施例启动分区需要对从库存购买的刀片机箱进行许多修改。具体来说,修改每个刀片中的BIOS以确定相同计算分区中的其他硬件资源,而不仅仅是同一刀片或刀片机箱中的硬件资源。在由SMN 220发出启动命令之后,集线器ASIC 340最终向处理器320提供适当的信号,以使用BIOS指令开始启动过程。依次地,BIOS指令从集线器ASIC

340获取分区信息,诸如:分区中的标识(节点)号、节点互连拓扑,存在于分区中的其他节点中的设备列表、分区中的所有节点使用的主时钟信号,等等。具有这些信息,处理器320可以采取所需的任何步骤来初始化刀片262,包括1)非HPC特定步骤,诸如初始化I/O设备332和非易失性存储器334,以及2)HPC特定步骤,诸如将本地硬件时钟同步到主时钟信号、初始化给定节点中的HPC专用硬件、管理包含关于分区中哪些其他节点访问其RAM的信息的存储器目录、以及准备分区范围物理内存映射(partition-wide physical memory map)。

[0070] 此时,每个物理BIOS具有自己的分区视图,并且每个节点中的所有计算资源都准备好用于加载OS。然后,BIOS根据多处理器系统领域已知的技术读取OS映像并执行该OS映像。即使硬件本身分散在多个刀片机箱和刀片之间,BIOS会向OS呈现分区硬件的视图,就像分区硬件都存在于单个、非常大的计算设备中一样。以这种方式,单个OS实例在分配给该OS实例的分区的刀片机箱和刀片中的某些或较佳地全部上传播(spread)。不同的操作系统可以被安装在各个分区上。如果OS映像不存在,例如紧接着创建分区之后,可以在分区启动之前使用本领域已知的过程安装OS映像。

[0071] 一旦OS安全地执行,该OS的分区可以作为单个逻辑计算设备来操作。用于执行预期计算的软件可以由HPC系统操作员安装到各个分区。然后,用户可以登录SMN 220。例如,从SMN 220对用户各自的分区的访问可以使用基于登录凭证的数据载体安装和目录许可来控制。系统操作员可以监控每个分区的运行状况,并在检测到硬件或软件错误时采取补救措施。长时间运行的应用程序的当前状态可以定期地或根据系统操作员或应用程序用户的命令来保存到非易失性存储器,以防止在系统或应用程序崩溃的情况下丢失工作。系统操作员或系统用户可以发出关闭应用程序的命令。HPC分区的其他操作可以是本领域普通技术人员已知的。当需要管理时,系统操作员可以完全关闭计算分区,重新分配或释放分区中的计算资源,或关闭整个HPC系统100。

[0072] 一般功率管理

[0073] 如上所述,说明性实施例管理节点功率以确保更有效率且有效的操作。为此,图4示意性示出了图1的高性能计算系统100中的多个节点(以下称为“节点400”)的另一个表示。具体地,该图示出了与中央功率控制器401通信以协调其功率需求的、标记为节点1、节点2、...、节点N的多个节点400。换句话说,如下面更详细地讨论的那样,在收集相关系统信息之后,中央功率控制器401设置节点400需要操作的功率范围。替代实施例在默认功率范围处简单地运行节点400。

[0074] 在说明性实施例中,中央功率控制器401与节点400分开地实施,因此中央功率控制器401具有独立部件,例如独立板(例如,主板,子板等)、微处理器、存储器等。例如,中央功率控制器401可以至少部分地被实施为控制计算机系统功率使用的系统管理控制器(“SMC”,由加利福尼亚州圣克拉拉市的英特尔公司发布)。节点400使用任何上述互连机制与中央功率控制器401进行通信。

[0075] 除了在本说明书的之前部分中描述的针对每个节点400部件之外,该表示仅简单地示出了每个节点400,该每个节点400执行应用程序(以下称为“应用程序402”)以及具有用于管理其功率功能的节点管理器404。尽管仅示出了一个应用程序402,但是该图应该被认为表示多个应用程序402,每个应用程序402横跨任何节点400中的一个或多个执行。例如,第一应用程序402可以横跨节点1、3和44执行,而第二应用程序可以仅在节点1上执行,

并且第三应用程序402可以横跨节点3、5和10-60执行。因此,在一个或几个节点上执行的单个应用程序的讨论仅用于简单目的。

[0076] 可以被实施为集线器ASIC 340的一部分或作为另一个部件的每个节点管理器404优选地具有一组规定的功率管理功能。为此,每个节点管理器404可以被认为是每个节点400上的专用管理设备。作为经由HPC100的管理网络通信的功能模块,每个节点管理器404测量并控制节点功率,提供电力和使用统计,并且管理并实施功率策略。因此,节点管理器404可以从电源读取平均功率,通过减少/增加处理器频率来减少/增加对处理器的功率消耗,以及将功率和频率调节到目标限制。例如,节点管理器404可以管理和实施由中央功率控制器401设置的功率策略。

[0077] 实际上,应该注意的是,如图1-3和图5(下面所讨论的图5),图4仅示意性地一般地示出某些部件。本领域技术人员应当理解,可以以各种常规方式来实施这些部件中的每一个,诸如通过使用横跨一个或多个其他功能部件的硬件、软件或硬件和软件的组合来实现。例如,中央功率控制器401可以使用执行固件的多个微处理器来实施。作为另一个示例,中央功率控制器401可以使用一个或多个专用集成电路(即,“ASIC”)和相关软件来实施,或使用ASIC、分立电子部件(例如,晶体管)和微处理器的组合来实施。因此,图4的单个框中的中央功率控制器401和其他部件的表示仅仅是为了简单的目的。事实上,在一些实施例中,图4的中央功率控制器401横跨多个不同的机器(未必在相同的外壳或机箱内的机器)而分布。

[0078] 应当重申,这些图中的部件的示意图表示了它们所代表的实际设备的显著简化表示。本领域技术人员应该理解,这样的设备可以具有许多其他物理和功能部件。因此,这个讨论决不旨在暗示这些附图表示附图中所示的部件中的全部元件。

[0079] 图5示意性示出了图4的节点400中一个节点400的具体细节。除了示出节点管理器404之外,该图还示出了处理器320以及控制每个处理器320使用的功率的调压器(voltage regulator)500。此外,该图还示出了电源502,该电源502测量功率并将输入AC功率转换为DC输出功率以供调压器500使用。节点400还具有基板管理控制器504,该基板管理控制器504连接到系统管理以便于读取、获取和设置功率限制/范围。

[0080] 图6示出了在图1的高性能计算系统100中管理节点功率的过程。应当注意,该过程基本上简化自通常用于管理节点400所使用的功率的更长的过程。因此,该方法可以具有许多步骤,诸如本领域技术人员可能会使用的用于制定适当功率水平的具体技术。此外,一些步骤可以以与所示的不同的顺序执行或者同时执行。因此,本领域技术人员可以适当地修改该过程。

[0081] 图6的过程在步骤600开始,其中中央功率控制器401收集与其管理的网络的节点400有关的应用信息。除了其他的,中央功率控制器401可以从多个节点400上的节点管理器404获取统计信息、策略和其他相关信息。

[0082] 一些实施例可以具有多个不同的中央功率控制器400,每个中央功率控制器400被分配给HPC系统100内的特定的多个节点400(例如,同一个刀片内的节点)。然而,其他实施例可以具有管理HPC系统100内的所有节点400的单个中央功率控制器401。为了简单起见,本讨论仅涉及管理HPC系统100的多个节点400的单个中央功率控制器401。该多个节点400还包括执行应用程序402的一组节点400。这组节点400可以包括所有多个节点400,或

者一些较小数量的节点400。事实上,该组节点400可以包括执行单个应用程序402的单个节点400。在替代实施例中,中央功率控制器401简单地以默认功率为节点400供电。

[0083] 在从节点收集相关信息之后,中央功率控制器401使用内部进程来为中央功率控制器401管理的所有节点400设置规定的功率范围。换句话说,使用收集的信息,中央功率控制器401设置多个节点400的集中的功率范围。在说明性实施例中,功率范围仅具有最大功率值。因此,由中央功率控制器401管理的节点400 可以以多达最大功率值的功率运行。然而,其它实施例中的范围可以具有最小功率值和最大功率值。

[0084] 接下来,在602,中央功率控制器401集中地将节点400的功率范围设置/强制(mandate)到在步骤600确定的指定范围。说明性实施例可以诸如通过限制驻留在节点400中的处理器322的时钟速度来控制提供功率的方式数量。在其他方式中,这个新的功率范围可以针对不确定时间段设置,或设定为持续一段规定的时间,诸如直到接收另一个强制功率范围或直到某些事件发生。

[0085] 为此,中央功率控制器401向节点管理器404转发节点管理器404管理的多个节点400的功率设置消息;以这种方式,中央功率控制器401要求这些节点400 中的每一个在集中强制的规定范围内操作。节点管理器404响应地将该信息存储在存储器中,并且将功率限制为消息中指定的范围。

