



(12) 发明专利申请

(10) 申请公布号 CN 114489306 A

(43) 申请公布日 2022. 05. 13

(21) 申请号 202210144973.7

(51) Int. Cl.

(22) 申请日 2016.06.14

G06F 1/3203 (2019.01)

(30) 优先权数据

G06F 1/3237 (2019.01)

14/812,056 2015.07.29 US

G06F 1/324 (2019.01)

(62) 分案原申请数据

G06F 1/329 (2019.01)

201680044544.5 2016.06.14

G06F 1/3293 (2019.01)

(71) 申请人 英特尔公司

G06F 9/38 (2006.01)

地址 美国加利福尼亚州

G06F 12/0831 (2016.01)

G06F 15/78 (2006.01)

(72) 发明人 A·詹德勒 L·诺瓦克夫斯基

K·V·西斯特拉 V·加格

D·穆拉 A·V·乔保尔

E·G·哈尔诺尔 K·C·威尔

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

专利代理师 陈依心 黄嵩泉

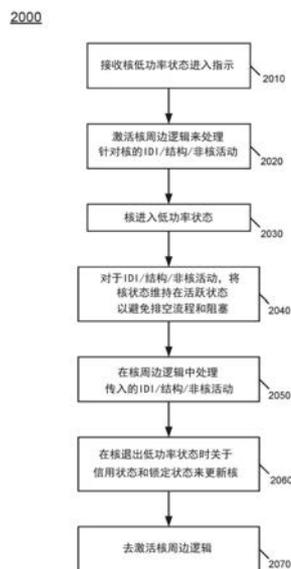
权利要求书3页 说明书27页 附图20页

(54) 发明名称

遮蔽处理器的核的功率状态

(57) 摘要

本申请公开了遮蔽处理器的核的功率状态。在一个实施例中,处理器包括用于执行指令的核以及耦合到该核的核周边逻辑。核周边逻辑可包括耦合到核的结构接口逻辑。进而,结构接口逻辑可包括第一存储,用于在核处于低功率状态时存储核的状态信息,并且使在核与非核之间耦合的管芯间互连在核进入低功率状态期间能被维持在活跃状态。描述了其他实施例并要求它们的权利。



1. 一种处理器,包括:

核,所述核用于执行指令;以及

核周边逻辑,所述核周边逻辑被耦合到所述核,所述核周边逻辑包括:

具有第一存储的结构接口逻辑,所述第一存储用于在所述核处于低功率状态时存储所述核的状态信息,

其中所述核周边逻辑用于使在所述核与非核之间耦合的管芯间互连在所述核进入低功率状态期间能被维持在活跃状态,并且

其中所述结构接口逻辑包括监听逻辑,所述监听逻辑用于在所述核处于所述低功率状态时从请求方接收监听请求,并在所述核将保持在所述低功率状态时将监听响应发送给所述请求方。

2. 如权利要求1所述的处理器,其特征在于,所述核周边逻辑用于使所述核的功率状态能对所述非核透明,以使所述非核能在所述核进入所述低功率状态的进入流程期间维持多个待决事务,而不排空所述多个待决事务。

3. 如权利要求1或2所述的处理器,其特征在于,所述核周边逻辑进一步包括边带事件逻辑,所述边带事件逻辑用于接收并存储经由被耦合到所述核周边逻辑的边带互连而接收的一个或多个边带事件,所述边带事件逻辑用于向功率控制器通知关于所述一个或多个边带事件的接收,以使所述功率控制器能确定是否要使所述核退出所述低功率状态来处理所述一个或多个边带事件中的至少一个。

4. 如权利要求1至3中任一项所述的处理器,其特征在于,所述核进一步包括报告寄存器,所述报告寄存器用于存储在所述核退出所述低功率状态之后、在恢复来自所述核的存储器的保持部分的状态信息期间的与不可纠正的错误的发生相关联的错误信息。

5. 如权利要求4所述的处理器,其特征在于,所述核将执行微代码来将来自所述报告寄存器的所述错误信息写入到机器校验架构存储,以使机器校验架构错误能被提出。

6. 如权利要求1至5中任一项所述的处理器,其特征在于,所述核是用于执行指令的多个核中的一个核。

7. 如权利要求1至6中任一项所述的处理器,进一步包括:

监测器逻辑,所述监测器逻辑用于在所述核处于低功率状态时监测一个或多个被监测位置以对存储在其中的值进行更新,其中在此类更新后,所述结构接口逻辑用于将唤醒请求传达到所述核。

8. 如权利要求1至7中任一项所述的处理器,其特征在于,所述结构接口逻辑处于第一功率域,在所述核处于所述低功率状态时所述第一功率域被提供功率和时钟信号。

9. 如权利要求8所述的处理器,其特征在于,所述核处于第二功率域,所述处理器进一步包括:

功率控制单元,所述功率控制单元用于独立地控制所述第一功率域和所述第二功率域的功率。

10. 一种处理器中的方法,包括:

在所述处理器的核处于低功率状态时,将所述核的状态信息存储在所述处理器的核周边逻辑的结构接口逻辑的第一存储中;

由所述核周边逻辑使在所述核与非核之间耦合的管芯间互连在所述核进入低功率状

态期间能被维持在活跃状态;以及

由所述结构接口逻辑的监听逻辑在所述核处于所述低功率状态时从请求方接收监听请求,并且由所述监听逻辑在所述核将保持在所述低功率状态时将监听响应发送给所述请求方。

11. 如权利要求10所述的方法,进一步包括:

由所述核周边逻辑使所述核的功率状态能对所述非核透明;以及

由所述核周边逻辑使所述非核能在所述核进入所述低功率状态的进入流程期间维持多个待决事务,而不排空所述多个待决事务。

12. 如权利要求10或11所述的方法,进一步包括:

由所述核周边逻辑的边带事件逻辑接收经由被耦合到所述核周边逻辑的边带互连而接收的一个或多个边带事件;

由所述边带事件逻辑存储所述一个或多个边带事件;以及

由所述边带事件逻辑向功率控制器通知关于所述一个或多个边带事件的接收,以使所述功率控制器能确定是否要使所述核退出所述低功率状态来处理所述一个或多个边带事件中的至少一个。

13. 如权利要求10至12中任一项所述的方法,进一步包括:

在所述核的报告寄存器中存储在所述核退出所述低功率状态之后、在恢复来自所述核的存储器的保持部分的状态信息期间的与不可纠正的错误的发生相关联的错误信息。

14. 如权利要求13所述的方法,进一步包括:

通过所述核执行微代码,将来自所述报告寄存器的所述错误信息写入到机器校验架构存储,以使机器校验架构错误能被提出。

15. 如权利要求10至14中任一项所述的方法,进一步包括:

由所述处理器的监测器逻辑在所述核处于低功率状态时监测一个或多个被监测位置以对存储在其中的值进行更新;以及

在此类更新后,由所述结构接口逻辑将唤醒请求传达到所述核。

16. 如权利要求10至15中任一项所述的方法,其特征在于,所述结构接口逻辑处于第一功率域,在所述核处于所述低功率状态时所述第一功率域被提供功率和时钟信号。

17. 如权利要求16所述的方法,其特征在于,所述核处于第二功率域,所述方法进一步包括:

由所述处理器的功率控制单元独立地控制所述第一功率域和所述第二功率域的功率。

18. 一种计算机程序产品,包括指令,所述指令当由处理器执行时,使所述处理器执行如权利要求10至17中任一项所述的方法。

19. 一种计算机可读介质,具有存储于其上的如权利要求18所述的计算机程序产品。

20. 一种多域处理器,包括:

核域,包括多个核;

图形域,包括一个或多个图形引擎;

系统代理域,包括功率控制单元和显示控制器,以独立于所述核域的独立频率执行;以

及

环形互连,用于耦合所述核域、所述图形域和所述系统代理域。

21. 一种处理器核,包括:
取出单元,用于取出指令;
解码单元,用于对所述指令解码;
发出逻辑,用于将经解码的指令发出到多个执行单元中的一个执行单元;
所述多个执行单元中的一个执行单元用于执行所述经解码的指令并将结果提供到写回单元。

22. 一种系统,包括:
第一处理器和第二处理器,经由点对点互连耦合,并且分别耦合至相应的存储器;
芯片组,经由点对点互连耦合至所述第一处理器和所述第二处理器;
第一总线,经由接口耦合至所述芯片组;
第二总线,经由总线桥接器耦合至所述第一总线;以及
数据存储单元,包括代码,所述数据存储单元耦合至所述第二总线。

遮蔽处理器的核的功率状态

本发明专利申请是国际申请号为PCT/US2016/037347,国际申请日为2016年6月14日,进入中国国家阶段的申请号为201680044544.5,名称为“遮蔽处理器的核的功率状态”的发明专利申请的分案申请。

技术领域

[0001] 各实施例涉及对系统的功率管理,更具体而言,涉及对多核处理器的功率管理。

背景技术

[0002] 半导体处理和逻辑设计中的进步已允许可存在于集成电路器件上的逻辑数量的增长。作为结果,计算机系统配置已从系统中的单个或多个集成电路演进到单个集成电路上的多个硬件线程、多个核、多个设备和/或多个完整系统。另外,随着集成电路的密度增长,对计算系统(从嵌入式系统到服务器)的功率要求也逐步升高。此外,软件的低效率以及其对硬件的要求也已导致计算设备能耗的增加。事实上,一些研究指出,计算设备消耗了诸如美国之类的国家的全部电力供应中相当大的百分比。作为结果,存在对与集成电路相关联的能效和节能的迫切需求。随着服务器、台式计算机、笔记本、超极本™、平板、移动电话、处理器、嵌入式系统等变得越来越流行(从被包括在典型的计算机、汽车和电视机中到生物技术中),这些需求将增长。

附图简述

- [0003] 图1是根据本发明的实施例的系统的部分的框图。
- [0004] 图2是根据本发明的实施例的处理器的框图。
- [0005] 图3是根据本发明的另一实施例的多域处理器的框图。
- [0006] 图4是包括多个核的处理器的实施例。
- [0007] 图5是根据本发明的一个实施例的处理器核的微架构的框图。
- [0008] 图6是根据另一实施例的处理器核的微架构的框图。
- [0009] 图7是根据又一实施例的处理器核的微架构的框图。
- [0010] 图8是根据再一实施例的处理器核的微架构的框图。
- [0011] 图9是根据本发明的另一实施例的处理器的框图。
- [0012] 图10是根据本发明的实施例的代表性SoC的框图。
- [0013] 图11是根据本发明的实施例的另一示例SoC的框图。
- [0014] 图12是可与多个实施例一起使用的示例系统的框图。
- [0015] 图13是可与多个实施例一起使用的另一示例系统的框图。
- [0016] 图14是代表性计算机系统的框图。
- [0017] 图15是根据本发明的实施例的系统的框图。
- [0018] 图16是根据本发明的实施例的处理器的框图。
- [0019] 图17是根据实施例的监听响应逻辑的进一步细节的框图。
- [0020] 图18是根据实施例的内部处理器存储器的高层级框图。

[0021] 图19是根据本发明的实施例的处理器框图。

[0022] 图20是根据实施例的功率控制方法的流程图。

具体实施方式

[0023] 在各实施例中,处理器或其他片上系统(SOC)的核或其他知识产权(IP)逻辑块的功率状态可对包括结构和其他IP块的处理器其他资产保持隐藏。更具体地,实施例提供核周边逻辑,核周边逻辑是与核相关联的、充当核与其他处理器组件之间的接口的、包括一个或多个结构和其他IP块的外围逻辑。

[0024] 通过使核的真实功率状态对处理器的剩余部分隐藏或维持透明,到核的通信即使在核处于低功率状态时仍可继续。该核周边逻辑可被配置为处理或者对于其他逻辑来说至少看上去正在处理传入业务量,使得核可在更长的时间段内保持在低功率状态(并且可能是更深的低功率状态)。此外,通过将功率状态的这种透明性提供给其他处理器组件,会出现进入和退出给定功率状态的进入和退出流程的减少,从而减少用于低功率状态转换的等待时间。此外,通过减少流程并且维持核的功率状态对其他组件不可见,避免了在功率管理进入和退出流程期间使此类其他组件排空事务的需要,从而大大减少了低功率状态进入和退出对处理器性能的影响。

[0025] 现在参考图1,所示的是根据本发明的实施例的系统的部分的框图。如图1所示,系统100可以包括各种组件,包括被示为多核处理器的处理器110。处理器110可经由外部电压调节器160被耦合至电源150,该外部电压调节器160可执行第一电压转换以将初级已调节电压提供给处理器110。

[0026] 如所见,处理器110可以是包括多个核120a-120n的单管芯处理器。另外,每一个核可与集成的电压调节器(IVR) 125a-125n相关联,这些集成的电压调节器接收初级已调节电压并生成操作电压,该操作电压被提供到与IVR相关联的处理器一个或多个代理。相应地,可提供IVR实现以允许对每一个单个核的电压进而对其功率和性能进行细粒度的控制。如此,每一个核都可在独立的电压和频率下操作,从而允许极大的灵活性,并提供平衡功耗与性能的广泛机会。在一些实施例中,对多个IVR的使用允许将组件分组到不同的功率层中,使得功率被IVR调节并且仅被提供给组中的那些组件。在功率管理期间,当处理器被置于某个低功率状态时,一个IVR的给定的功率层可以被掉电或断电,而另一IVR的另一功率层保持为活跃的或完全供电的。