[0086] 因此,该步骤将多个节点400的功率限制设定为指定范围。然而,对于某些节点400的组或所有节点400来说,这样的功率范围可能不是最佳的。具体来说,运行给定应用程序402的几个节点400可能在某些时间需要更多的功率来有效率并有效地执行。在这种情况下,中央强制功率可能太低,使那几个节点400的处理器322的速度变慢的减小的时钟速度,该减小的时钟速度对于应用程序402必须完成的任务来说是不足的。如果给定的应用程序402不能使用额外的功率(例如,加速其处理器322),则它可以使在系统上执行的其他线程/应用程序402滞后,导致许多局部和系统问题。例如,这种延迟可能会导致系统抖动、局部和系统错误和一般的处理缓慢。

[0087] 为了最小化发生这些问题的可能性,说明性实施例允许给定的应用程序以与针对整个系统的集中强制的功率水平不同的功率水平进行操作。例如,给定的应用程序可以在规定时间段以比针对节点400所强制的功率级别高的功率级别操作。

[0088] 更具体地,给定的应用程序通常是关于其功率需求的可靠信息源,并且可以确定其功率需求。因此,可以最佳地确定有效率地并且有效地完成处理所需的最佳功率。因此,在从中央功率控制器401接收到中央强制功率范围之前、期间或之后,给定的应用程序402可以请求并接收针对不同功率范围的批准。

[0089] 为此,在步骤604,在确定功率需求之后,给定的应用程序402生成功率命令或请求,该命令或请求被转发给中央功率控制器401以供批准。实际上,本领域技术人员理解,这个命令可以经由多个中间功能模块在应用程序402和中央功率控制器401之间转发。因此,不要求应用程序402将功率命令直接转发到中央功率控制器401。

[0090] 该功率命令具有明确地要求中央功率控制器401改变其上正在执行的节点组 400的功率范围的指令。中央功率控制器401在步骤606响应性地确定中央功率控制器401是否可以实际上批准/允许该请求。例如,虽然增加的功率对于运行给定应用程序402的节点组400可能是有益的,但是如果被允许,可能对那些或其他节点400/应用程序402具有不可逆

转的不利影响。如果是这种情况,或者有其他拒绝请求的原因,则该过程简单地回到步骤604,并且不批准该请求。这种拒绝可能会无限的重复,也可能是针对规定数量的请求。系统管理员或其他实体可以设置用于批准请求的策略及其可以接收的请求的数量。

[0091] 相反,中央功率控制器401可以批准请求并在规定的参数下设置适当的功率。例如,中央功率控制器401可以简单地改变功率范围而没有对节点400的其他要求作出任何其它改变。其他实施例可以简单地消除功率限制或制约,允许至少一个节点400使用其所需的任何功率。

[0092] 然而,在一些实施例中,为了补偿增加的功率需求,中央功率控制器401可以允许该请求,而要求应用在某些时间段在较低功率下运行。例如,中央功率控制器401可以允许在规定时间期间的增加的功率,但要求由该组节点400使用的总功率不超过规定的总值。在这种情况下,应用程序402可以具有需要最小功率量的某些执行路径。当执行这些路径时,应用程序402可被配置为使用比最初想要使用的低的功率量。以这种方式,系统可以在满足一定时期的功率尖峰的同时保持一致的整体功率需求。

[0093] 因此,如果中央功率控制器401在步骤606批准该请求,则该过程将改变运行有请求新的功率范围的应用的节点组400上的功率范围。为此,中央功率控制器401的说明性实施例将功率改变消息转发到受影响的节点组400的节点管理器 404。在接收到消息之后,每个节点管理器404调整其内部设置以在中央功率控制器401设置的参数内操作。如上所述,这些参数可以包括功率值、时期和使用该范围的时间量。事实上,这些参数可以包括用于不同时期的多个不同功率范围。如上所述,这些不同时期可以基于本领域技术人员所规定的事件、具体时期或其它基础来设定。

[0094] 在一些实施例中,如果应用程序402跨越多个节点400,则它可以仅向中央功率控制器401作出一个请求,用于正在执行应用程序402的节点400中的一个、一些或所有中的功率改变。然而,其他实施例可以进行多个这样的请求。

[0095] 即使在应用程序402最初改变其节点组400的功率范围之后,图6的过程也可以重复以动态地改变功率范围。此外,一些实施例可以使用其他逻辑来以所描述的方式控制功率需求,而不是使应用程序402确定并请求功率需求。这样的实施例可以被认为与应用程序402协作并且有效地充当应用程序402。

[0096] 因此,说明性实施例使得给定的应用程序402能够动态地改变应用程序402 的功率需求以优化系统性能。因此,给定的应用可以具有以确保一致性、效率和总体有效性的速度进行操作的能力。

[0097] 热管理

[0098] 高性能计算系统,诸如上述的一个高性能计算系统,消耗大量功率,并因此相应产生大量的热量。因此,本领域技术人员已经开发出许多用于冷却系统的生热部件(例如,处理器320、存储器326等)的不同技术。一种技术使用包含液体冷却剂(例如水)的闭环通道,该液体冷却剂从生热元件吸收热并通过环境的较冷部分(例如包含计算机系统的建筑物的外部)循环,。

[0099] 冷却器环境简单地可以引导携带液体冷却剂的盘管(coil)通过在盘管上吹出较冷的空气的风扇/鼓风机,从而产生对流冷却效果。不理想的,这种对流冷却系统(称为“干式冷却系统”)在较暖的气候条件下、甚至在较冷的气候条件下几天内可能不能提供足够的

冷却。为了解决这个问题,一些系统使用“湿式冷却系统”,除了吹送空气到盘管上之外,通常将水喷射到盘管上以增强冷却。因此,这些系统通常被称为“蒸发冷却系统”。为了获得两个系统的优点,一些高性能计算系统采用混合方式,在需要时,“混合冷却器”可以在干式冷却模式(即,如干式冷却系统)或湿式冷却模式(即,如湿式冷却系统)下运行。例如,大多数时候,混合式冷却器可以在干式冷却模式下工作,但是在某些异常热的日子里可以切换到湿式冷却模式。

[0100] 然而,世界上许多地区处于干旱状况或缺水严重。许多人预测,缺水将成为 21 世纪的主要问题。因此,示例性实施例控制高性能计算系统的功耗以最小化使用更有效的冷却过程(例如,湿式冷却过程)的需要。为此,高性能计算系统内的逻辑优选地确定系统的热分布(thermal profile),并在规定时期暂时减慢处理器速度,使得高性能计算系统不需要湿式冷却过程。虽然预计整体系统速度将会降低,但可以减少冷却系统的用水需求。

[0101] 例如,当在环境中检测到的温度或在高性能计算系统的某一部分中检测到的温度达到或超过X摄氏度时,混合冷却系统可以开始湿式冷却模式。因此,中央功率控制器401可以管理节点400、刀片或整个系统内的处理器320的速度,试图要确保所述温度不超过X摄氏度。具体地,降低处理器320的速度通常减少处理器320产生的热。换句话说,向处理器320供给较少功率,导致处理器320产生较少的热。

[0102] 在实施这些目标的其他方式中,中央功率控制器401(或其他系统逻辑)可以递增地降低处理器速度(例如,以1%的增量或不均匀增量降低处理器速度),并且周期性地检查使用闭环过程的一些系统相关区域的温度。其他实施例可以简单地对功率控制器400编程以在规定时间期间将处理器速度降低到规定的较低速度。为了有效地操作,这些后面的实施例可能需要系统的行为的一些预测性知识和预期温度。如上所述,处理器速度可以在节点级别(例如,在给定节点400中的一个或多个处理器320内)、刀片级别、机架级别、系统级别或整个数据中心级别下控制。

[0103] 实际上,本领域技术人员可以应用于管理功率的其它技术。例如,在炎热的日子中,系统的某些或全部可能受到严格的功率限制/限制,从而以所述方式进行处理器速度管理。例如,在节点400上执行的高优先级应用程序402可能需要速度的“突发(burst)”,并因此可以被允许超过如上述部分所述的某些限制。当发生这种情况时,中央电源控制器401应确保总功率预算不接近或超过临界极限。

[0104] 换句话说,当高优先级应用程序402以较高速度运行时,总的系统功率仍应低于功率限制。如果不是,则高优先级应用程序402的例外不应该被允许,即高优先级应用程序402也将以较慢的速率执行。因此,功率控制器400可以从平均功率使用设置具有一定量的净空的功率限制,以允许高优先级应用402的较高速度。可替代地或另外,功率控制器400可以通过降低在系统的某些其他部分处的处理器320的速度甚至进一步低于功率限制来补偿高优先级应用402所需的增加的功率。

[0105] 因此,示例性实施例可以消除对混合冷却系统的需要。事实上,说明性实施例也可以消除对昂贵的冷却装置(chiller)的需要,冷却装置将冷却盘管通过制冷剂,以在热环境中更加有效地冷却系统。例如,一些数据中心既有湿式冷却系统,又有一些冗余的冷却装置,以确保发热问题不会对系统性能产生不利影响。