[0027] 仍参考图1,附加的组件可存在于处理器内,包括输入/输出接口132、另一接口134以及集成存储器控制器136。如所见,这些组件中的每一个都可由另一集成的电压调节器125_x来供电。在一个实施例中,接口132可以是用于英特尔®快速路径互连(QPI)互连的使能操作,该互连在包括多个层的高速缓存一致性协议中提供点对点(PtP)链路,所述多个层包括物理层、链路层和协议层。进而,接口134可经由外围组件互连快速(PCIe™)协议来通信。

[0028] 同样被示出的是功率控制单元(PCU) 138,其可包括用于执行关于处理器110的功率管理操作的硬件、软件和/或固件。如所见,PCU 138经由数字接口将控制信息提供给外部电压调节器160以使该电压调节器生成合适的已调节电压。PCU 138也经由另一数字接口将控制信息提供给多个IVR 125以控制所生成的操作电压(或使相应的IVR在低功率模式下被

禁用)。在各实施例中,PCU 138可包括用于执行基于硬件的功率管理的各种功率管理逻辑单元。此类功率管理可以是完全被处理器控制的(例如,通过各种处理器硬件,并且其可以是被工作负荷和/或功率、热或其他处理器约束触发的),并且/或者功率管理可以响应于外部源(诸如平台或管理功率管理源或系统软件)而被执行。

[0029] 尽管为了易于说明没有示出,但应理解诸如非核(uncore)逻辑和其他组件(诸如,内部存储器(例如,一个或多个层级的高速缓存存储器层次结构等))之类的附加组件可存在于处理器110内。此外,尽管在图1的实现中示出为具有集成的电压调节器,但是,多个实施例不限于此。

[0030] 注意,本文中所描述的功率管理技术可以独立于基于操作系统(OS)的功率管理(OSPM)机制,并与其互补。根据一个示例OSPM技术,处理器可在各种性能状态或等级(所谓的P状态,即从P0到PN)上操作。一般而言,P1性能状态可对应于可由OS请求的最高保证的性能状态。除此P1状态之外,OS可进一步请求更高的性能状态,即,P0状态。该P0状态因此可以是机会型状态或超频模式状态,在该状态中,当功率和/或热预算是可用的时,处理器硬件可配置处理器或其至少多个部分,以便以高于保证频率的频率进行操作。在许多实现中,处理器可包括在制造期间被融合或以其他方式写入到处理器中的,高于P1保证的最大频率的,超出特定处理器的最大峰值频率的多个所谓的元(bin)频率。此外,根据一个OSPM机制,处理器可在各种功率状态或等级上进行操作。对于功率状态,OSPM机制可以指定不同的功耗状态,一般将其称为C状态(C0、C1到Cn状态)。当核是活跃的时,它以C0状态运行,而当该核是空闲的时,它可被置于核低功率状态,也被称为核非零C状态(例如,C1-C6状态),每一个C状态都处于更低的功耗等级(使得C6是比C1更深的低功率状态,等等)。

[0031] 应理解许多不同类型的功率管理技术可以在不同实施例中单独使用或组合使用。作为代表性示例,功率控制器可以控制处理器以由一些形式的动态电压频率缩放(DVFS)来对该处理器进行功率管理,在DVFS中一个或多个核或其他处理器逻辑的操作电压和/或操作频率可被动态地控制以在某些情形下减少功耗。在示例中,可以使用加利福尼亚州圣克拉拉市的英特尔公司提供的增强型Intel SpeedStep™技术来执行DVFS,以在最低功耗水平下提供最优性能。在另一示例中,可以使用Intel TurboBoost™技术来执行DVFS以使得一个或多个核或其他计算引擎能基于条件(例如,工作负荷和可用性)以高于保证操作频率的频率进行操作。

[0032] 可以在某些示例中使用的另一功率管理技术是在不同计算引擎之间动态交换工作负荷。例如,处理器可以包括以不同功耗水平操作的非对称核或其他处理引擎,使得在功率约束的情形下,一个或多个工作负荷可以被动态地切换以在较低功率核或其他计算引擎上执行。另一示例性功率管理技术是硬件轮停(HDC),硬件轮停(HDC)可以使核和/或其他计算引擎根据工作周期被周期性地启用和禁用,使得一个或多个核在工作周期的不活跃时段变为不活跃的,而在工作周期的活跃时段变为活跃的。虽然以这些特定示例描述,但应理解许多其他功率管理技术可以在特定实施例中使用。

[0033] 可在用于各种市场的处理器(包括服务器处理器、台式机处理器、移动处理器等)中实现多个实施例。现在参考图2,所示为根据本发明的实施例的处理器的框图。如图2中所示,处理器200可以是包括多个核 210_a - 210_n 的多核处理器。在一个实施例中,每一个此类核可以是独立的功率域,并且可被配置成基于工作负荷进入和退出活跃状态和/或最大性能

状态。每个核210可与相应的核周边逻辑相关联。一般来说,核周边逻辑可包括在核电路与处理器的剩余部分之间提供接口的一个或多个独立功率/频率域。此外,每个核周边逻辑可与多个组成逻辑单元一起被配置为在相应的核处于给定低功率状态时,代表相应的核执行各种操作,以允许该核的功率状态对处理器的剩余部分隐藏。此外,每个核周边逻辑的一个或多个独立存储单元可适于存储相关联的核的至少某些上下文信息以允许快速进入和退出特定低功率状态并且进一步允许某些处理器操作(诸如中断处理、监听响应、信用处理等等)在相应核处于低功率状态时发生。

[0034] 可经由互连215将各个核耦合到包括各种组件的系统代理或非核220。如所见,非核220可包括共享高速缓存230,该共享高速缓存230可以是末级高速缓存。此外,非核可以包括集成存储器控制器240,用于例如经由存储器总线与系统存储器(图2中未示出)通信。非核220还包括各种接口250a-250n和功率控制单元255,功率控制单元255可包括用于执行本文所述的功率管理技术的逻辑。在一些情形中,除了在给定核210与非核220之间的直接连接之外,核周边逻辑还可被直接耦合到非核220的至少一部分。

[0035] 此外,通过接口250a-250n,可完成到诸如外围设备、大容量存储设备等之类的各种芯片外组件的连接。尽管在图2的实施例中以特定实现示出,但是本发明的范围不限于该方面。

[0036] 现在参考图3,所示出的是根据本发明的另一实施例的多域处理器的框图。如图3的实施例中所示,处理器300包括多个域。具体而言,核域310可以包括多个核310a-310n,图形域(例如图形引擎320)可以包括一个或多个图形引擎,并且还可以存在系统代理域(例如,系统代理350)。在一些实施例中,系统代理域(例如,系统代理350)能以独立于核域的独立频率执行,并且可在全部的时刻保持被供电以处理功率控制事件和功率管理,使得可以控制域310和320以使其动态地进入和退出高功率状态和低功率状态。域310和320中的每一个能以不同的电压和/或功率操作。注意,虽然仅以三个域示出,但是应理解,本发明的范围不限于此方面,并且附加的域可存在于其他实施例中。例如,多个核域可存在,每一个核域包括至少一个核。

[0037] 一般而言,除各种执行单元和附加的处理元件之外,每个核310a-310n还可包括多个低层级高速缓存。进而,各个核可彼此耦合,并且被耦合到共享高速缓存存储器,该共享高速缓存存储器由末级高速缓存(LLC)340a-340n的多个单元形成。在各实施例中,可在多个核与图形引擎以及各种媒体处理电路之间共享LLC 340a-340n。如所见,环形互连330由此将多个核耦合到一起,并且在多个核、图形域(例如图形引擎320)和系统代理域(例如系统代理350)之间提供互连。在一个实施例中,互连330可以是核域的部分。然而,在其他实施例中,该环形互连可以属于其自身的域。如进一步所示,多个核周边逻辑每个可与给定核相关联,并且除了提供(例如,如在低功率进入和退出情形期间所使用的)有效存储和上下文信息的检索之外,还允许相应核的功率状态对其他处理器代理维持隐藏。在图3的例示中,核周边逻辑被示为在相应核310a-310n与环形互连330之间耦合。然而,可以理解,在一些实施例中,核310a-310n与环形互连330之间的直接连接可连同核周边逻辑与环形互连330之间的相应的直接连接一起存在。

[0038] 如进一步所见,系统代理域(例如,系统代理350)可包括显示控制器352,该显示控制器352可提供对相关联的显示器的控制以及到它的接口。如进一步所见,系统代理域(例

如,系统代理350)可包括功率控制单元355,该功率控制单元355可包括用于执行本文所描述的功率管理技术的逻辑。

[0039] 如在图3中进一步所见的,处理器300可进一步包括集成存储器控制器(IMC)370,该集成存储器控制器(IMC)370可提供到诸如动态随机存取存储器(DRAM)之类的系统存储器的接口。多个接口380a-380n可存在以允许处理器与其他电路之间的互连。例如,在一个实施例中,可提供至少一个直接媒体接口(DMI)接口以及一个或多个PCIeTM接口。此外,为了提供诸如附加处理器或其他电路之类的其他代理之间的通信,还可以提供一个或多个QPI接口。尽管在图3的实施例中在该高层级示出,但是会理解,本发明的范围不限于该方面。

[0040] 参考图4,例示出了包括多个核的处理器实施例。处理器400包括任何处理器或处理设备,诸如微处理器、嵌入式处理器、数字信号处理器(DSP)、网络处理器、手持式处理器、应用处理器、协处理器、片上系统(SoC)、或用于执行代码的其它器件。在一个实施例中,处理器400包括至少两个核——核401和402,它们可包括非对称核或对称核(所例示的实施例)。然而,处理器400可包括可以是对称的或非对称的任何数量的处理元件。

[0041] 在一个实施例中,处理元件指的是用于支持软件线程的硬件或逻辑。硬件处理元件的示例包括:线程单元、线程槽、线程、进程单元、上下文、上下文单元、逻辑处理器、硬件线程、核和/或能够保持处理器的诸如执行状态或架构状态之类的状态的任何其他元件。换言之,在一个实施例中,处理元件指的是能够与诸如软件线程、操作系统、应用、或其他代码之类的代码独立地相关联的任何硬件。物理处理器通常指的是集成电路,其潜在地包括诸如核或硬件线程之类的任意数量的其他处理元件。

[0042] 核通常指的是位于集成电路上的能够维持独立架构状态的逻辑,其中每个独立维持的架构状态与至少一些专用执行资源相关联。与核相反,硬件线程通常指的是位于集成电路上的能够维持独立架构状态的任何逻辑,其中独立维持的架构状态共享对执行资源的访问。如可见的,当某些资源是共享的而其他资源是专用于架构状态的时候,硬件线程与核的术语之间的界线交叠。但核和硬件线程常常被操作系统视为单个的逻辑处理器,其中,操作系统能够分别在每一个逻辑处理器上调度操作。

[0043] 如在图4中所例示的,物理处理器400包括两个核——核401和402。在此,核401和402被认为是对称核,即,具有相同的配置、功能单元和/或逻辑的核。在另一实施例中,核401包括乱序处理器核,而核402包括有序处理器核。然而,核401和402可从任何类型的核中单独地选出,诸如原生核、受软件管理的核、适于执行原生指令集架构(ISA)的核、适于执行经转换ISA的核、协同设计的核或其他已知核。不过,为进一步讨论,以下将进一步详细地描述在核401中所例示的功能单元,同样地,核402中的单元以类似方式操作。

[0044] 如所描绘的,核401包括两个硬件线程,它们还可被称为硬件线程槽。因此,在一个实施例中,诸如操作系统之类的软件实体潜在地将处理器400视为四个单独的处理器,即能够并发地执行四个软件线程的四个逻辑处理器或处理元件。如上所述,第一线程与架构状态寄存器401a相关联,第二线程与架构状态寄存器401b相关联,第三线程可与架构状态寄存器402a相关联,而第四线程可与架构状态寄存器402b相关联。在此,架构状态寄存器(401a、401b、402a和402b)中的每一个可被称为处理元件、线程槽或线程单元,如上所述。如所例示的,架构状态寄存器401a被复制在架构状态寄存器401b中,因此能够为逻辑处理器存储各个架构状态/上下文。在核401中,也可复制用于线程的其他较小资源,诸如,指令指

针以及分配器和重命名器块430中的重命名逻辑。可通过分区来共享诸如重排序/引退单元435中的重排序缓冲器、ILTB 420、加载/存储缓冲器和队列之类的一些资源。可潜在地完全共享诸如通用内部寄存器、页表基寄存器、低层级数据高速缓存和数据TLB 451、执行单元440和重排序/引退单元435的多个部分之类的其他资源。

[0045] 处理器400通常包括其他资源,它们可被完全共享,可通过分区被共享,或可由处理元件专用/专用于处理元件。在图4中,例示出具有处理器的说明性逻辑单元/资源的纯示例性处理器的实施例。注意,处理器可包括或省略这些功能单元中的任何单元,并包括未描绘出的任何其他已知的功能单元、逻辑或固件。如所例示的,核401包括简化的、代表性的乱序(000)处理器核。但是,在不同实施例中,可利用有序处理器。000核包括用于预测将被执行/进行的分支的分支目标缓冲器420以及用于存储指令的地址转换条目的指令转换缓冲器(I-TLB) 420。

[0046] 核401进一步包括被耦合至取出单元来解码所取出的元素的解码模块425。在一个实施例中,取出逻辑包括分别与各个线程槽相关联的各个序列发生器。通常,核401与第一ISA相关联,该第一ISA定义/指定在处理器400上可执行的指令。通常,作为第一ISA的部分的机器代码指令包括引用/指定将被执行的指令或操作的指令的部分(被称为操作码)。解码逻辑425包括从这些指令的操作码识别出这些指令并在流水线上传递经解码的指令以用于第一ISA所定义的处理的电路。例如,在一个实施例中,解码器425包括被设计成用于或适于识别诸如事务性指令之类的特定指令的逻辑。作为由解码器425识别的结果,架构或核401采取特定的、预定义的动作来执行与适当指令相关联的任务。重要的是应注意,可响应于单条或多条指令来执行本文中所描述的任务、块、操作和方法中的任何一个;所述单条或多条指令中的一些可以是新指令或旧指令。

[0047] 在一个示例中,分配器和重命名器块430包括用于保持资源的分配器,诸如,用于存储指令处理结果的寄存器堆。然而,各个线程潜在地能够进行乱序执行,其中,分配器和重命名器块430还保持其他资源,诸如,用于跟踪指令结果的重排序缓冲器。分配器和重命名器块430还可包括寄存器重命名器,该寄存器重命名器用于将程序/指令引用寄存器重命名为处理器400内部的其他寄存器。重排序/引退单元435包括用于支持被乱序执行的指令的乱序执行和稍后的有序引退的组件,诸如上述的重排序缓冲器、加载缓冲器和存储缓冲器。

[0048] 在一个实施例中,调度器和执行单元块440包括用于在执行单元上调度指令/操作的调度器单元。例如,在具有可用的浮点执行单元的执行单元的端口上调度浮点指令。与执行单元相关联的寄存器堆也被包括以用于存储信息指令处理结果。示例性的执行单元包括浮点执行单元、整数执行单元、跳转执行单元、加载执行单元、存储执行单元以及其他已知的执行单元。

[0049] 较低层级的数据高速缓存和数据转换缓冲器(D-TLB) 450被耦合至执行单元440。数据高速缓存用于存储最近使用/操作的元素(诸如,数据操作数),这些元素在存储器一致性状态下被潜在地保持。D-TLB用于存储最近的虚拟/线性地址至物理地址的转换。作为特定示例,处理器可包括页表结构,页表结构用于将物理存储器分解成多个虚拟页。

[0050] 在此,核401和402共享对较高级或进一步远离的高速缓存410的访问,该高速缓存410用于缓存最近取出的元素。注意,较高级或进一步远离指的是高速缓存层级增加或

进一步远离执行单元。在一个实施例中,较高级高速缓存410是末级数据高速缓存——处理器400上的存储器层次结构中的末级高速缓存——诸如,第二或第三级数据高速缓存。然而,较高级高速缓存410不限于此,因为它可与指令高速缓存相关联或包括指令高速缓存。替代地,跟踪高速缓存——一种指令高速缓存——可被耦合在解码器425之后来存储最近经解码的踪迹。

[0051] 在所描绘的配置中,处理器400还包括总线接口模块405和功率控制器460,功率控制器460可执行根据本发明的实施例的功率管理。在该情形下,总线接口405将与处理器400外部的设备(诸如系统存储器和其他组件)通信。

[0052] 存储器控制器470可以与诸如一个或多个存储器之类的其他设备对接。在示例中,总线接口405包括环形互连,其与用于与存储器对接的存储器控制器和用于与图形处理器对接的图形控制器互连。在SoC环境中,诸如网络接口、协处理器、存储器、图形处理器以及任何其它已知计算机设备/接口之类的甚至更多的设备可被集成到单个管芯或集成电路上,以提供具有高功能性和低功耗的小形状因数。

[0053] 现在参照图5,所示出的是根据本发明的一个实施例的处理器核的微架构的框图。如图5所示,处理器核500可以是多级流水线类型的乱序处理器。核500可基于接收到的操作电压在各种电压下操作,所述接收到的操作电压可从集成电压调节器或外部电压调节器接收。

[0054] 如在图5所见的,核500包括前端单元510,该前端单元510可用于取出将被执行的指令并将这些指令准备好以供稍后在处理器流水线中使用。例如,前端单元510可包括取出单元501、指令高速缓存503以及指令解码器505。在一些实现中,前端单元510可进一步包括跟踪高速缓存、微代码存储以及微操作存储。取出单元501可,例如,从存储器或指令高速缓存503取出宏指令并将它们馈送至指令解码器505以将它们解码为原语,即,用于被处理器执行的微操作。

[0055] 乱序(000)引擎515耦合在前端单元510与执行单元520之间,该乱序引擎515可用于接收微指令并将它们准备好以供执行。更具体地,000引擎515可包括各种缓冲器,所述各种缓冲器用于对微指令流进行重排序并分配执行所需的多个资源,以及提供对各寄存器堆(诸如,寄存器堆530和扩展寄存器堆535)中的存储位置上的逻辑寄存器的重命名。寄存器堆530可包括用于整数和浮点操作的单独的寄存器堆。出于配置、控制和附加操作的目的,一组机器专用寄存器(MSR)538也可存在并且可由核500内(以及核外部)的各种逻辑来访问。例如,功率限制信息可以存储在一个或多个MSR中并且如本文中所描述的被动态地更新。

[0056] 在执行单元520中可存在各种资源,包括,例如,各种整数、浮点和单指令多数据(SIMD)逻辑单元以及其它专门硬件。例如,此类执行单元可包括一个或多个算术逻辑单元(ALU)522和一个或多个向量执行单元(VEU)524,以及其他此类执行单元。

[0057] 来自执行单元的结果可被提供至引退逻辑,即重排序缓冲器(ROB)540。更具体地,ROB 540可包括用于接收与被执行的指令相关联的信息的各种阵列和逻辑。然后,该信息被ROB 540检查以确定指令是否可以有效引退并且结果数据是否被提交至处理器的架构状态,或阻止指令的正常引退的一个或多个异常是否发生。当然,ROB 540可处理与引退相关联的其他操作。

[0058] 如图5所示, ROB 540被耦合至高速缓存550, 在一个实施例中, 高速缓存550可以是低层级高速缓存(例如, L1高速缓存), 尽管本发明的范围不限于此。而且, 执行单元520可直接耦合至高速缓存550。从高速缓存550, 可发生与更高层级高速缓存、系统存储器等等的通信。虽然在图5的实施例中以此高级别示出, 但应理解本发明的范围不限于此方面。例如, 虽然图5的实现方式涉及了诸如具有Intel® x86指令集架构 (ISA) 的乱序机器, 但本发明的范围在此方面不受限制。即, 其他实施例可在以下处理器中实现: 有序处理器; 诸如基于ARM的处理器器的精简指令集计算 (RISC) 处理器; 或具有另一类型ISA的处理器, 该另一类型的ISA可经由仿真引擎和相关联的逻辑电路来仿真不同ISA的指令和操作。

[0059] 现在参照图6, 所示出的是根据另一实施例的处理器核的微架构的框图。在图6的实施例中, 核600可以是具有不同的微架构的低功率核, 诸如设计为降低功耗的具有相对受限制的流水线深度的Intel® 基于凌动™ (Atom™) 的处理器。如所见, 核600包括被耦合以将指令提供到指令解码器615的指令高速缓存610。分支预测器605可被耦合到指令高速缓存610。注意, 指令高速缓存610可以进一步被耦合到高速缓存存储器的另一层级, 诸如L2高速缓存(为了易于说明, 图6中未示出)。进而, 指令解码器615将经解码的指令提供到发出队列 (IQ) 620以供存储和传递到给定的执行流水线。微代码ROM 618被耦合到指令解码器615。

[0060] 浮点流水线630包括浮点寄存器堆632, 该浮点寄存器堆632可包括具有给定位(诸如128、256或512位)的多个架构寄存器。流水线630包括用于调度指令以在流水线的多个执行单元之一上执行的浮点调度器634。在所示的实施例中, 此类执行单元包括ALU 635、混洗单元636和浮点加法器638。进而, 在这些执行单元中生成的结果可以被往回提供到缓冲器和/或寄存器堆632的寄存器。当然, 应理解虽然以这几个示例执行单元示出, 但是在另一实施例中可以存在附加的或不同的浮点执行单元。

[0061] 整数流水线640也可被提供。在所示的实施例中, 流水线640包括整数寄存器堆642, 整数寄存器堆642可包括具有给定位(诸如128或256位)的多个架构寄存器。流水线640包括用于调度指令以在流水线的多个执行单元之一上执行的整数调度器644。在所示的实施例中, 此类执行单元包括ALU 645、移位器单元646和跳转执行单元648。进而, 在这些执行单元中生成的结果可以被往回提供到缓冲器和/或寄存器堆(例如, 整数寄存器堆642)的寄存器。当然, 应理解虽然以这几个示例执行单元示出, 但是在另一实施例中可以存在附加的或不同的整数执行单元。

[0062] 存储器执行调度器650可以调度存储器操作以在地址生成单元 (AGU) 652中执行, 地址生成单元652还被耦合到TLB 654。如所见, 这些结构可被耦合到数据高速缓存660, 数据高速缓存660可以是L0和/或L1数据高速缓存, 其进而耦合到高速缓存存储器层次结构的附加层级, 包括L2高速缓存存储器。

[0063] 为了提供对乱序执行的支持, 可以提供分配器/重命名器670以作为重排序缓冲器680的附加, 重排序缓冲器680被配置为对被乱序地执行的指令进行重排序以供有序引退。虽然以图6的图示中的该特定流水线架构示出, 但是应理解许多变型和替代是可能的。

[0064] 注意, 在具有非对称核的处理器中, 诸如根据图5和6的微架构, 由于功率管理的原因, 可以在核之间动态地交换工作负荷, 因为这些核虽然具有不同的流水线设计和深度, 但是可以具有相同或相关的ISA。可以按照对用户应用(并且也可能对内核)透明的方式来执行此类动态核交换。

[0065] 参照图7,所示为根据又一实施例的处理器核的微架构的框图。如在图7中所例示的,核700可以包括用于以非常低的功耗水平来执行的多级有序流水线。作为一个此类示例,处理器700可以具有根据可从加利福尼亚州桑尼威尔市的ARM控股有限公司得到的ARM Cortex A53设计的微架构。在实现中,可以提供被配置为执行32位和64位代码的8级流水线。核700包括取出单元710,该取出单元710被配置为取出指令并将这些指令提供到解码单元715,解码单元715可以解码指令,例如具有给定ISA (诸如ARMv8 ISA) 的宏指令。进一步注意,队列730可以耦合到解码单元715以存储经解码的指令。经解码的指令被提供到发出逻辑725,其中可以将经解码的指令发出到多个执行单元中的给定的一个。

[0066] 进一步参考图7,发出逻辑725可以将指令发出到多个执行单元中的一个。在所示的实施例中,这些执行单元包括整数单元735、乘法单元740、浮点/向量单元750、双发出单元760以及加载/存储单元770。可以将这些不同的执行单元的结果提供到写回单元780。应理解,虽然为了便于说明示出单个写回单元,但是在一些实现中,分开的多个写回单元可以与执行单元中的每一个相关联。此外,应理解,虽然图7中示出的单元中的每一个和逻辑被表示为在高层级,但是特定实现可以包括更多或不同的结构。可以在许多不同的最终产品(从移动设备扩展到服务器系统)中实现使用一个或多个具有如图7中的流水线的核来设计的处理器。

[0067] 参照图8,所示出的是根据再一实施例的处理器核的微架构的框图。如在图8中所例示的,核800可以包括用于以非常高的性能水平(其可在高于图7的核700的功耗水平下发生)执行的多级多发出乱序流水线。作为一个此类示例,处理器800可以具有根据ARM Cortex A57设计的微架构。在实现中,可以提供被配置为执行32位和64位代码的15(或更大)级流水线。此外,流水线可以提供3路(或更大)宽度和3路(或更大)发出操作。核800包括取出单元810,取出单元810被配置为取出指令并将这些指令提供到解码器/重命名器/分派器815,解码器/重命名器/分派器815可以对指令(例如,具有ARMv8指令集架构的宏指令)进行解码、对指令内的寄存器引用进行重命名、并且将指令(最终)分派到所选择的执行单元。经解码的指令可被存储在队列825中。注意,虽然为了便于说明在图8中示出单个队列结构,但是应理解分开的多个队列可以被提供给多个不同类型的执行单元中的每一个。