[0106] 然而,许多这样的系统每年只有几天需要冷却装置的能力。其余的时间,湿式冷却

系统或甚至是干式冷却系统就足够了。尽管如此,没有使用本发明的各种实施例,这种系统的操作者通过获取冷却装置不必要地承受增加的资本成本。

[0107] 事实上,许多其他高性能计算系统通常都将冷却装置作为唯一的冷却来源。虽然只有一个冷却源可能降低资本成本,但这些系统具有比干式或湿式冷却系统更高的增量能源/运营成本。因此,选择性地降低处理器速度允许使用较便宜的冷却源,并且消除对一些冗余的备用冷却系统的需要。

[0108] 本领域技术人员可以使用多个过程和设备实施该系统热管理。为此,图7示意性地示出了包含高性能计算系统及伴随高性能计算系统的热管理设备的空间 700。应当注意,该图可以被认为是查看图1至图4所示的整个系统的另一种方式。

[0109] 图7的每个部件通过任何常规的互连机构可操作地连接。图7简单地示出了联络每个组件的总线702。本领域技术人员应当理解,可以修改该广义表示以包括其他常规的直接连接或间接连接。因此,对总线702的讨论并不旨在限制各种实施例。

[0110] 实际上,应当注意,图7仅示意性示出了每个下面讨论的每个部件。本领域技术人员应当理解,可以以各种常规方式来实施这些部件中的每一个,诸如通过使用横跨一个或多个其他功能部件的硬件、软件或硬件和软件的组合来实施。例如,中央功率控制器401可以使用执行固件的多个微处理器来实施。作为另一个示例,中央功率控制器401可以使用一个或多个专用集成电路(即,“ASIC”)和相关软件,或ASIC、分立电子部件(例如,晶体管)和微处理器的组合来实施。因此,图7的单个框中的中央功率控制器401和其他部件的表示仅仅是为了简单的目的。事实上,在一些实施例中,中央功率控制器401横跨多个不同的设备(未必在相同的外壳或机箱内)而分布。

[0111] 应该重申的是,图7的表示是高性能计算系统部件的显著简化表示。本领域技术人员应当理解,系统和所示子系统具有不在该图中(但是可能在其它附图中显示)的许多其它物理和功能部件,例如中央处理单元320和RAM 316。因此,这个讨论决不是暗示图7表示在高性能计算系统中使用的装置的所有元件。

[0112] 具体来说,图7的系统包括与上述冷却系统(在图7中使用附图标记“704”标识)物理和逻辑地连接的上述高性能计算系统100。因此,如上所述,冷却系统 704具有填充有制冷剂(诸如水)的一个或多个盘管706,该一个或多个盘管706 从高性能计算系统的发热元件抽取能量。冷却系统704可以通过诸如上面讨论的那些方式中的任何一种来对这些盘管706进行外部冷却。因此,冷却系统704可以具有上述干式冷却模式,一种或额外的多种湿式冷却模式(例如,用水喷射盘管706的一种模式)和/或使用冷却装置的另一模式。

[0113] 在说明性实施例中,热管理系统708控制并协调冷却系统704和高性能计算系统,以根据由高性能计算系统提取的功率来优化冷却。例如,如上所述,热管理系统708可以降低作为一些环境温度的函数的高性能计算系统的处理器速度和/ 或改变冷却系统704的模式。

[0114] 为此,热管理系统708具有确定某些环境条件(下面将讨论)的温度的温度传感器710以及与节点管理器404交互以根据所确定的温度控制处理器速度的处理器控制器712。在该示例中,处理器控制器712被实施为中央功率控制器401的一部分(图4)。然而,其他实施例可以独立于中央功率控制器401来实施处理器控制器712。

[0115] 图7还示出了错误管理系统714,该错误管理系统714包括中央功率控制器 401、处

理器控制器712和检测某些规定的错误条件的错误检查器716。以类似于热管理系统708的方式,并且在下面的“系统鲁棒性”部分中更详细地讨论,错误管理系统714根据检测到的可校正错误控制处理器速度。

[0116] 空间700可以是通常容纳高性能计算系统的任何常规建筑物或大厦。例如,空间700可以是管理各种商业、科学和/或工业实体的高性能计算系统的气候控制数据中心。尽管未示出,盘管706可以在空间700的外部延伸到另一空间,或者在包含空间700的建筑物的外部。

[0117] 图8示出了管理根据本发明的说明性实施例的管理高性能计算系统的热分布的过程。以类似于图6的方式,应当注意,该过程基本上简化自通常用于管理节点400所使用的功率的较长的过程。因此,该方法可以具有许多步骤,诸如本领域技术人员可能会使用的用于制定适当功率水平的具体技术。此外,一些步骤可以以与所示的不同的顺序执行或者同时执行。因此,本领域技术人员可以适当地修改该过程。

[0118] 该过程从步骤800开始,其中温度传感器710监测与高性能计算系统有关的某些环境或设备的温度。为此,温度传感器710可以与节点管理器404协作,独立于节点管理器404获取温度数据,或者完全依赖于来自节点管理器404的数据。在一些实施例中,步骤800监视容纳高性能计算系统的空间700的温度。例如,可以包括在不同位置处的多个分开的温度传感器710的温度传感器710可以位于高性能计算系统附近。在这种情况下,温度传感器710检测空间700的某些部分的空气温度。本领域技术人员可以基于许多因素将传感器定位在适当的位置,诸如与高性能计算系统的发热部件的接近性。可替代地或另外,传感器可以检测空间700外的环境条件,例如在包含空间700的建筑物的外部的环境中进行检测。

[0119] 其他实施例可以将传感器定位在高性能计算系统的特定部件的附近或特定部件处。除其他之外,传感器可以被定位为感测处理器320、存储器(例如,RAM卡326和RAM芯片326)、集线器ASIC 340和电源502中的一些或全部的操作温度。

[0120] 中央功率控制器401接收这些温度读数中的一些或全部,然后确定是否应改变处理器320的功耗;即,是否应该改变处理器320的处理速度。因此,该过程继续到步骤802,其中中央功率控制器401确定检测到的温度是否单独或一起满足规定的高温度要求或低温度要求。

[0121] 具体来说,中央功率控制器401使用算法来确定是否应该改变处理速度。除了其他方法之外,这些算法可以使用人工智能、实验、经验方法和其他类似技术来开发。在简单的情况下,中央功率控制器401可以接收单个温度值(例如,空间700的区域中的温度、或电源502中的一个的温度),并将该单个温度值与规定的上阈值和/或下阈值比较。如果该单个温度值大于规定的上阈值,则功率控制器400可以降低功耗。然而,其他实施例可以使用来自多个不同传感器的温度读数(例如环境温度和部件温度两者)来确定中央功率控制器401是否应采取行动来增加或减少功率/处理速度。

[0122] 例如,中央功率控制器401可从空间700的15个不同部件和/或不同区域(诸如处理器320、电源502和存储器326)接收温度值。内部算法可以将比例因子和权重应用于不同的温度,以确定总体温度改变是否保证功耗的改变。类似的示例可以从空间700的相同的15个不同部件和/或不同区域接收温度值,并且如果这些温度值的一些规定的数量超过所述规定的上阈值和下阈值限定的范围,则改变功率消耗。这两个例子也可以具有暂时部件,该暂

时部件在采取行动之前以规定次数要求超出所述范围的温度。因此,这种时间方法可以避免短期正或负温度尖峰的系统延迟、抖动和其他不利影响。

[0123] 因此,继续步骤802,中央功率控制器401确定温度(即,实际温度值或作为从温度传感器710接收的温度值的函数产生的值)是否在以上所讨论的规定的温度范围内。应当注意,一些实施例不具有较低的温度范围。如果实施例具有较低的温度阈值,并且温度低于该低温阈值,则该过程继续到步骤804,在步骤804确定是否可以增加处理器速度。存在许多不增加处理器速度的原因。处理器320可能以由中央功率控制器401设置的最大速度运行,或者简单地处理器320可能不能更快地运行,以及其他原因。

[0124] 如果处理器320不能增加速度,则该过程回到监视相关温度(步骤800)。然而,如果处理器320可以提高速度,则处理器控制器712将处理器速度提高一些量(步骤806)。除了其他方法,处理器控制器712可以将处理器320的速度提高预定百分比、提高一绝对量、提高至一提高的速度,或根据高性能计算系统100的其他过程和要求提高处理器320的速度。

[0125] 除了其他方式,处理器控制器712可以无限地增加速度长达规定的时间量,或者基于一些其它标准无限地增加速度。在规定的的时间量之后,处理器控制器712可以将速度改变为另一个值,诸如返回到处理器320执行的一个或多个较早的速度,或由处理器控制器712指定的另一个速度。然而,所选择的时间量可以被中断,因此,如果某些其他事件需要处理速度的不同改变(例如,当重复地执行图8的过程时,后续的温度改变需要处理速度的不同改变),则所选择的时间量可以改变。

[0126] 返回到步骤802,如果温度在该范围内,则该过程简单地回到步骤800以监测相关温度。最后,如果步骤802确定温度超过高温阈值,则过程继续到步骤808,在步骤808确定处理器速度是否可以减小。尽管温度升高,但还是有不降低速度的原因。例如,如上所述,处理器速度的降低可能不利地影响应用402或其他系统功能的执行。如果中央功率控制器401确定处理器速度能够降低,则处理器控制器712将减速消息转发到适当的节点400,降低处理器速度(步骤810)。在其他方式中,处理器控制器712可以将处理器320的速度降低指定百分比、降低一绝对量、降低至一减小的速度,或者根据高性能计算系统的其他过程和要求降低速度。应当注意,中央功率控制器400可以在高性能计算系统100的任何方便的部分中降低功率。

[0127] 在类似于上面关于步骤806所讨论的方式中,处理器控制器712可以无限地降低速度长达规定的时间量,或者基于一些其它标准无限地降低速度。在规定的的时间量之后,处理器控制器712可以将速度改变为另一个值,诸如返回到处理器320执行的一个或多个较早的速度,或由处理器控制器712指定的另一个速度。然而,所选择的时间量可以被中断,因此,如果某些其他事件需要处理速度的不同改变(例如,当重复地执行图8的过程时,后续的温度改变需要处理速度的不同改变),则所选择的时间量可以改变。

[0128] 处理器控制器712还可以降低处理速度,直到所监视的温度(或各种不同区域的多温度功能和/或超过规定的时间帧)降低到规定的温度。例如,处理器控制器712可以使用规定的温度值和一定的公差,之后在使用干式冷却模式时、或者使用较少的水、或使用能量密集型湿式冷却模式时,提高处理器速度。

[0129] 可以基于多种不同的技术来确定被选择以降低(或增加)时钟速度的时间量。其中,除了其他之外,本领域技术人员可以使用人工智能、当前系统数据和历史系统数据、以

及经验实验数据。

[0130] 在步骤810降低处理速度之后,该过程回到步骤800,以继续监视相关温度。然而,如果步骤808确定处理器速度不能减小,则该过程继续到步骤812,步骤 812开始下一个冷却模式。例如,该过程可以将冷却系统704的模式从诸如干式冷却模式的较低模式改变为诸如湿式冷却模式的水密集型冷却模式。一些实施例可以跳过该步骤,并且简单地向系统操作者发送警告或通知消息。

[0131] 尽管处理速度减小,但也可能存在所监视的温度继续上升的情况。在这种情况下,在响应于所监视的温度将处理速度减小到较低速度之后,中央功率控制器 401可以将处理速度保持在减小的量并且将冷却系统704切换到更高的模式,诸如从干式冷却模式到湿式冷却模式,或者甚至冷却装置。对于冷却系统704处于湿式冷却模式的时间的至少一部分,处理速度可以继续不大于减小的值。然而,随着盘管706冷却,处理速度可以被提高,或者冷却模式可以切换回干式冷却模式。

[0132] 一些实施例可以根据冷却模式中的一个的已知冷却能力来抢先地降低处理速度。例如,中央功率控制器401可以确定诸如干式冷却模式的较低冷却模式的冷却能力在某些时间需要一规定的最大处理速度。如果处理速度较高,则该较低的冷却模式可能无法充分地冷却系统100。可以根据多个变量作出该确定,该多个变量包括当日时间、高性能计算系统的历史性能、执行应用程序502等。因此,中央功率控制器401可以抢先地降低处理速度以维持冷却系统704和一个或多个更低、更有效的冷却模式。

[0133] 中央功率控制器401还可以监视冷却系统704的运行状况,并且如果检测到冷却系统704故障,则采取适当的动作。例如,在检测到冷却系统704中的诸如完整的冷却系统失效的故障之后,中央功率控制器401可以将处理速度降低到能够容忍失去冷却功能的水平、和/或关闭冷却系统704。当然,冷却系统的故障未必会去除完整的冷却功能。一些故障可能会去除某些冷却模式,或降低某些模式的效率。例如,如果中央功率控制器401检测到湿式冷却模式不起作用,则其处理器控制器712可以抢先地降低处理速度以适应更有效率但效果稍差的干式冷却模式。作为另一示例,如果中央功率控制器401检测到干式冷却模式不工作,但是湿式模式正常工作,则其处理器控制器712可以增加处理速度以利用湿式冷却模式的增加的冷却能力。

[0134] 以类似的方式,中央功率控制器401的一些实施例可以具有用于检测冷却系统704中即将到来的失效的工具。例如,冷却系统704可以以指示即将发生的失效或故障的特定方式运转。在这种情况下,中央功率控制器401可以抢先地降低处理速度以避免潜在的灾难性的系统性崩溃。

[0135] 应当注意,虽然在节点级别上讨论了处理速度改变,但是本领域技术人员可以选择性地改变在单个节点400上的处理器320的子组或横跨节点400的特定处理器320的子组之中的处理速度。也像上面讨论的其它实施例,处理速度改变可以横跨其他级别作出,诸如横跨刀片级别、机架级别、系统级别或整个数据中心级。

[0136] 因此,示例性实施例允许用户使用较便宜的冷却替代方案。然而,当需求出现时(例如,外部天气条件超出某些限制),则系统可以切换到在冷却能力中的下一个“升级”。例如,当需要时,系统可以从1)正常干式冷却转换到下一个升级,这可能是湿式冷却,或2)正常湿式冷却切换到冷却装置(即下一个升级)。除了其他优点,用户因此可以获取下一个升

级的冷却系统704作为当天气条件超出限制时的备份(即,用户可以在限制的情况下切换到下一级别的系统以节省水和/或能量)。可替代地,用户可以仅使用单级冷却系统704(即,不获取下一个升级的系统作为备份),从而节省资本成本。

[0137] 系统鲁棒性

[0138] 与其他计算机系统类似的方式,高性能计算系统将会面临失效,诸如在运行时发生不可校正的错误。然而,在运行时间期间发生的许多错误是可校正的,因此不会立即导致系统失效。例如,除其他职位,这些可校正的错误可能包括某些类型的存储器错误、处理器错误或网络错误。

[0139] 发明人认识到,系统失效通常某些条件之后,诸如当系统经历相对较多数量的可校正错误时。发明人还认识到,当以高速运行高性能计算系统的处理器320时,高性能计算系统可能更容易出现这种错误。

[0140] 为了解决这个问题,发明人发现,当系统上的逻辑检测到某些条件时,它们可以降低在高性能计算系统上运行的处理器320中一些或全部的速度。如上所述,这些条件可以包括在一定时间内检测到规定数量的可校正错误或者规定类型的可校正错误。当然,本领域技术人员可以在检测到其他条件降低处理器速度。因此,对某些数量和类型的可校正错误的讨论是检测潜在系统问题的一种方式示例,因此不旨在限制某些实施例。

[0141] 因此,当说明性实施例检测到所规定的条件时,逻辑可以降低系统100中的一个或多个处理器320的处理速度。像其他实施例一样,处理器速度可以在节点级别(例如,在给定节点400中的一个或多个处理器320内)、刀片级别、机架级别、系统级别或整个数据中心级别下被控制。本领域技术人员可以选择多种方法中的任何一种来减慢处理速度。例如,逻辑可以使系统逐渐减小到当前可校正错误的数量低于规定阈值的程度。然而,作为第二示例,逻辑可以将系统降低到规定速度(例如,正常速度的10%或15%),直到条件满足某些正常要求。

[0142] 一些实施例可以减小速度,直到某些线程或进程完成其执行。例如,进程可能已经执行了两个星期,需要另外一天才能完成处理(或者另一天来到达适当的停止点)。如果系统减速百分之十,那么该进程可能需要另外1.1天才能完成处理。在这种较慢的速度下,系统应该受到较少的压力,因此更稳定,降低系统在完成此过程之前失效的可能性。然而,如果系统不减速,那么系统更有可能失效,不理想地失去两个星期的处理。因此,将处理时间增加如此少的量(相对于总处理时间)可以产生实质的益处。

[0143] 在上述示例中的处理已经完成执行之后,或者系统已经达到某些其他规定的标准之后,系统的一些或全部可以被关闭,以便技术人员可以修复错误的来源。可替代地,为了修复错误源,一些实施例可以热插拔系统的特定部分,而不是关闭系统。

[0144] 为此,图9示出了根据本发明的说明性实施例的管理高性能计算系统100的错误的过程。该过程优选地由错误管理系统714执行。以类似于图6和8的方式,该过程显著地简化自通常被用于管理高性能计算系统100的错误的较长的过程。因此,该方法可以具有许多步骤,诸如本领域技术人员可能会使用的用于制定适当功率水平的具体技术。此外,一些步骤可以以与所示的不同的顺序执行或者同时执行。因此,本领域技术人员可以适当地修改该过程。