[0068] 在图8中还示出发出逻辑830,存储在队列825中的经解码的指令可以从发出逻辑830被发出到所选择的执行单元。在特定实施例中,还可用针对与发出逻辑830耦合的多个不同类型的执行单元中的每一个的单独的发出逻辑来实现发出逻辑830。

[0069] 可以将经解码的指令发出到多个执行单元中的给定的一个。在所示的实施例中,这些执行单元包括一个或多个整数单元835、乘法单元840、浮点/向量单元850、分支单元860以及加载/存储单元870。在实施例中,浮点/向量单元850可被配置为处理128或256位的SIMD或向量数据。此外,浮点/向量执行单元850可以执行IEEE-754双精度浮点操作。这些不同的执行单元的结果可被提供到写回单元880。注意,在一些实现中,单独的写回单元可以与执行单元中的每一个相关联。此外,应理解,虽然图8中示出的单元中的每一个和逻辑被表示为在高层级,但是特定实现可以包括多个或不同的结构。

[0070] 注意,在具有非对称核的处理器中,诸如根据图7和8的微架构,由于功率管理的原因,可以动态地交换工作负荷,因为这些核虽然具有不同的流水线设计和深度,但是可以具有相同或相关的ISA。可以按照对用户应用(并且也可能对内核)透明的方式来执行此类动

态核交换。

[0071] 可以在许多不同的最终产品(从移动设备扩展到服务器系统)中实现使用具有如在图5-8中的任意一个或多个中的流水线的的一个或多个核设计的处理器。现在参考图9,所示出的是根据本发明的另一实施例的处理器的框图。在图9的实施例中,处理器900可以是包括多个域的SoC,该SoC的每一个域可被控制以在独立的操作电压和操作频率下进行操作。作为特定的说明性示例,处理器900可以是基于英特尔®架构核™的处理器,诸如,i3、i5、i7或可从英特尔公司得到的另一个此类处理器。然而,诸如可从加利福尼亚州桑尼威尔的超微半导体有限公司(AMD)得到的、来自ARM控股有限公司或其被许可方的基于ARM的设计的、或来自加利福尼亚州桑尼威尔的MIPS技术公司或它们的被许可方或采用者的基于MIPS的设计之类的其他低功率处理器可替代地存在于诸如苹果A7处理器、高通骁龙处理器或德州仪器OMAP处理器之类的其他实施例中。此类SoC可用于诸如智能电话、平板计算机、平板手机计算机、超极本™计算机或其他便携式计算设备之类的低功率系统中。

[0072] 在图9所示的高层级视图中,处理器900包括多个核单元910a-910n。每一个核单元都可包括一个或多个处理器核、一个或多个高速缓存存储器和其他电路。每一个核单元910a-910n都可支持一个或多个指令集(例如,x86指令集(具有与较新版本一起添加的一些扩展);MIPS指令集;ARM指令集(具有诸如NEON的可选附加扩展))或其他指令集或其组合。注意,核单元中的一些可以是异构资源(例如,具有不同设计)。此外,每一个此类核可被耦合到高速缓存存储器(未示出),在实施例中,该高速缓存存储器可以是共享层级(L2)高速缓存存储器。可使用非易失性存储器930来存储各种程序和其他数据。例如,可以使用该存储设备来存储微代码的至少多个部分、诸如BIOS之类的引导信息、其他系统软件等。

[0073] 每一个核单元910a-910n还可包括诸如总线接口单元之类的接口,其用于实现向处理器的附加电路的互连。在实施例中,每一个核单元910a-910n都耦合到一致性结构,该一致性结构可充当进而耦合到存储器控制器(MC)935的主高速缓存一致性管芯上互连。如在本文中所同样描述的,每个核单元910a-910n可包括相应的核周边逻辑(未在图9中具体示出),该核周边逻辑用于允许核的功率状态对处理器的剩余部分保持透明,以及允许增强的通信并且提供从低功率状态高效的进入和退出,以及其他功能。进而,存储器控制器935控制与诸如DRAM之类的存储器(为了易于说明,在图9中未示出)的通信。

[0074] 除这些核单元之外,附加的处理引擎也存在于该处理器内,包括至少一个图形单元920,该图形单元920可包括用于执行图形处理以及可能在图形处理器上执行通用操作(所谓的GPGPU操作)的一个或多个图形处理单元(GPU)。此外,可以存在至少一个图像信号处理器925。信号处理器925可被配置成处理从一个或多个捕捉设备(无论是SoC内部的还是芯片外的)接收到的传入的图像数据。

[0075] 也可以存在其他加速器。在图9的例示中,视频编码器950可执行编码操作,包括对于视频信息的编码和解码,例如,提供对高清晰度视频内容的硬件加速支持。可进一步提供显示控制器955以加速显示操作,包括提供对系统的内部和外部显示器的支持。此外,可以存在安全处理器945,用于执行诸如安全引导操作、各种加密操作等的安全操作。

[0076] 多个单元中的每一个可以经由功率管理器940来控制其功耗,功率管理器940可以包括用于执行本文中所描述的各种功率管理技术的控制逻辑。

[0077] 在一些实施例中,处理器900还可包括被耦合到一致性结构(各种外围设备可耦合

到该一致性结构)的非一致性结构。一个或多个接口960a-960d允许与一个或多个芯片外设备的通信。此类通信可以经由各种通信协议,诸如,PCIe™、GPIO、USB、I²C、UART、MIPI、SDIO、DDR、SPI、HDMI以及其他类型的通信协议。尽管在图9的实施例中在该高层级示出,但是会理解,本发明的范围不限于该方面。

[0078] 现在参考图10,所示出的是代表性SoC的框图。在所示实施例中,SoC 1000可以是针对低功率操作所配置的多核SoC,该多核SoC待优化以便结合进智能电话或诸如平板计算机或其他便携式计算设备之类的其他低功率设备中。作为示例,可使用诸如较高功率核和/或低功率核(例如,乱序核和有序核)的组合之类的非对称核或不同类型核来实现SoC 1000。在不同的实施例中,这些核可以基于英特尔®架构™核设计或ARM架构设计。在另一些实施例中,可在给定的SoC中实现英特尔核和ARM核的混合。

[0079] 如在图10中所见,SoC 1000包括具有多个第一核1012a-1012d的第一核域1010。在示例中,这些核可以是低功率核,诸如可与相应的核周边逻辑对接的有序核,所述核周边逻辑在实际上核处于给定低功率状态时,允许该核的功率状态看上去为活跃的,如本文所描述的那样。在一个实施例中,可将这些第一核实现为ARM Cortex A53核。进而,这些核耦合到核域1010的高速缓存存储器1015。此外,SoC 1000包括第二核域1020。在图10的例示中,第二核域1020具有多个第二核1022a-1022d。在示例中,这些核可以是比第一核1012a-1012d消耗更高功耗的核。在实施例中,这些第二核可以是可实现为ARM Cortex A57核的乱序核。进而,这些核耦合到核域1020的高速缓存存储器1025。注意,虽然图10中所示的示例在每个域中包括4个核,但是会理解,在其他示例中,更多或更少的核可存在于给定的域中。

[0080] 进一步参考图10,也提供图形域(例如,GPU 1030),该图形域可包括一个或多个图形处理单元(GPU),这些GPU被配置成用于独立地执行例如由核域1010和1020中的一个或多个核所提供的图形工作负荷。作为示例,除了提供图形和显示渲染操作之外,图形域(例如,GPU 1030)可被用于提供对各种屏幕尺寸的显示支持。

[0081] 如所见,各种域耦合至一致性互连1040,在实施例中,该一致性互连1040可以是进而耦合至集成存储器控制器1050的高速缓存一致性互连结构。在一些示例中,一致性互连1040可包括共享高速缓存存储器,诸如,L3高速缓存。在实施例中,存储器控制器1050可以是用于提供与芯片外存储器进行的通信的多个通道(诸如,DRAM的多个通道(在图10中为了易于说明而未示出))的直接存储器控制器。

[0082] 在不同的示例中,核域的数量可以变化。例如,对于适于结合进移动计算设备的低功率SoC,可以存在诸如图10中所示的有限数量的核域。更进一步地,在此类低功率SoC中,包括较高功率核的核域1020可具有较少数量的此类核。例如,在一个实现中,可提供两个核1022以允许以降低的功耗等级进行的操作。此外,也可将不同的核域耦合到中断控制器以启用在不同域之间的工作负荷的动态交换。

[0083] 在另一些实施例中,可以存在更大数量的核域以及附加可选的IP逻辑,如此,可按比例将SoC放大到更高的性能(和功率)水平,以便结合进诸如台式机、服务器、高性能计算系统、基站等的其他计算设备中。作为一个此类示例,可提供各自具有给定数量的乱序核的4个核域。更进一步地,除了可选的GPU支持(作为示例,其可采取GPGPU的形式)之外,还可提供用于对特定功能(例如,web服务、网络处理、交换等)提供优化的硬件支持的一个或多个加速器。此外,可以存在用于将此类加速器耦合到多个芯片外组件的输入/输出接口。

[0084] 现在参考图11,所示出的是另一示例SoC的框图。在图11的实施例中,SoC 1100可包括用于允许多媒体应用、通信和其他功能的高性能的各种电路。由此,SoC 1100适于结合进多种便携式和其他设备,诸如,智能电话、平板计算机、智能电视等。在所示示例中,SoC 1100包括中央处理器单元(CPU)域(例如,CPU 1110)。在实施例中,多个单独的处理器核可存在于CPU域(例如,CPU 1110)中。作为一个示例,CPU域(例如,CPU 1110)可以是具有4个多线程核的四核处理器。此类处理器可以是同构或异构处理器,例如,低功率和高功率处理器核的混合。

[0085] 进而,提供GPU域(例如,GPU 1120)以在一个或多个GPU中执行高级图形处理,从而处理图形并计算API。DSP单元1130可提供一个或多个低功率DSP,这些低功率DSP除了处理可在多媒体指令的执行期间发生的高级计算之外,还处理诸如音乐回放、音频/视频等的低功率多媒体应用。进而,通信单元1140可包括用于经由各种无线协议来提供连接性的各种组件,各种无线协议诸如蜂窝通信(包括3G/4G LTE)、诸如蓝牙™和IEEE 802.11之类的无线局域协议等。

[0086] 更进一步地,可使用多媒体处理器1150来执行对高清晰度视频和音频内容的捕捉和回放,包括对用户姿势的处理。传感器单元1160可包括多个传感器和/或用于对连接到存在于给定平台中的各种芯片外传感器的传感器控制器。可向图像信号处理器(ISP) 1170提供一个或多个分开的ISP,以便参照从平台的一个或多个相机(包括静止相机和视频相机)捕捉到的内容来执行图像处理。

[0087] 显示处理器1180可为向给定像素密度的高清晰度显示器的连接提供支持,包括无线地传递内容以在此类显示器上回放的能力。更进一步地,位置单元1190可包括具有对多个GPS星座图的支持的GPS接收机,以便向应用提供使用此类GPS接收机所获取的高精度定位信息。会理解,虽然在图11的示例中以该特定的一组组件示出,但是许多变型和替代是可能的。

[0088] 现在参考图12,所示出的是可与多个实施例一起使用的示例系统的框图。如所见,系统1200可以是智能电话或其他无线通信器。基带处理器1205被配置成执行关于将从该系统发送或由该系统接收的通信信号的各种信号处理。进而,基带处理器1205被耦合到应用处理器1210,该应用处理器1210可以是系统的主CPU,用于执行除了诸如许多公知的社交媒体与多媒体应用的用户应用之外的OS以及其他系统软件。应用处理器1210可以进一步被配置成为该设备执行各种其他计算操作。

[0089] 进而,应用处理器1210可以耦合到用户接口/显示器1220,例如,触摸屏显示器。此外,应用处理器1210可耦合至包括非易失性存储器(即,闪存1230)和系统存储器(即,动态随机存取存储器(DRAM) 1235)的存储器系统。如进一步所见,应用处理器1210进一步耦合至诸如可记录视频和/或静止图像的一个或多个图像捕捉设备之类的捕捉设备1241。

[0090] 仍然参考图12,包括订户身份模块并且可能包括安全存储设备和加密处理器的通用集成电路卡(UICC) 1240也被耦合到应用处理器1210。系统1200可进一步包括可以耦合到应用处理器1210的安全处理器1250。多个传感器1225可以耦合到应用处理器1210以允许诸如加速度计与其他环境信息之类的各种感测到的信息的输入。音频输出设备1295可提供接口,以便例如以语音通信、播放的或流音频数据等的形式输出声音。

[0091] 如进一步所例示的,经由NFC天线1265在NFC近场中通信的近场通信(NFC)非接触

式接口1260被提供。尽管在图12中示出分离的天线,请理解在一些实现中,可以提供一根天线或不同组的天线以实现各种无线功能。

[0092] 功率管理集成电路 (PMIC) 1215耦合到应用处理器1210以执行平台级别功率管理。为此,PMIC 1215可根据需要将功率管理请求发出至应用处理器1210以使其进入某个低功率状态。此外,基于平台约束,PMIC 1215还可控制系统1200的其他组件的功率级别。

[0093] 为了允许发送与接收通信,可以在基带处理器1205与天线1290之间耦合各种电路。具体而言,可以存在射频 (RF) 收发机1270与无线局域网 (WLAN) 收发机1275。一般而言,可以根据诸如3G或4G无线通信协议 (诸如根据码分多址 (CDMA)、全球移动通信系统 (GSM)、长期演进 (LTE) 或其他协议) 之类的给定的无线通信协议,使用RF收发机1270接收并发送无线数据和呼叫。此外,可以存在GPS传感器1280。也可以提供诸如无线电信号 (例如,AM/FM与其他信号) 的接收与发送之类的其他无线通信。此外,也可以经由WLAN收发机1275实现本地无线通信。

[0094] 现在参考图13,所示出的是可与多个实施例一起使用的另一示例系统的框图。在图13的示图中,系统1300可以是诸如平板电脑、2:1平板、平板手机或其他转换式或独立式平板系统之类的移动低功率系统。如所例示的,SoC 1310存在并且可配置成作为设备的应用处理器进行操作。

[0095] 各种设备可耦合至SoC 1310。在所示的例示中,存储器子系统包括被耦合至SoC 1310的闪存1340和DRAM 1345。此外,触摸面板1320被耦合至SoC 1310以提供显示能力和经由触摸的用户输入,包括在触摸面板1320的显示器上提供虚拟键盘。为了提供有线网络连接性,SoC 1310耦合至以太网接口1330。外围中枢1325被耦合至SoC 1310以允许与各种外围设备的对接,诸如,各种外围设备可通过各种端口或其他连接器中的任何一个而被耦合到系统1300。

[0096] 除了SoC 1310中的内部功率管理电路和功能之外,PMIC 1380还被耦合至SoC 1310以提供基于平台的功率管理,例如,基于该系统是由电池1390供电还是经由AC适配器1395由AC功率供电。除了该基于功率源的功率管理之外,PMIC 1380还可执行基于环境和和使用条件的平台功率管理活动。更进一步地,PMIC 1380可将控制和状态信息传达给SoC 1310以引起SoC 1310中的各种功率管理动作。

[0097] 仍然参考图13,为了提供无线能力,WLAN单元1350被耦合至SoC 1310,并进而耦合至天线1355。在各种实现中,WLAN单元1350可以根据一个或多个无线协议提供通信。

[0098] 如进一步所例示的那样,多个传感器1360可耦合至SoC 1310。这些传感器可包括各种加速度计、环境和其他传感器,包括用户姿势传感器。最后,音频编解码器1365被耦合至SoC 1310以向音频输出设备1370提供接口。当然会理解的是,虽然以图13中的该特定实现示出,但是许多变型和替代是可能的。

[0099] 现在参考图14,所示出的是诸如笔记本、超极本™或其他小形状因子系统的代表性计算机系统的框图。在一个实施例中,处理器1410包括微处理器、多核处理器、多线程处理器、超低电压处理器、嵌入式处理器,或其他已知处理元件。在所示出的实现中,处理器1410充当主处理单元以及用于与系统1400的各种组件中的许多组件进行通信的中央枢纽。作为一个示例,处理器1400被实现为SoC。

[0100] 在一个实施例中,处理器1410与系统存储器 (例如,LPDDR3 1415) 通信。作为示例

性示例,系统存储器被实现为经由多个存储器设备或模块来提供给定量的系统存储器。

[0101] 为提供对诸如数据、应用、一个或多个操作系统等之类的信息的持续存储,大容量存储设备(例如,SSD或HDD 1420)还可耦合到处理器1410。在各种实施例中,为了允许更薄和更轻的系统设计以及为了改善系统响应,该大容量存储设备可以经由SSD被实现,或者该大容量存储设备可以主要使用具有较小量的SSD存储设备的硬盘驱动器(HDD)来实现,SSD存储设备充当SSD高速缓存,以在掉电事件期间实现对上下文状态和其他此类信息的非易失性存储,从而在系统活动重新启动时,快速上电可以发生。图14还示出,闪存设备(例如,BIOS、FW闪存1422)可以例如经由串行外围接口(SPI)而被耦合到处理器1410。该闪存设备可以提供对系统软件的非易失性存储,所述系统软件包括基本输入/输出软件(BIOS)以及系统的其他固件(FW)。

[0102] 系统1400内可以存在各种输入/输出(I/O)设备。在图14的实施例中具体示出的是显示器1424,显示器1424可以是进一步提供触摸屏1425的高清晰度LCD或LED面板。在一个实施例中,显示器1424可以经由显示互连而被耦合到处理器1410,该显示互连可实现为高性能图形互连。触摸屏1425可以经由另一互连而被耦合到处理器1410,在实施例中,该另一互连可以是I²C互连。如在图14中进一步所示,除触摸屏1425之外,经由触摸的用户输入也可以经由触板1430进行,触板1430可被配置在机架内,并且还可被耦合到与触摸屏1425相同的I²C互连。

[0103] 出于感知计算和其他目的,各种传感器可以存在于系统内,并且能以不同的方式被耦合到处理器1410。某些惯性传感器和环境传感器可例如,经由I²C互连,通过传感器中枢1440耦合到处理器1410。在图14所示出的实施例中,这些传感器可以包括加速度计1441、环境光传感器(ALS) 1442、罗盘1443以及陀螺仪1444。其他环境传感器可以包括一个或多个热传感器1446,在一些实施例中,这些热传感器1446经由系统管理总线(SMBus)总线耦合到处理器1410。

[0104] 图14中还可见,各种外围设备可以经由低引脚计数(LPC)互连耦合到处理器1410。在所示的实施例中,各种组件可以通过嵌入式控制器(EC) 1435而被耦合。此类组件可包括键盘1436(例如,经由PS2接口耦合)、风扇1437以及热传感器1439。在一些实施例中,触板1430还可以经由PS2接口耦合到EC 1435。此外,诸如可信任平台模块(TPM) 1438之类的安全处理器也可经由该LPC互连耦合到处理器1410。

[0105] 系统1400能以各种方式(包括以无线方式)与外部设备通信。在图14所示出的实施例中,存在各种无线模块,其中的每一个无线模块都可以对应于针对特定的无线通信协议而配置的无线电装置。用于在短距离(诸如,近场)内进行无线通信的一种方式可以经由NFC单元1445,在一个实施例中,该NFC单元1445可以经由SMBus与处理器1410进行通信。注意,经由该NFC单元1445,彼此紧邻的设备可以进行通信。

[0106] 如图14中进一步所见,附加的无线单元可包括其他短距离无线引擎,包括WLAN单元1450和蓝牙单元1452。通过使用WLAN单元1450,可以实现Wi-Fi™通信,而通过蓝牙单元1452,可以发生短距离蓝牙™通信。这些单元可以经由给定的链路与处理器1410通信。

[0107] 另外,无线广域通信(例如,根据蜂窝式或其他无线广域协议)可以经由WWAN单元1456进行,该WWAN单元1456进而可以耦合到订户身份模块(SIM) 1457。另外,为了允许接收并使用位置信息,还可以存在GPS模块1455。注意,在图14中示出的实施例中,WWAN单元1456

和诸如相机模块(例如,USB3.0相机1454)的集成捕捉设备可以经由给定的链路进行通信。

[0108] 集成的相机模块(例如,USB3.0相机1454)可以被纳入到盖子中。为了提供音频输入和输出,可以经由数字信号处理器(DSP)1460来实现音频处理器,DSP 1460可以经由高清晰度音频(HDA)链路来耦合到处理器1410。类似地,DSP 1460可以与集成的编码器/解码器(CODEC)和放大器(例如,音频编解码器和D类放大器1462)进行通信,集成的编解码器和放大器(例如,音频编解码器和D类放大器1462)进而可以耦合到可以在机架内实现的输出扬声器1463。类似地,放大器和CODEC(例如,音频编解码器和D类放大器1462)可被耦合到话筒(MIC)1465以从话筒1465接收音频输入,在实施例中,话筒1465可以经由双阵列话筒(诸如数字话筒阵列)被实现,以提供高质量音频输入来实现对系统内的各种操作的语音激活的控制。另外注意,可以将音频输出从放大器/CODEC(例如,音频编解码器和D类放大器1462)提供到头戴式耳机1464。虽然在图14的实施例中利用这些特定组件示出,但应理解本发明的范围不限于此方面。

[0109] 实施例可在许多不同的系统类型中实现。现在参考图15,所示出的是根据本发明的实施例的系统的框图。如图15所示,多处理器系统1500是点对点互连系统,且包括经由点对点互连1550耦合的第一处理器1570和第二处理器1580。如图15所示,处理器1570和1580中的每一个可以是包括第一和第二处理器核(即处理器核1574a和1574b以及处理器核1584a和1584b)的多核处理器,尽管这些处理器中可能存在更多核。出于低功率状态进入和退出的目的以及出于在正常操作期间信息的通信的目的两者,此类处理器核可耦合到相应的核周边逻辑1577a和1577b以及核周边逻辑1587a和1587b以提供关于相应处理器核的功率状态的透明性,并且允许上下文和其他信息的高效通信。处理器中的每一个可以包括PCU或其他功率管理逻辑以执行本文所述的基于处理器的功率管理。

[0110] 仍参考图15,第一处理器1570进一步包括存储器控制器中枢(MCH)1572和点对点(P-P)接口1576和1578。类似地,第二处理器1580包括MCH 1582和P-P接口1586与1588。如图15所示,MCH 1572和1582将处理器耦合到相应的存储器(即,存储器1532和存储器1534),这些存储器可以是本地附连到相应处理器的系统存储器(例如,DRAM)的部分。第一处理器1570与第二处理器1580可以分别经由P-P互连1562与1564而被耦合到芯片组1590。如图15中所示,芯片组1590包括P-P接口1594和1598。

[0111] 此外,芯片组1590包括用于通过P-P互连1539将芯片组1590与高性能图形引擎1538耦合的接口1592。进而,芯片组1590可以经由接口1596被耦合到第一总线1516。如图15所示,各种输入/输出(I/O)设备1514以及总线桥接器1518可被耦合到第一总线1516,总线桥接器1518将第一总线1516耦合到第二总线1520。在一个实施例中,各种设备可被耦合到第二总线1520,包括例如,键盘/鼠标1522、通信设备1526以及诸如可包括代码1530的盘驱动器或者其他大容量存储设备之类的数据存储单元1528。此外,音频I/O 1524可以被耦合到第二总线1520。可将多个实施例合并到其他类型的系统中,包括诸如智能蜂窝电话、平板计算机、上网本、超极本™等之类的移动设备。

[0112] 现在参考图16,所示出的是根据本发明的实施例的处理器框图。如图16所示,处理器1600包括核1610以及各个核周边逻辑。应当理解,为了便于说明,仅单个核1610被示出。然而,在许多实施例中,多核处理器包括多个核,每一个都具有它自己的核周边逻辑。在图16所示出的高层级示图中,处理器1600的组件全部在单个半导体管芯上被实现。如所见,

核1610包括存储(例如,SRAM 1615),在实施例中,存储1615可以是静态随机存取存储器(SRAM),核的各种上下文或状态信息都被存储在其中。注意,术语“状态信息”以及“上下文信息”在本文中被互换地使用,并且指代诸如控制寄存器值、数据信息、寄存器所存储的信息以及与在核或其他逻辑上执行的线程相关联的其他信息之类的信息。当相应的线程例如由于进入低功率状态或迁移到另一个核而从该核被切换出时,此类信息可被保存。

[0113] 在实施例中,存储1615可被配置成在核处于某个低功率状态时保持通电。作为示例,存储1615在核处于给定低功率状态(例如,C6)并且处理器封装处于封装活跃状态(C0)时可维持信息。然而,在其他低功率状态中,此类功率可能不是可用的,并且上下文信息可被发送到如本文所描述的其他存储。如在本文中将描述的,考虑到在从C6状态退出时以及在某个机器校验架构逻辑可用之前错误会发生在SRAM 1615中,可提供MCA报告寄存器1617以记录在SRAM 1615中发生的不可纠正的错误情形。

[0114] 核1610进一步包括用来与IDI 1670对接的管芯内互连(IDI)接口1618。虽然为了易于说明而未被示出,应该理解,IDI 1670可将核1610与处理器中的各种其他电路(为了易于说明而未在图16中示出)耦合,包括一个或多个其他核,外围控制器中枢(PCH)、一个或多个高速缓存存储器和/或其他非核电路。为了在核1610与能以不同频率操作的处理器内的其他组件之间提供接口,可提供时钟交叉逻辑(例如BGF 1619),在实施例中,时钟交叉逻辑(例如BGF 1619)可被实现为气泡发生器先进先出(FIFO)缓冲器。可提供反压控制的该时钟交叉逻辑的细节在下文进一步被描述。

[0115] 为了使核1610在可用时能够进入特定的以及更深的低功率状态,第一核周边逻辑,即结构接口逻辑(FIL) 1620被耦合到核1610。FIL 1620可以属于第一维持功率域,因为在处理器的至少多个部分处于低功率状态时,向FIL 1620提供功率和时钟信号。如所见,FIL 1620经由IDI 1670和第二互连(例如,CRI 1675)两者耦合至核1610,在实施例中,第二互连是控制寄存器互连(CRI)。互连(例如,CRI 1675)可以是相对简单并且低性能的互连,用于在低功率状态进入和退出的保存和恢复操作期间提供状态信息的通信。