[0145] 该方法在步骤900开始,其中错误检查器716针对多个可校正错误条件中的何一个

监视高性能计算系统100的部分或全部。如上所述,这些错误可以包括各种规定的或未规定的可校正错误条件中的任一个。例如,这些错误条件可以包括通过不正确的校验和以及散列函数检测到的存储器可校正错误和网络可校正错误。然而,一些实施例可以间接地检测错误条件的证据,而不是直接检测错误条件。例如,诸如处理器320、电源502或存储器芯片326的某些系统部件的升高的温度可以指示出错误条件的高可能性。

[0146] 不管检测错误的模式如何,中央功率控制器401具有所规定的、用于确定可校正错误是否成为处理器速度减小的依据的规则和/或算法的集合。除了其他方式,这些算法可以使用人工智能、实验、经验方法、当前的系统性能和其他类似技术来开发。例如,中央功率控制器401可以以加权的比例对各种错误条件进行评分,将加权值相加,并且确定这些权重之和是否超过规定的阈值。作为第二示例,中央功率控制器401可以具有特定的错误条件,无论加权或算法如何,都需要降低处理器速度。作为另一个示例,中央功率控制器401可以简单地基于整个系统100中或在系统100的特定部分上的可校正错误的总数来减小速度。

[0147] 因此,在检测到可校正的错误条件之后,中央功率控制器401在步骤902确定是否可以减小处理器速度。例如,关键应用402可以在所讨论的节点400上执行,并且不能被干扰。在这种情况下,该过程可以回到步骤900,以针对错误可校正条件继续监视。然而,中央功率控制器401在那时可以采取额外的动作,以确保系统的操作。例如,中央功率控制器401可以选择性地降低至其他处理器320的功率,和/或向系统用户发送指示潜在问题的警告消息。

[0148] 如果中央功率控制器401确定可以减小处理器速度,则该过程继续到步骤904,步骤904以各种方式中的任一种降低了处理器速度。除了其他方式,处理器控制器712可以将处理器320的速度降低预定百分比、降低绝对量、降低至减小的速度,或根据高性能计算系统100的其他过程和要求降低处理器320的速度。

[0149] 以类似于上面关于图8所讨论的方式,处理器控制器712可以无限地降低速度,长达或持续预定时间,或者基于一些其它标准无限地降低速度。在规定时间之后,处理器控制器712可以将速度改变为另一个值,例如返回到处理器320执行的一个或多个较早的速度,或由处理器控制器712指定的另一个速度。然而,所选择的时间量可以被中断,因此,如果某些其他事件需要处理速度的不同改变(例如,当重复地执行图8的过程时,后续的温度改变需要处理速度的不同改变),则所选择的时间量可以改变。

[0150] 处理器控制器712还可以降低速度,直到所监视的温度降低到规定的温度。可以基于多种不同的技术来确定被选择以用于减少处理速度的时间量。其中,本领域技术人员可以使用人工智能,当前系统数据和历史系统数据、以及经验实验数据。

[0151] 重要的是,处理速度在一段时间内以最佳地确保系统有效性的速度降低,从而降低了系统失效的可能性。

[0152] 在减小速度之后,过程继续到步骤906,其中逻辑或操作者校正所定位的错误。除了其他方式,系统可以(选择的)采用以降低的速度执行的处理器320来运行,直到技术人员可以关闭系统并修理或替换潜在损坏或损坏的部件。可替代地,系统可以采用具有降低的速率的上述处理器320来运行,直到技术人员可以通过热插拔或类似技术来修复错误。例如,技术人员可以热插拔具有缺陷部件的某些板,新的板具有未损坏的相似部件。

[0153] 在步骤908,通过将处理速度恢复到正常状态来完成该过程。

[0154] 因此,以这种方式降低微处理器速度为系统提供了更好机会以执行到后面的时刻,在该后面的时刻能够在没有大量的使用中断的情况下采取补救措施。

[0155] 本发明的各种实施例可以至少部分地在任何传统的计算机编程语言中实施。例如,一些实施例可以以过程编程语言(例如,“C”)或面向对象的编程语言(例如,“C++”)来实施。本发明的其他实施例可以被实施为预先配置的备用硬件元件和/或作为预编程的硬件元件(例如,专用集成电路、FPGA和数字信号处理器)或其它相关部件。

[0156] 在替代实施例中,所公开的装置和方法(例如,参见上面描述的各种流程图)可以被实施为用于与计算机系统一起使用的计算机程序产品。这样的实施方式可以包括固定在诸如计算机可读介质(例如,软盘、CD-ROM、ROM或固定盘)的有形的非暂时介质上的一系列计算机指令。一系列计算机指令可以体现本文前面关于系统所描述的全部或部分功能。

[0157] 本领域技术人员应该理解,这样的计算机指令可以用许多编程语言编写,以与许多计算机体系结构或操作系统一起使用。此外,这样的指令可以存储在诸如半导体、磁性、光学或其他存储器件的任何存储设备中,并且可以使用诸如光学、红外、微波或其它传输技术的任何通信技术来传输。

[0158] 除了其他方式,这样的计算机程序产品可以被发布为附带有印刷或电子文档的可移动介质(例如,紧缩套装软件(shrink wrapped software)),使用计算机系统预下载(例如,在系统ROM或固定盘上),或从网络上的服务器或电子公告板发布(例如,互联网或万维网)。事实上,一些实施例可以在软件即服务模型(“SAAS”)或云计算模型中实施。当然,本发明的一些实施例可以被实施为软件(例如,计算机程序产品)和硬件的组合。本发明的其它实施例被实施为完全是硬件或完全是软件。

[0159] 例如,说明性实施例可以包括用于在计算机系统上使用的用于热管理高性能计算系统的计算机程序产品,计算机程序产品包括其上具有计算机可读程序代码的有形非瞬态计算机可用介质,该计算机可读介质程序代码包括:

[0160] 用于监测以下至少一个的温度的程序代码:a)高性能计算系统的环境和b)高性能计算系统的至少一部分,高性能计算系统具有多个具有微处理器的节点;以及

[0161] 用于根据所监视的温度中至少一个来控制多个节点中的至少一个上的至少一个微处理器的处理速度。

[0162] 另一个示例可以涉及一种在用于管理高性能计算系统的计算机系统上使用的计算机程序产品,该计算机程序产品包括其上具有计算机可读程序代码的有形非瞬态计算机可用介质,该计算机可读程序代码包括:

[0163] 用于检测高性能计算系统的规定错误条件的程序代码,高性能计算系统具有多个具有微处理器的节点;以及

[0164] 用于在检测到规定错误条件之后降低多个节点中的至少一个上的微处理器中的至少一个微处理器的处理速度。

[0165] 虽然上述讨论公开了本发明的各种示例性实施例,但是应当显而易见的是,本领域技术人员可以进行各种修改,以达到本发明的一些优点而不脱离本发明的真实范围。

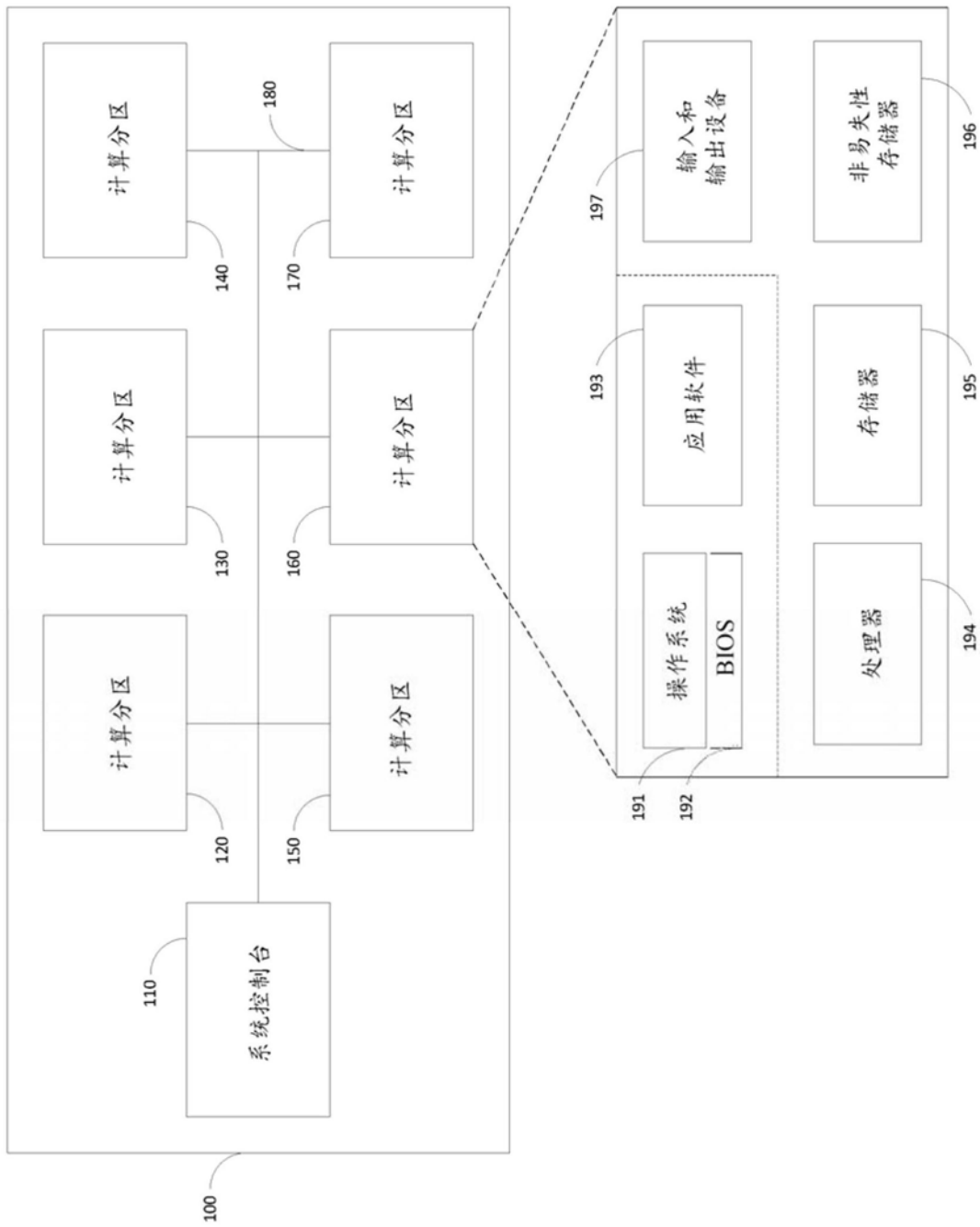


图1

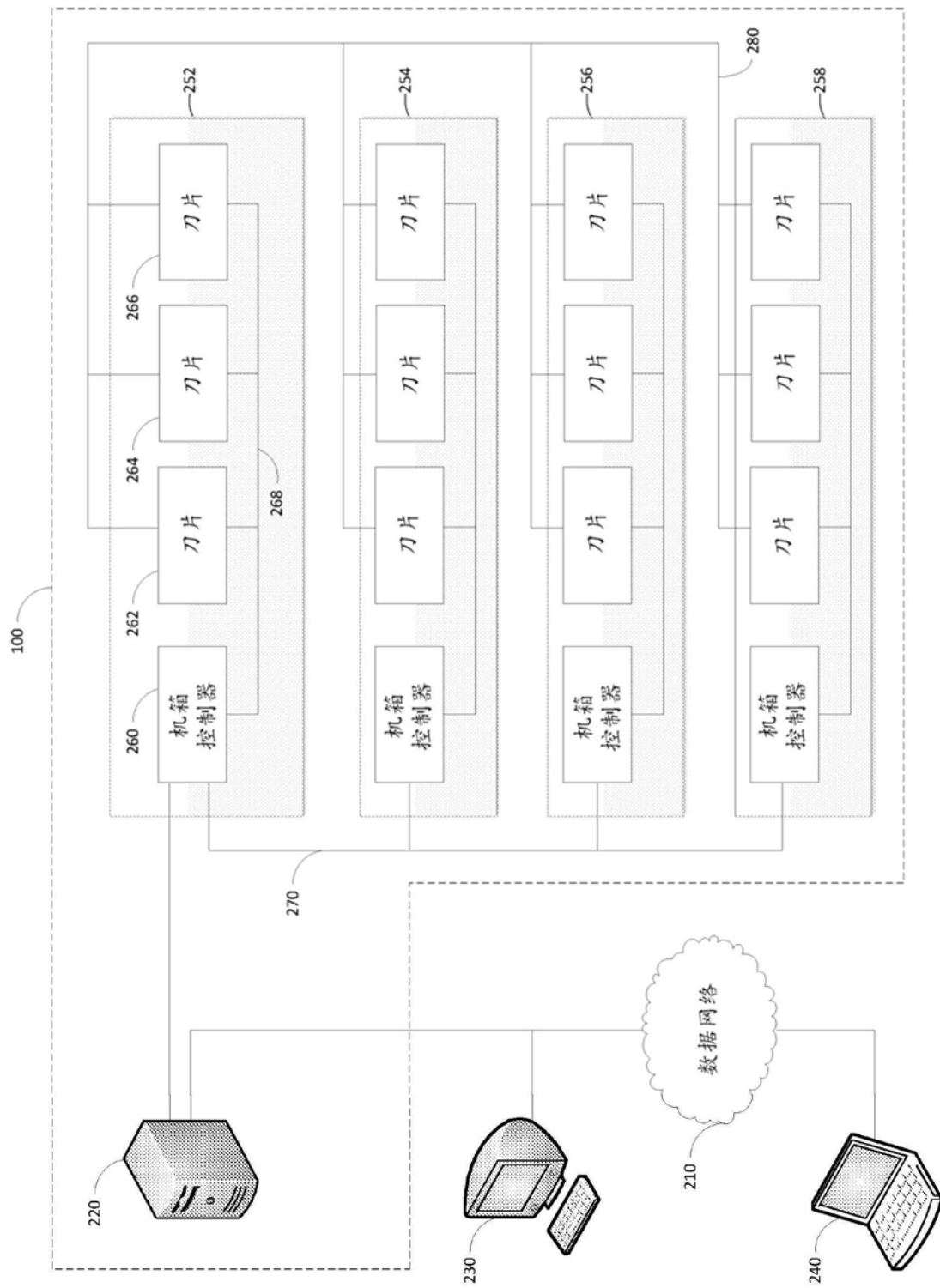


图2

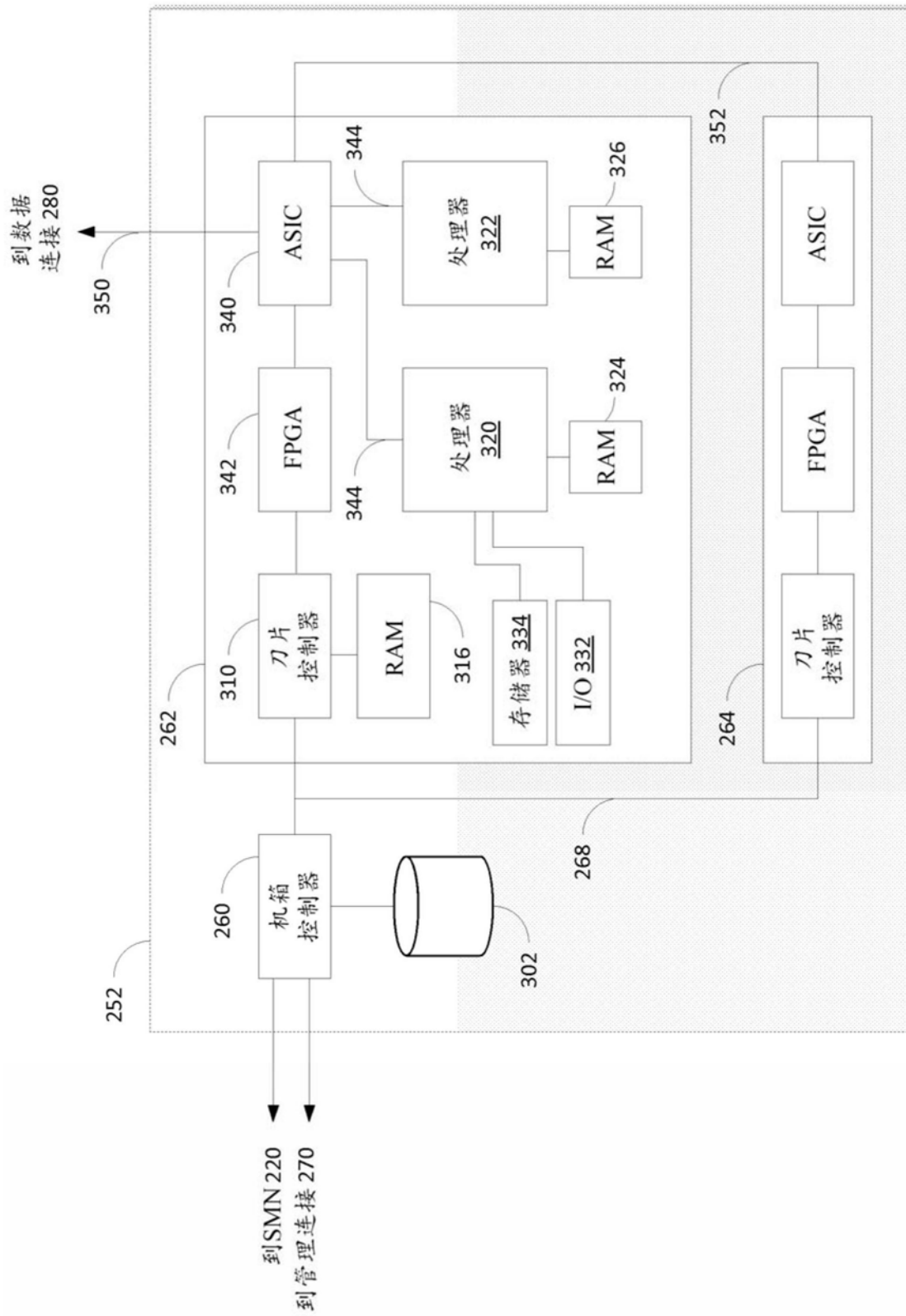


图3

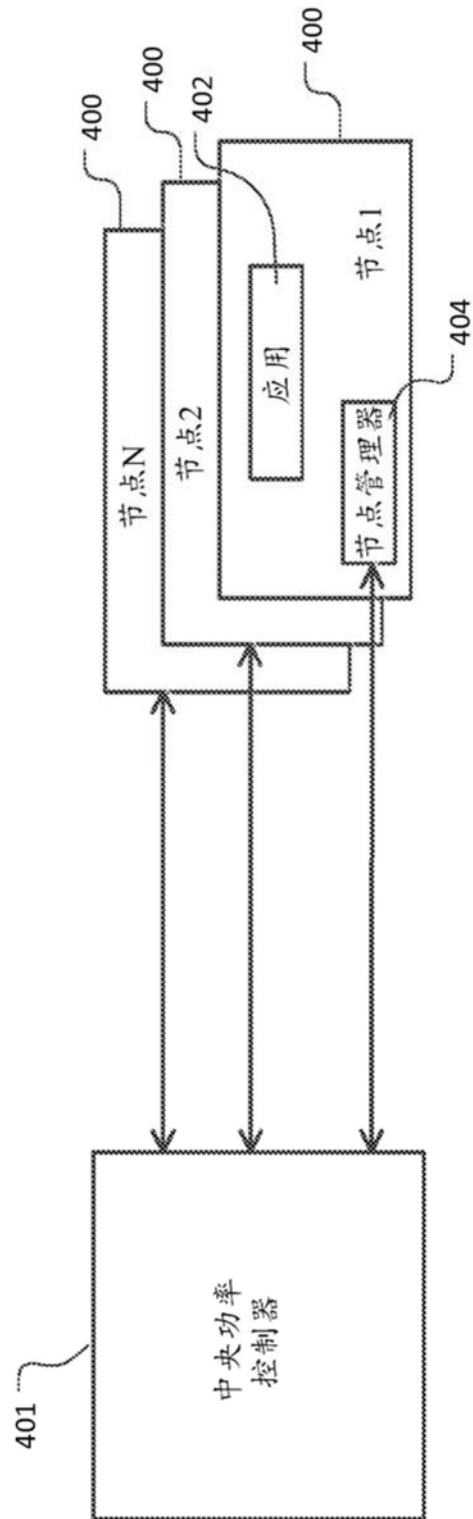


图4

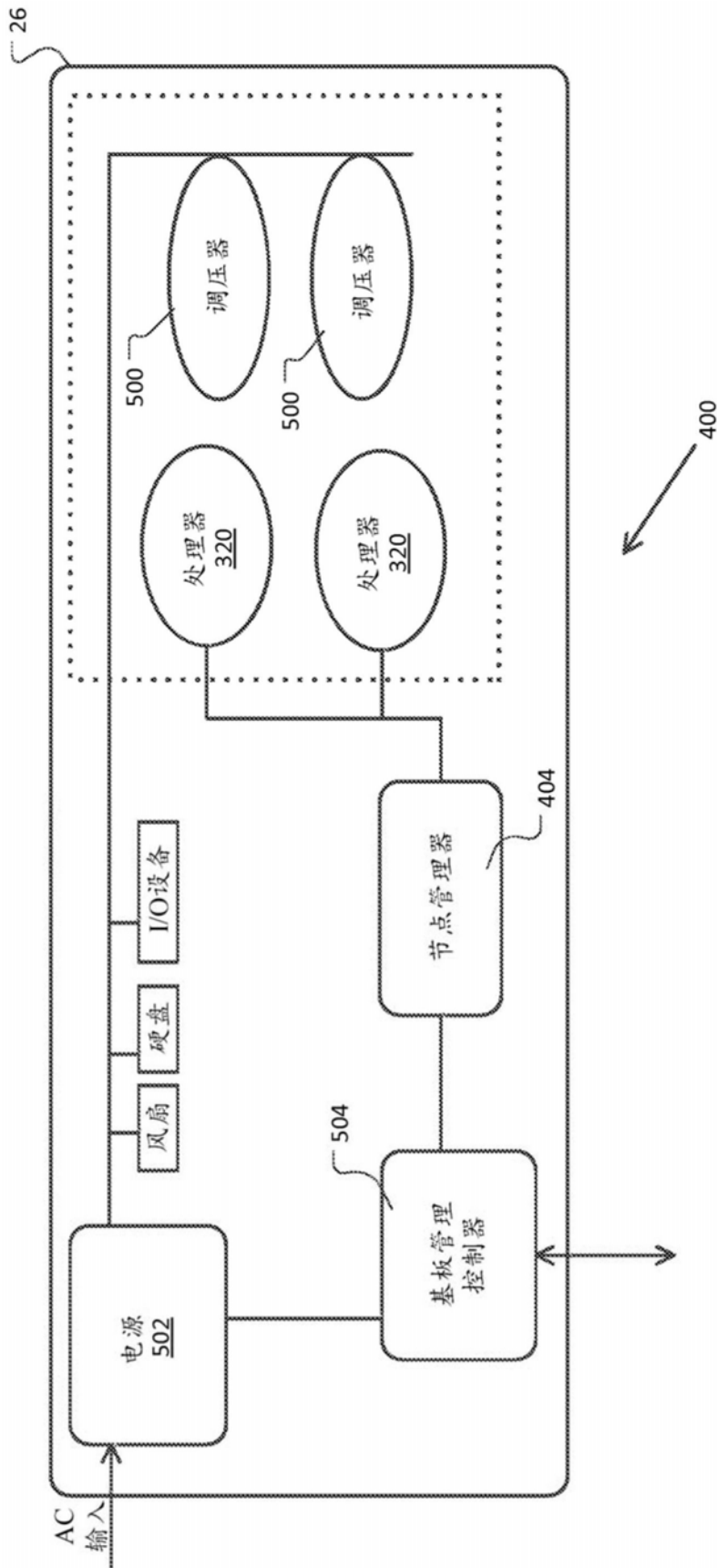