[0116] 在图16所示的实施例中,FIL 1620包括存储1622,存储1622可被实现为被配置成存储在核进入给定低功率状态之前从核1610接收到的状态信息的多个寄存器。可维持到FIL 1620的功率,直到在一致性结构进入低功率状态时,处理器封装进入更深的封装低功率状态(例如,封装C6状态)。如进一步示出的,FIL 1620包括监测器逻辑1624、中断控制逻辑1625以及监听响应逻辑1626。一般来说,监测逻辑1624可被配置为,在核1610处于低功率状态时,监测一个或多个被监测位置以对存储在其中的值进行更新。经过该更新,FIL 1620可将唤醒请求传达到核1610。在实施例中,监测逻辑1624可因此被配置为在核1610处于低功率状态时实现MONITOR/MWAIT操作。进而,中断控制逻辑1625可被配置为在核1610处于低功率状态时处理传入中断。此类处理可包括推迟该中断和/或对该中断发送响应。更进一步地,在一些情形中,该处理可包括唤醒核1610来处理该中断。

[0117] 同样,FIL 1620包括监听响应逻辑1626,监听响应逻辑1626可被配置为响应于在核1610处于低功率状态时传入的监听请求的接收来发送监听响应。也就是说,因为在核处于低功率状态时不存在用于监听请求的相应的高速缓存行,监听响应逻辑1626因此可发送响应来指示核1610不包括与监听请求相关联的高速缓存行的副本。关于监听响应逻辑1626的细节会在下文进一步描述。

[0118] 如图16进一步所示,FIL 1620还包括锁定响应逻辑1627。一般来说,锁定响应逻辑1627可被配置为在核1610处于给定低功率状态(例如,C6或更深的低功率状态)时处理锁定流程。更具体地,可在核1610处于C6或更深的低功率状态时从非核或其他SoC位置接收锁定静默消息。可在SoC的给定IP块试图执行诸如读取—修改—写入(RMW)流程之类的原子操作时接收此类锁定静默消息。典型地,在此类RMW流程中,所有其他IP块在原子操作的持续时间内被阻止发送任何事务,并且进一步要确认该IP块不具有任何运行中的事务。如此,响应于锁定静默消息的接收,如果这些条件都被满足,那么其他IP块发送确认消息来指示该IP块不具有任何运行中事务并且在原子流程期间将不会发送任何事务。当从所有相关IP块接收到有效确认时,则允许请求IP块执行原子操作。

[0119] 然而,当给定核处于深低功率状态时,它不能对传入的锁定静默消息作出响应。在这种情形下,锁定响应逻辑1627可生成对该消息的默认响应。更具体地,锁定响应逻辑1627可被配置为提供默认响应来指示或确认核已经接收到该消息并且将不发送任何事务。更进一步地,锁定响应逻辑1627可在原子操作结束之际接收锁定释放消息并处理该消息。

[0120] 注意,在核1610于原子操作的中途被唤醒的情形中,锁定响应逻辑1627可将此类活跃的原子流程的存在传达给该核以阻止该核发送任何事务直到接收到锁定释放消息。如此,锁定响应逻辑1627被配置为在核处于低功率状态时处理对传入锁定消息的责任,并且如果核在原子操作期间被唤醒,则负责正确地遵守正在进行的原子操作。

[0121] 如进一步所示,FIL 1620还包括信用逻辑1628。更具体地,信用逻辑1628可在核1610处于特定低功率状态(例如,C6或更深低功率状态)时被用于执行信用管理活动。在实施例中,信用逻辑1628可被配置为在核处于此类低功率状态时处理信用交换。作为示例,当信用通过路径被返回到核1610但核1610处于此类低功率状态时,信用逻辑1628可通过提供一组阴影信用寄存器或其他存储来维持并更新与各信息通道相关联的信用值来容纳该信用更新。在一个实施例中,可存在用于各种通道的信用计数器,包括用于传入和传出方向两者中的请求、响应以及数据中的每一个的单独的计数器。然后,当核1610进入活跃状态时,信用更新进程可在信用逻辑1628与核1610之间发生,使得该核的相应的信用逻辑(例如,包括一组信用计数器,每个计数器与特定信息通道相关联)可被适当地更新。注意,在该核进入活跃状态并接收对各信用值的更新时,该核还对信用活动负责。

[0122] 仍旧参考图16,附加的核周边逻辑是机架适配器块(CAB)单元1630。一般来说,CAB单元1630可被配置为经由边带互连1690来向其他处理器和系统组件提供接口。更进一步地,CAB单元1630可被配置为在FIL 1620本身被置于低功率状态时存储核1610的状态信息。CAB单元1630可以是第二维持功率域,因为在处理器1600的其他部分(包括FIL 1620)处于低功率状态时,向CAB单元1630提供功率和时钟信号。CAB单元1630包括存储1632,该存储1632被配置为存储从FIL 1620获得的状态信息。在实施例中,CAB单元1630的存储1632可以是快速存储阵列,例如,被实现为SRAM。

[0123] 在所示的实施例中,CAB单元1630包括功率管理代理(PMA)1634,可包括一个或多个有限状态机(FSM)的熔丝拔出器逻辑,一个或多个有限状态机用于在CAB单元1630本身被置于低功率状态时来对存储1622和存储器层次结构的更远部分(例如,系统存储器)两者执行保存和恢复操作。例如,存储在存储1622中的信息可在处理器封装进入更大程度的更深封装低功率状态(例如,封装C10状态)时被转储清除至系统存储器。在实施例中,这些FSM可

以是基于SoC的FSM,因为它们允许核周边逻辑与SoC的其他部分(以及存储器层次结构的更远部分)之间的交互。注意,PMA 1634可以是在CAB单元1630通电时可处于活跃状态的处理器的功率管理逻辑的部分。在一些情形中,PMA 1634可与诸如PCU之类的处理器的主功率控制器或其他功率管理实体对接。

[0124] CAB单元1630进一步包括事件阻塞逻辑1638,该事件阻塞逻辑1638可被配置为在处理器处于特定低功率状态时阻止传入事件。更进一步地,CAB单元1630还包括边带接口1639,边带接口1639可与边带互连1690对接。注意,在各实施例中,IDI 1670是用于适应高速运行的SoC的各个核与其他IP块之间的主要(并且常常是性能关键的)通信的宽高速互连。替代地,边带互连1690可被配置为经由更窄通道和更低操作频率的使用来用于较低性能关键的通信。

[0125] 在实施例中,事件阻塞逻辑1638可在核处于C6或其他深低功率状态时阻止传入边带事件。一般来说,事件阻塞逻辑1638可存储此类传入事件,例如,低性能中断或其他事件。基于这些传入事件的接收,事件阻塞逻辑1638可向PMA 1634发信号以允许PMA决定是否唤醒核1610来处理传入事件。经过核1610的唤醒,可从事件阻塞逻辑1638将任何存储的事件推送到核1610。

[0126] 在实施例中,可允许CAB单元1630的存储1632被PMA 1634访问或被经由边带接口1639接收的经验证的访问来访问。在一个此类实施例中,该接口可包括安全属性标识符(SAI)逻辑,该安全属性标识符逻辑用于确定对存储1632的访问请求是否具有有效的SAI安全保护(例如,与该请求一起接收的SAI值和与要访问的存储位置相关联的SAI值相匹配)。如此,存储1632可被保护以存储敏感内容。

[0127] 在实施例中,适当的时钟逻辑可被应用于各个核周边逻辑以允许其中的存储和逻辑在特定的低功率状态下被访问。在实施例中,双时钟逻辑可被应用于维持功率域中的存储。作为一个示例,高速缓存一致性结构(CCF)时钟可被提供到用于标准读/写操作的存储。进而,CRI时钟可被提供到用于保存/恢复操作的存储。

[0128] 要理解,处理器可包括附加的组件和电路。在图16的例示中,处理器1600进一步包括功率递送单元1640,在实施例中,该功率递送单元1640可包括一个或多个完全集成电压调节器;时钟电路1650,在实施例中,该时钟电路1650可被实现为锁相环;以及数字热传感器1660。如所见,这些组件中的每一个可经由互连(例如,CRI 1675)与处理器1600的其他组件通信。请理解,尽管在图16中以此特定处理器实现示出,但是许多变型和替代是可能的。

[0129] 现在参考图17,所示出的是根据实施例的监听响应逻辑的进一步细节的框图。如图17所示,处理器1600可采取如以上在图16中所描述的类似的形式。无论如何,关于监听响应逻辑1626的进一步的细节被示出。如所见,该监听响应逻辑位于FIL 1620内,被配置于核1610与处理器的非核部分(为了易于说明而未在图17中示出,但它们可经由管芯间互连而被连接到FIL 1620)之间。在所示出的特定实现中,时钟交叉缓冲器1705a和1705b提供用于传入非核到核请求(U2C请求)以及传出核到非核响应(C2U响应)的接口。

[0130] 对于在核1610处于活跃状态时经由IDI接收到的传入监听请求,此类请求作为IDI合格监听请求通过被实现为AND门的逻辑门1710(经由BGF 1619)到达核1610。如此,取决于核1610中与该请求相关联的数据的状态(例如,存在或不存在以及高速缓存一致性状态(如果存在)),相应的核响应经由复用器1730,经由C2U响应通道而被提供到非核。

[0131] 然而,在核1610处于特定低功率状态(例如,C6或更深低功率状态)的情形中,经确定在该核内没有处于有效状态的数据存在,如此,可提供默认监听响应。更进一步地,可提供该响应而不会导致核1610本身从给定低功率状态被唤醒。

[0132] 监听响应逻辑1626被配置为响应于从PMA 1634接收的低功率状态信号(核处于C6状态的信号)来实现该操作。当该信号是活跃的时,逆变器1708使AND门不将监听请求传递给核1610。取而代之的是逻辑门1715(被实现为另一个AND门)被激活以成为到SLR响应队列1720中的活跃的输入。更具体地,可用标识该监听请求的源的相应的源标识符来为该接收到的监听请求填充条目。当该条目通过队列1720时,它作为默认响应(即,所请求的数据不存在,诸如作为无效的监听响应)而被输出到复用器1730。在此种情况下,复用器1730进而可(经由被实现为另一AND门的逻辑门1735)被控制以将该默认响应传递给非核。如此,由PMA 1634输出的C6状态信号中的核的水平控制逻辑门1710、1715和1735的状态,以在核处于至少阈值低功率状态时引起默认监听响应,或者在核处于活跃状态(或处于相对浅低功率状态)时引起核控制监听响应。要理解,虽然在图17的实施例中以该高层级示出,但许多变型和替换方案是可能的。

[0133] 现在参考图18,所示出的是根据实施例的内部处理器存储器的高层级框图。更具体地,图18示出了SRAM 1800的框图,SRAM 1800可以是用于实现对数据本地、快速访问的处理器的一组内部存储器。如所见,存在多个高速缓存组 1810_0-1810_n 。要理解,虽然在图例中用八个此类组来示出,但更多或更少组可存在于特定实施例中。此外,提供冗余块1820以用于冗余。要理解,在低功率模式下,存储在高速缓存组 1810_0-1810_n (以及冗余块1820)中的信息可被转储清除,并且这些组在至少特定低功率状态期间不通电。

[0134] 然而,实施例提供附加的保持组 1830_0-1830_m 。在各实施例中,组 1830_0-1830_m 可持续地提供数据的存储,即便在给定核处于特定低功率状态(例如,C6或甚至更深的低功率状态)时。在一个此类实施例中,保持组 1830_0-1830_m 可保持通电以在C6状态期间在高速缓存内部本地地保持状态信息。虽然本发明的范围不限于此方面,但在一些情形中,只有与在核上执行的一个或多个线程相关联的某个上下文信息和核配置信息可在进入给定低功率状态之前被存储在保持组 1830_0-1830_m 中。通过在核内部的SRAM内提供这些保持组,实现了改善的C6保存/恢复等待时间。同样要理解,虽然用其中保持组 1830_0-1830_m 在物理上位于该存储器外部的特定拓扑来示出,但是其他位置是可能的。

[0135] 如在图18中进一步示出的,功率复用器1840可提供对这些组的传入电压的选择。例如,当处于活跃状态时,可将活跃的电压水平提供给所有的组。反之当处于低功率状态(例如,C6)时,可将保持电压提供给保持组 1830_0-1830_m ,同时不向高速缓存组1810提供电压。

[0136] 通过提供核内部的C6 SRAM存储,在恢复操作期间,当存储在保持组1840中的状态信息被恢复成处理器状态时,MCA错误会在不可纠正的错误在读出期间发生时发生。例如,假设在纠错码(ECC)存储器中已确定存在不可纠正的错误,例如,由于所读取的给定存储器块内的多位错误而引起。这种错误会导致MCA错误。然而,在此恢复操作期间,处理器尚未被配置为用于MCA操作,因为存储在保持组1840中的核的各种状态信息还尚未被恢复成核的配置。如此,在读取存储在保持组1640中的状态信息完成之前,该核没有被配置为用于适当的MCA操作。