图5

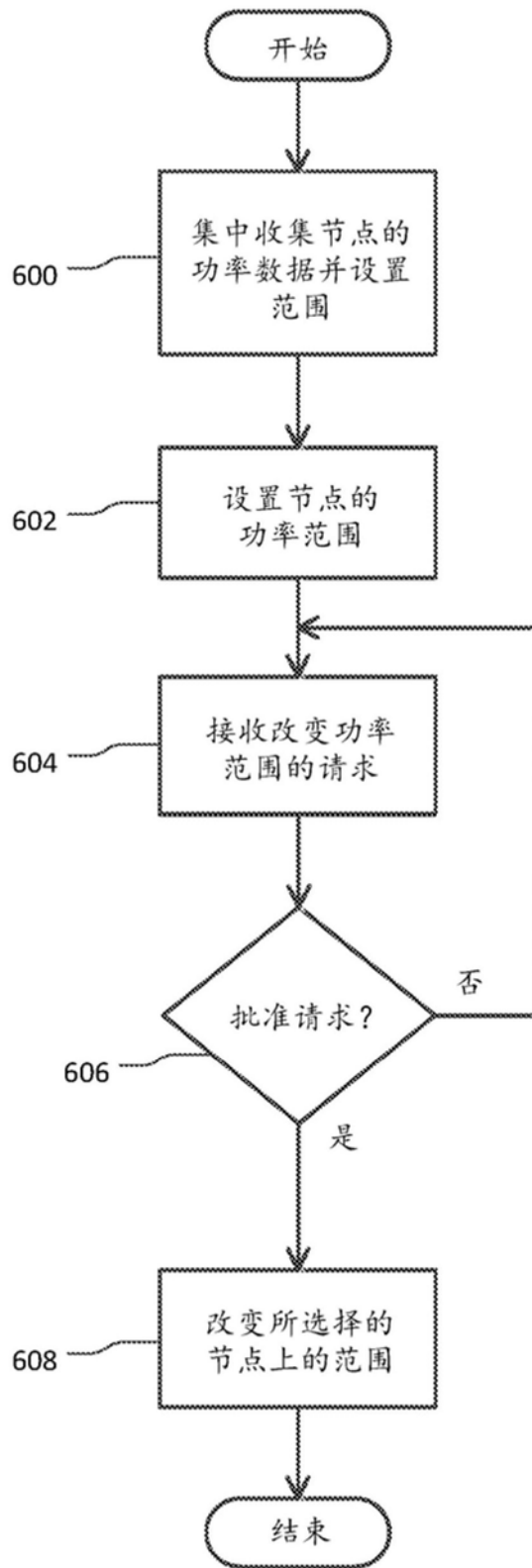


图6

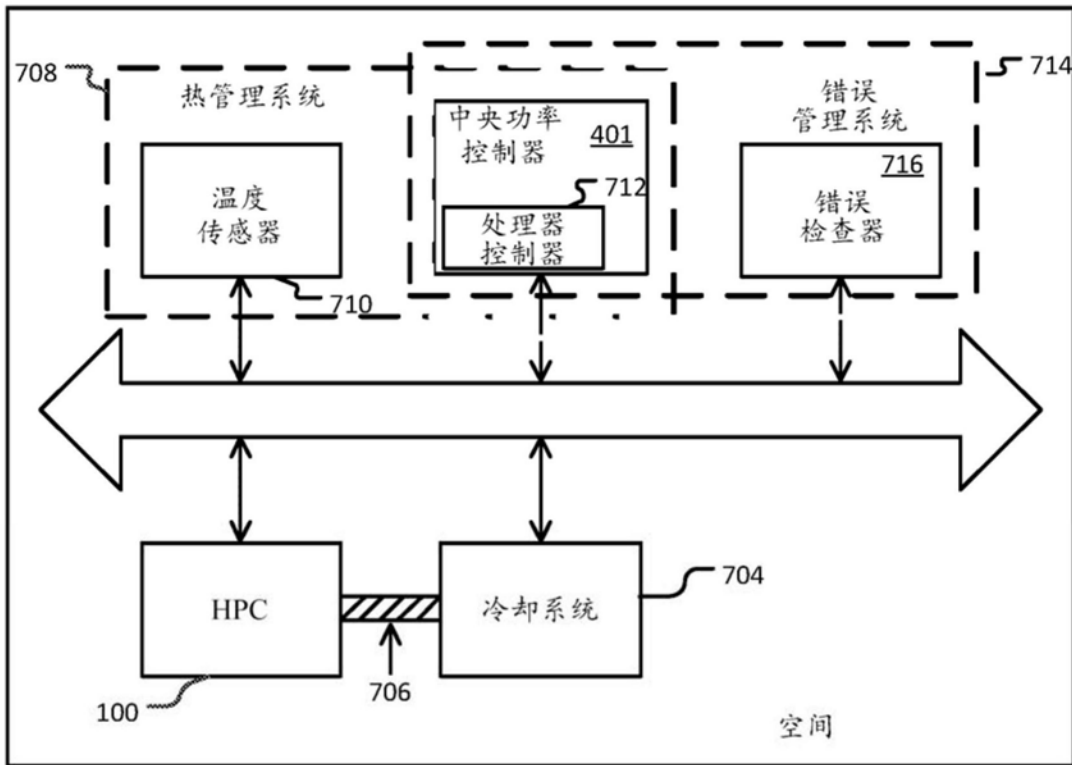


图7

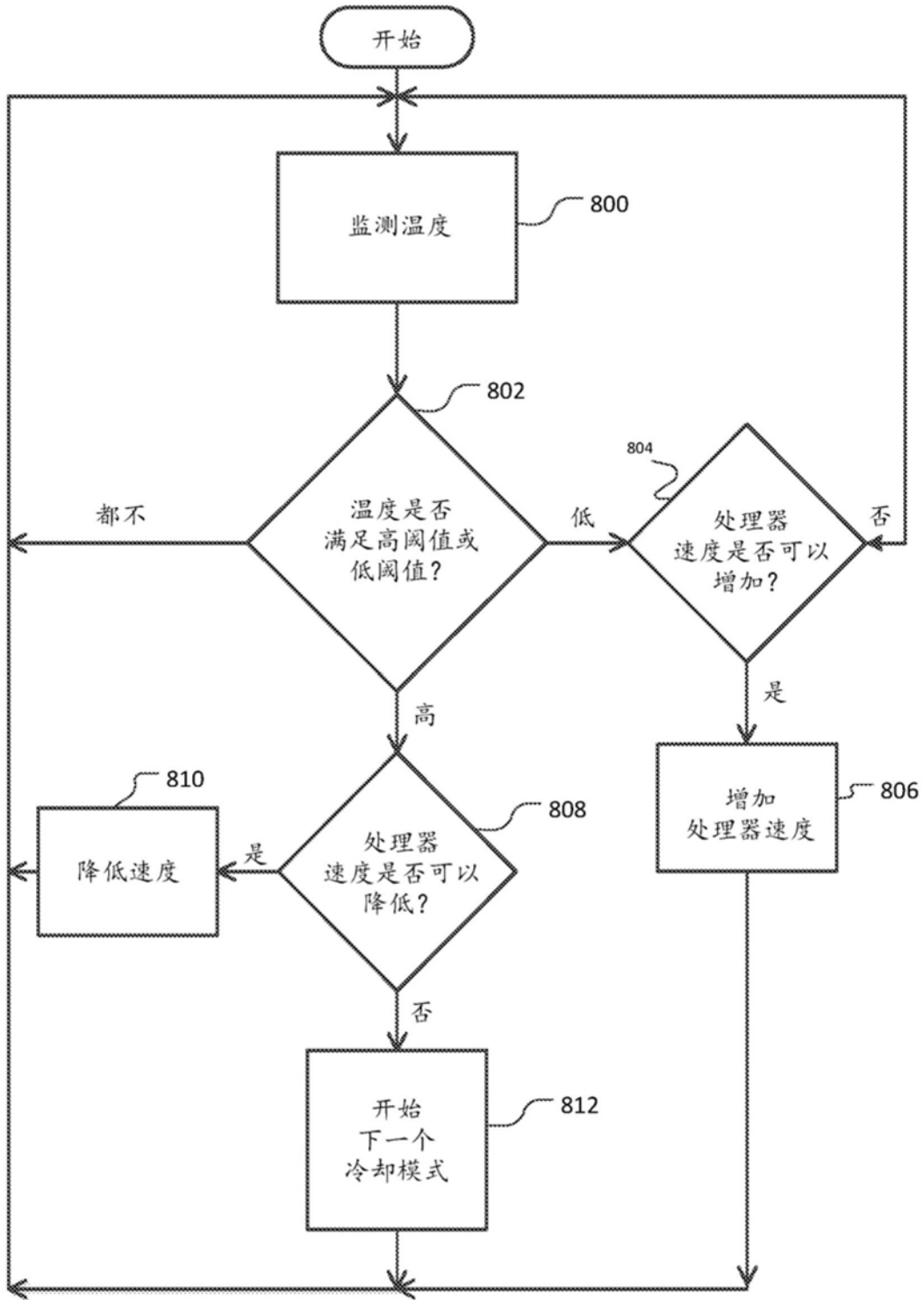


图8

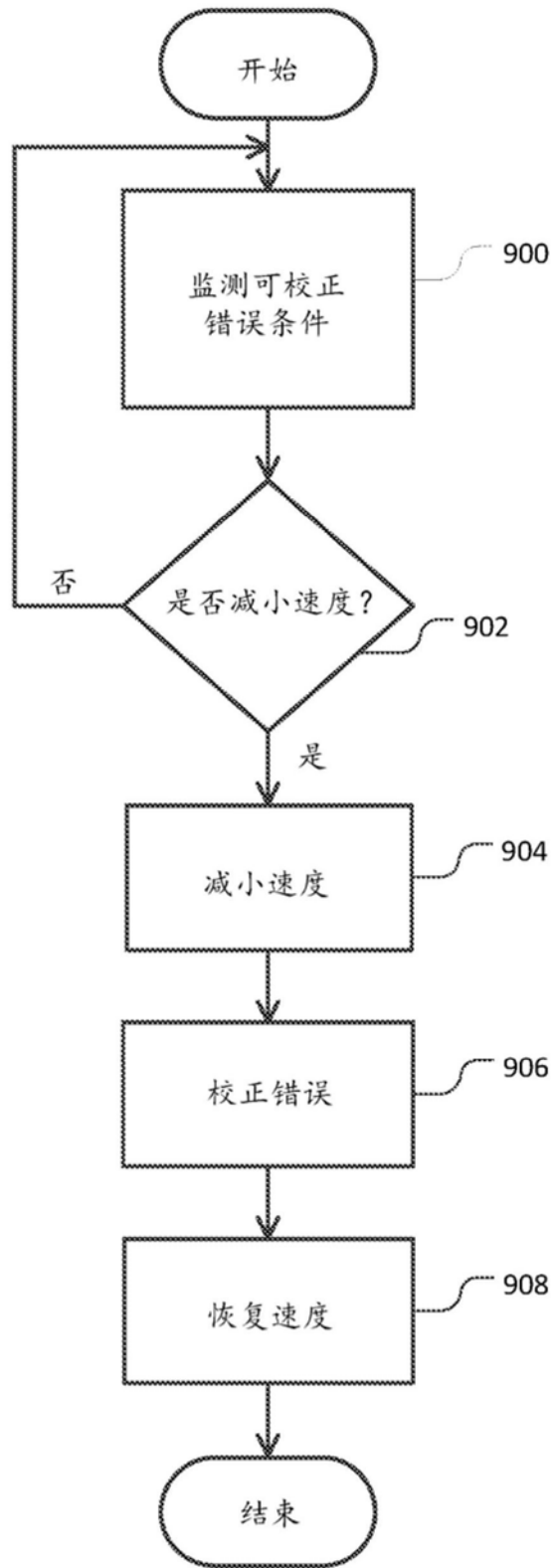


图9