[0137] 替代地,实施例提供MCA报告寄存器1627(在图16中被示出)。该寄存器可记录在C6恢复操作期间发生的此类MCA错误。在恢复流程期间,微代码可恢复(被实现为MCA架构的)MCA逻辑的MCA组。这些组可被实现为每个核子域的一组寄存器。在此恢复操作期间,MCA数据被重写。因此为了实现对该错误的正确标识以及对MCA错误的触发,MCA逻辑可读取报告寄存器以确定一个或多个MCA错误的存在并且采取诸如引起MCA错误、使处理器重置或其他操作之类的适当的动作来解决该错误。要理解,虽然在图18的实施例中以该高层级示出,但许多变型和替换方案是可能的。

[0138] 如以上所描述的,处理器可具有多个域,每个域以给定的时钟频率来操作。例如,写入域(例如,处理器的非核)可以写入时钟频率来运行而读取域(例如,处理器的核)可以读取时钟频率来操作。在一些情形中,可控制一个或另一个域以使其与其他域处于不同的功率状态。例如,核域可在非核域保持活跃时被置于低功率状态。或者,操作系统可,例如,为了保存功率而决定暂时关闭一个域以改变该域的时钟频率,同时处理器的另一个域保持活跃。在写入域保持活跃时,读取域可关闭以便于具有其调整后的时钟频率。例如,读取域可被关闭以便转换到活动减少(例如,更慢的时钟频率)并且功耗减少的状态,例如,若干活动减少的P状态中的一个。

[0139] 根据实施例,反压方案可在处理器的写入域与读取域之间被实现。反压方案可允许在读取域被关闭的时间段期间对写入域内的缓冲器(例如,有效载荷气泡发生器先进先出缓冲器(有效载荷BGF))的写入。在读取域关闭期间,没有存储在有效载荷BGF中的分组被读取域读取。经过读取域的重启,存储在写入域的有效载荷BGF中的分组可被读取域读取。

[0140] 反压机机制包括用于维持写入信用计数的写入器域中的写入源信用逻辑。最初,从例如,读取域内的BGF信用逻辑(BGF信用计数器)将固定数目的写入信用提供给写入源信用逻辑。每个写入信用使写入源能够将数据分组写入到有效载荷BGF,不论读取域是活跃还是关闭。经过对有效载荷BGF的每次写入,写入源信用逻辑中的写入源信用计数器被递减。如果写入源信用计数器指示在写入域中没有写入信用可用,那么对有效载荷BGF的写入被禁止。随着有效载荷BGF的每次写入,额外的写入信用被BGF信用逻辑提供给写入源信用逻辑。写入信用的使用阻止有效载荷BGF内所存储分组的重写。

[0141] 在以下的讨论中,值“真”由数值“1”表示,而值“假”由数值“0”表示。在其他实施例中,“真”可由“0”表示而“假”可由“1”表示而不失关于所提概念的一般性。

[0142] 每个域可具有生成门控信号的相关联的气泡发生器(BG)。也就是说,写入域中的写入BG生成写入信号(时隙有效写入(TSVW)),该写入信号在写入时钟具有值1的各种情况下来门控对有效载荷BGF的写入。当写入BG和写入时钟两者都具有值“1”时,并且当写入源具有用于写入到有效载荷的分组时,写入可发生。当写入BG或写入时钟中的任一个具有值“0”时,将不会发生对有效载荷BGF的写入。在实施例中,对有效载荷BGF的写入进一步被写入源信用门控,例如,如果在写入域中没有写入源信用可用(如由写入源信用逻辑来确定),那么没有写入将会发生。如果写入源信用是可用的,那么基于对有效载荷BGF的写入的执行,写入源信用的计数递减一。

[0143] 读取域具有读取BG,该读取BG与读取时钟组合在一起来门控每次读取。也就是说,读取域中的读取BG生成读取信号(时隙有效读取(TSVR)),该读取信号在读取时钟具有值“1”(例如,“真”)的各种情况下来门控是否允许(例如,从有效载荷BGF)读取。当读取BG输出

“1”并且读取时钟输出“1”时,允许对有效载荷BGF的读取(例如,在“读取窗口”期间允许读取)。如果读取BG或读取时钟中的任一个具有值“0”,那么将不会发生读取。在一些实施例中,读取信用的系统可被用于进一步门控是否允许对有效载荷BGF的读取,例如,如果至少一个读取信用是可用的,则允许对有效载荷BGF的读取。读取信用可由读取域中的读取器信用发出逻辑来发出,并且读取信用的计数可由读取域中的出口(读取器)信用计数器来维持。读取BG具有读取BG频率(TSVR频率)而写入BG具有写入BG频率(TSVW频率)。在实施例中,读取BG频率和写入BG频率被设置成使得,对每个基本时钟周期,都将存在相等数量的写入机会和读取机会,以便于维持从写入域到读取域的分组的稳定的状态流。在实施例中,读取BG与写入BG之间的关系如以下所述:具有更慢的域时钟频率的域可被设置为具有BG值(时隙有效(TSV))‘1’(例如,常‘真’),而对于具有更快域时钟频率的域来说,BG频率被设置为与域频率的比值(例如,更快的域频率/更慢的域频率)成比例。该比例方案在每个基本时钟周期内允许相等数量的读取机会(“读取窗口”)和写入机会(“写入窗口”)。

[0144] 替代地,在实施例中,写入域具有有效BG FIFO缓冲器(有效BGF),该有效BGF包括有效FIFO缓冲器(本文中的“有效FIFO”),用于在写入BG具有值“1”时为每个写入域时钟周期存储值(“有效指示符”)。对于给定的写入域时钟周期,如果写入源将分组写入到有效载荷BGF,那么有效指示符具有值‘1’并且被存储在有效FIFO中。如果写入源没有将分组写入到有效载荷BGF,那么有效指示符具有值‘0’并且被存储在有效FIFO中。当写入BG具有值‘0’时,没有值会被存储到有效FIFO中。

[0145] 在实施例中,在读取域关闭期间,读取BG和写入BG两者都被关闭,例如,TSVW和TSVR维持恒定值‘0’。因此,对有效载荷BGF的读取是不可能的。虽然没有进行对有效FIFO的写入,但是写入源可继续将分组写入到有效载荷BGF。当写入BG被关闭时,有效BGF中的写入累加器为对有效载荷BGF的每一次写入存储(例如,累加)有效指示符(例如,每一个有效指示符具有值‘1’)。

[0146] TSV通过重置BG而被停止。在P状态转换结束而新的时钟频率在读取和写入侧的一者(或两者)中被建立之后,TSVR和TSVW可被重启。有效BGF和有效载荷BGF将在P状态转换关闭之前从与由它们的相应指针指示的位置相同的相应位置开始操作。

[0147] 图19是根据本发明的实施例的处理器1900的框图。处理器1900包括时钟域A 1902,该时钟域A 1902以时钟频率A操作,以及时钟域B 1904,该时钟域B 1904以时钟频率B操作。时钟域A 1902包括气泡发生器(BG) 1912、写入源信用逻辑1910、写入源逻辑1914、有效载荷气泡发生器先进先出缓冲器(有效载荷BGF) 1922以及有效BGF 1920,其中有效载荷BGF 1922包括有效载荷写入指针1919和有效载荷先进先出缓冲器(有效载荷FIFO) 1921,有效BGF 1920包括有效写入指针1917和有效FIFO 1918。时钟域B 1904包括气泡发生器(BG) 1940、有效读取指针1932、有效载荷读取指针1946、读取mux(复用器) 1950、读取mux 1952以及BGF信用逻辑1960。处理器1900还包括一个或多个核(未示出)、功率控制单元(未示出),并且可包括其他硬件、软件和/或固件逻辑(未示出)、高速缓存等等。

[0148] 在操作中,从例如PCU(未示出)接收的BGF运行信号启动气泡发生器(BG) 1912和BG 1940。对于写入时钟的每个‘真’值,写入域时钟允许写入源逻辑1914将数据分组(如果存在要被写入的数据分组)写入到有效载荷FIFO 1921中由有效载荷写入指针1919所指示的位置,有效载荷写入指针1919随每次写入递增其指针地址。

[0149] 写入气泡发生器1912将提供时隙有效写入 (TSVW) 信号。写入窗口在TSVW信号和写入时钟都为‘真’，例如具有值‘1’时出现。在每个写入窗口出现时，写入源逻辑1914将相应的有效指示符发送到有效BGF 1920。每个有效指示符具有指示数据分组是否将被写入到有效载荷FIFO 1921的值，例如，有效指示符具有值‘1’，用于指示在写入窗口期间数据分组正被写入到有效载荷FIFO 1921，或者值‘0’，用于指示在写入窗口期间没有数据分组正被写入到有效载荷FIFO 1921。如果写入气泡发生器1912 (例如，在时钟域B 1904关闭期间) 被关闭，则写入源逻辑1914可仍旧将数据分组写入到有效载荷FIFO (每一次写入都由写入时钟来门控)，并且被写入有效载荷FIFO的每个数据分组的指示会被存储在累加器1916中。

[0150] 气泡发生器1912和气泡发生器1940被配置为产生时隙有效信号 (分别为TSVW和TSVR)，并且它们可被配置为门控它们各自所在域中的活动，使得写入和读取在每一个基本时钟周期内都以相等的比率发生。例如，在实施例，时钟域A 1902以时钟频率A操作，时钟频率A比时钟域B 1904的时钟频率B要快。具有更慢时钟频率的时钟域B可被配置为产生持续有效的TSV信号 (值‘1’)，使得时钟频率B门控域B中的动作的执行 (例如，对有效载荷FIFO 1921的读取)。(具有更快时钟频率的)时钟域A产生门控动作的TSV信号并且，与时钟频率A耦合使得能够以与域时钟频率B相同的速率来执行动作。

[0151] 作为示例，处理器具有100MHz的基本时钟频率，时钟域B具有300MHz的时钟频率B，而时钟域A具有500MHz的时钟频率A。在一个基本时钟周期中，时钟域B具有3个时钟周期而时钟域A具有5个时钟周期。气泡发生器1940可产生持续信号，使得在一个基本时钟周期期间，3个时钟周期中的每一个都门控时钟域B内的动作。气泡发生器1912可产生TSVW，该TSVW与时钟频率A的5个时钟周期结合，同样门控时钟域A中的3个时钟周期 (例如，在一个基本时钟周期内，TSVW在五个频率A时钟周期中的两个期间具有‘假’值而在五个频率A时钟周期中的三个期间具有‘真’值)。通过选择具有更高时钟域频率的域适当的的气泡发生器频率，可使两个域在每个基本时钟周期内都以相同的速率来执行读取/写入。

[0152] 在实施例，当气泡发生器1912在操作时，有效写入指针1917和有效读取指针1932将从保持恒定的指针间隔N开始，例如，以防止写入后读取不稳定性和读取后写入不稳定性两者。有效写入指针1917指向有效FIFO 1918内的当前位置，该当前位置将存储指示有效载荷FIFO 1921内的相应位置是否存储了数据分组的有效指示符。在数据分组将被存储在有效载荷FIFO 1921中的按顺序的下一个位置中时，有效指示符的值为‘1’。当在当前写入窗口期间没有数据分组将被写入到有效载荷FIFO 1921时，有效指示符被写为‘0’。例如，当将数据分组写入有效载荷FIFO 1921发生时，有效指示符具有值‘1’并且被写入以对应于有效载荷FIFO 1921中的存储该数据分组的当前位置。经过每个写入窗口，有效写入指针1917都增加。有效载荷写入指针1919响应于对有效载荷FIFO 1921的写入将发生的指示而增加。有效写入指针1917和有效读取指针1932维持着对有效载荷FIFO 1921的写入和读取之间的指针间隔。

[0153] 写入源信用逻辑1910存储写入信用和/或维持接收到的写入信用的计数。对于由写入源逻辑1914进行的对有效载荷FIFO 1921的每一次写入，写入信用被使用并且写入信用的计数递减。

[0154] 有效BGF 1920包括累加器1916，累加器1916输入到有效FIFO 1918并且使有效指示符的值 (例如，‘1’) 能比TSVW周期发生更快地被写入到有效BGF 1920。例如，如果在TSVW

为非真时写入源逻辑1914在(例如,时钟域A的)写入时钟周期中写入,那么累加器1916被递增,但是没有对有效FIFO 1918的写入发生。

[0155] 当TSVW信号为真(值‘1’)时,如果没有数据分组要被写入有效载荷FIFO 1921并且如果累加器为非零,那么‘1’将会被写入有效FIFO 1918中的按顺序的下一位置并且累加器将递减(例如,将‘1’从累加器传送到有效FIFO 1918)。如果TSVW真周期发生并且如果不存在从写入源逻辑1914到有效载荷FIFO 1921的数据分组的写入,并且不存在之前累加的写入(例如,累加器具有全零,并且因此在有效载荷FIFO 1921中不存在已存储的分组),则0被写入到有效FIFO 1918。

[0156] 在实施例中,如果所有的写入信用都已被耗尽,则写入源逻辑1914将被停止。写入信用的初始分配可与有效载荷FIFO 1921的容量相等(并且与有效FIFO 1918的容量相等)。累加器1918可根据有效FIFO 1918的尺寸来确定尺寸。例如,在一个实施例中,累加器1916可与有效FIFO 1918的容量相等。(附加地,累加器不会递减到低于零)。

[0157] 对有效FIFO 1918的读取可在读取时钟域B 1904的每个读取时钟周期中被完成,其中气泡发生器1940发出时隙有效读取(TSVR)的信号。有效读取指针1932在每次读取之后移动。当‘1’从有效FIFO 1918退出并进入读取mux 1950时,该‘1’可允许经由读取mux 1952从有效载荷FIFO 1921(如由有效载荷读取指针1946来指示)读取数据分组和/或可递增计数器(未示出),例如,当读取信用被应用并且读取被推迟时(读取信用在下文讨论)。对有效载荷FIFO 1921的读取的每个实例在从有效FIFO 1918读取的有效指示符为‘1’并且被输入到具有第二输入(TSVR信号)的读取mux 1950时发生。在实施例中,当读取mux 1950的两个输入都为真时,对有效载荷FIFO 1921的读取经由读取mux 1952而发生,并且BGF信用逻辑1960将写入信用发出到写入源信用逻辑1910,该写入信用在将来的写入窗口处允许另一写入发生。

[0158] 有效读取指针1932被初始化为少于有效写入指针1917的“N”,其中N是指针间隔。在一些实施例中,有效FIFO 1918的尺寸为有效读取指针1932与有效写入指针1917之间的间隔的至少两倍,以允许在两个方向上的偏斜,例如,来防止写入后读取不稳定性以及读取后写入不稳定性。有效读取指针1932和有效写入指针1917持续地移动,因此典型地不存在重叠。借助于指针总是相互分开的事实,不稳定性被典型地避免。写入到任何给定有效FIFO位置与读取该有效FIFO位置之间的时间延迟由指针间隔来控制。当有效FIFO 1918被停止(BGF运行信号被减小)时,有效写入指针1917被典型地重置为指向第一条目并且有效FIFO从锁存器被重置。

[0159] 在物理上,诸如有效载荷FIFO 1921之类的有效载荷FIFO可与有效FIFO类似。主要差异在于写入和读取指针如何操作,例如,有效载荷FIFO的写入指针和读取指针两者都以0开始。在实施例中,对有效载荷FIFO的写入仅在存在分组要被写入时并且存在可用的有效载荷信用时发生。与其中有效写入指针在每次存在写入窗口时都递增的有效FIFO不同,在有效载荷FIFO的情形中,写入指针仅在数据分组被写入到有效载荷FIFO时递增。对有效载荷FIFO的读取在有效FIFO的输出指示在有效载荷FIFO中存在有效数据时发生。将有效载荷FIFO与有效FIFO配对确保了在读取发生前,写入指针位于读取指针之前,从而防止亚稳态并且允许时钟偏移。

[0160] 在实施例中,有效FIFO/有效载荷FIFO组合操作使得写入仅在有效载荷信用可用

时进行；写入进行到下一个有效载荷FIFO位置，并且写入指针递增；当写入器源具有分组时，‘1’被发送到有效FIFO，这将会使该分组在下一个写入窗口中被写入到有效载荷FIFO；而在读取器的一侧，当有效FIFO的输出为‘1’时，有效载荷FIFO中的数据可被读取。

[0161] 在各种实施例中，反压机制（例如，写入信用的使用）使写入器侧仅在写入器侧具有信用时能够将分组写入到有效载荷FIFO。只有在有效载荷FIFO中存在空位置时，写入器侧才将接收信用。每当读取器侧从有效载荷FIFO读取时，有效载荷FIFO中的位置就会被空出，并且读取器侧将把信用发送到写入器侧。

[0162] 现在参考图20，示出了根据实施例的功率控制方法的流程图。更具体地，在图20中所示的方法2000被用于使具有对处理器的剩余部分或其他SoC隐藏的功率状态的核能够对处理器的剩余部分透明的方式进入或退出某个低功率状态。以这种方式，其他处理器组件可在此类低功率状态期间继续正常操作，包括发送事件和其他信号以及从核接收所期望的信号。

[0163] 如所示，方法2000通过接收核低功率状态进入指示开始（框2010）。该低功率状态进入指示可，例如，在核要进入低功率状态时由与核相关联的核周边逻辑从功率控制器接收。下一步在框2020处，一个或多个核周边逻辑可被激活来处理针对核的某个活动。更具体地，该活动可以是与IDI/结构/非核相关联的任何活动，诸如以上所描述的各种监听、信用以及锁定操作。此类逻辑激活可响应于，例如，从PMA接收的低功率状态信号。此后在框2030，核可进入适当的低功率状态。出于讨论的目的，假设该低功率状态为C6低功率状态。在此实例中，核高速缓存存储器被转储清除并且核的上下文信息可被存储到该核内的C6 SRAM，该C6 SRAM由此低功率状态中的保持电压来保持供电。

[0164] 仍然参考图20，在框2040处，核状态被维持在活跃状态，虽然该核处于低功率状态。注意，维持状态为活跃的不同方式在不同的实施例中是可能的。通过使低功率状态进入对处理器代理维持隐藏，与低功率状态进入相关联的对结构接口、结构以及其他组件的各种排空流程可被避免。更进一步地，也可避免阻塞操作。

[0165] 仍然参考图20，在框2050处，在核处于低功率状态时，任何对核周边逻辑的传入活动可由核周边逻辑的组成逻辑来处理。此类活动可包括各种IDI/结构/非核活动，包括传入监听、锁定消息、信用消息、事件等等。

[0166] 假设现在核退出低功率状态而进入活跃状态并且将此次退出通知核周边逻辑。控制随后传递到框2060，其中核可用各种状态信息来更新，各种状态信息包括信用状态信息和锁定状态信息。当然，在不同的实施例中，额外的状态信息也可被传达到核。此后，控制传递到框2070，其中各种核周边逻辑可被去激活，使得传入事件、信号、请求等等可被提供以由该核来直接处理，而不是由核周边逻辑的组成逻辑来处理。要理解，虽然在图20的实施例中以该高层级示出，但许多变型和替换方案是可能的。

[0167] 以下各示例涉及进一步的实施例。

[0168] 在一个示例中，处理器包括用于执行指令的核以及耦合到该核的核周边逻辑，该核周边逻辑包括具有第一存储的结构接口逻辑，第一存储用于在核处于低功率状态时存储该核的状态信息。核周边逻辑用于使在核与非核之间耦合的管芯间互连在核进入低功率状态期间能被维持在活跃状态。

[0169] 在示例中，结构接口逻辑包括监听逻辑，该监听逻辑用于在核处于所述低功率状

态时从请求方接收监听请求并将监听响应发送给请求方,其中核将保持在低功率状态。

[0170] 在示例中,监听响应包括默认监听响应,该默认监听响应用于指示核不包括与监听请求相关联的数据。

[0171] 在示例中,监听逻辑用于在核处于活跃状态时将监听请求提供给核,并且用于在核处于低功率状态时将默认监听响应发送给请求方。

[0172] 在示例中,核周边逻辑用于使核的功率状态能对非核为透明,以使非核在核进入低功率状态的进入流程期间维持多个待决事务,而不排空多个待决事务。

[0173] 在示例中,结构接口逻辑进一步包括信用逻辑,该信用逻辑用于在核处于低功率状态时维持多个信用值。

[0174] 在示例中,信用逻辑进一步用于在核处于低功率状态时更新多个信用值中的一个或多个。

[0175] 在示例中,核周边逻辑进一步包括边带事件逻辑,该边带事件逻辑用于接收并存储经由被耦合到核周边逻辑的边带互连而接收的一个或多个边带事件,该边带事件逻辑用于向功率控制器通知关于一个或多个边带事件的接收,以使功率控制器能确定是否要使核退出低功率状态来处理一个或多个边带事件的至少一个。

[0176] 在示例中,结构接口逻辑进一步包括锁定响应逻辑,该锁定响应逻辑用于接收用于原子操作的锁定静默消息,并且当核处于低功率状态时,该锁定响应逻辑用于将默认响应消息发送给非核。

[0177] 在示例中,锁定响应逻辑用于在核在原子操作完成之前退出低功率状态时向该核通知原子操作,以防止该核在原子操作期间发送一个或多个事务。

[0178] 在示例中,核进一步包括报告寄存器,该报告寄存器用于存储在核退出低功率状态之后、在恢复来自核的存储器的保持部分的状态信息期间的与不可纠正的错误的发生相关联的错误信息。

[0179] 在示例中,核将执行微代码来将来自报告寄存器的错误信息写入到机器校验架构存储,以使机器校验架构错误能被提出。

[0180] 注意,可使用各种方式来实现上述处理器。

[0181] 在示例中,处理器包括被结合在用户装备启用触摸的设备中的SoC。

[0182] 在另一示例中,系统包括显示器和存储器,并且包括以上示例中的一个或多个中的处理器。

[0183] 在另一示例中,方法包括:使被耦合到多核处理器的核的核周边逻辑的一个或多个逻辑单元能响应于核将进入第一低功率状态的指示来处理对核的传入活动;以及在核处于第一低功率状态时将核的状态维持为活跃,以向处理器的结构以及一个或多个IP逻辑指示该核处于活跃状态,从而使核的第一低功率状态能对结构以及一个或多个IP逻辑隐藏。

[0184] 在示例中,该方法进一步包括,当核处于第一低功率状态时,在核周边逻辑的一个或多个逻辑单元中处理从结构以及一个或多个IP逻辑被引导至核的传入活动。

[0185] 在示例中,该方法进一步包括在核从第一低功率状态退出时,至少部分地基于存储在核周边逻辑的一个或多个逻辑单元的至少一个中的状态信息来更新关于信用状态和处理器锁定状态中的一个或多个的状态信息。

[0186] 在示例中,该方法还包括:在核处于第一低功率状态时在核周边逻辑的第一逻辑

单元中接收来自请求方的监听请求,将监听响应从所述第一逻辑单元发送给请求方,该监听响应包括默认监听响应,默认监听响应用于指示核不包括与监听请求相关联的数据。

[0187] 在示例中,该方法还包括:当核处于第一低功率状态时在核周边逻辑的第二逻辑单元中接收用于原子操作的锁定静默消息;经由结构来发送默认响应消息以确认锁定静默消息;以及当核在原子操作完成之前退出第一低功率状态时向核通知原子操作。

[0188] 在另一示例中,一种计算机可读介质,包括指令,所述指令用于执行上述示例中的任一项的方法。

[0189] 在另一示例中,一种计算机可读介质,包括数据,所述数据用于由至少一个机器使用以制造至少一个集成电路来执行上述示例中的任一项的方法。

[0190] 在另一示例中,设备包括用于执行上述示例中的任一项的方法的装置。

[0191] 在另一示例中,一种系统,包括多核处理器,所述多核处理器包括多个核、非核、结构以及用于耦合多个核与结构的管芯间互连,其中管芯间互连用于在多个核中的一个或多个处于低功率状态时保持活跃,处于低功率状态的一个或多个核的状态对管芯间互连和结构隐藏,以使多个事务在进入低功率状态的进入流程期间以及从低功率状态的退出流程期间能被维持在结构中;以及系统可以进一步包括耦合至多核处理器的系统存储器。

[0192] 在示例中,非核包括:有效载荷时钟交叉逻辑,用于存储用来向第一核发送的数据分组;以及写入信用逻辑,写入信用逻辑用于维持写入信用的计数,其中当第一核处于低功率状态时,该有效载荷时钟交叉逻辑用于响应于写入信用的计数超过第一值来存储新的数据分组。

[0193] 在示例中,写入信用逻辑用于响应于数据分组正被写入有效载荷时钟交叉逻辑来更新写入信用的计数。

[0194] 在示例中,如果写入信用的计数小于第一值,那么将阻止写入源将第二分组写入到有效载荷时钟交叉逻辑。

[0195] 应理解,上述示例的各种组合是可能的。

[0196] 实施例可以被用于许多不同类型的系统中。例如,在一个实施例中,可以将通信设备布置为用于执行本文所述的各种方法与技术。当然,本发明的范围不限于通信设备,相反,其他实施例可以涉及用于处理指令的其他类型的装置,或一个或多个机器可读介质,该机器可读介质包括指令,响应于在计算设备上执行这些指令,这些指令使该设备执行本文所述的方法与技术中的一个或多个。

[0197] 实施例可以实现在代码中,并且可以存储在非暂态存储介质中,该非暂态存储介质具有存储于其上的指令,该指令可以被用来对系统编程以执行指令。各实施例还可以在数据中实现,并且可以存储在非瞬态存储介质中,该非瞬态存储介质如果被至少一个机器使用,将使得该至少一个机器制造至少一个集成电路以执行一个或多个操作。存储介质可以包括但不限于,任何类型的盘,包括软盘、光盘、固态驱动器(SSD)、紧致盘只读存储器(CD-ROM)、紧致盘可重写(CD-RW)以及磁光盘;半导体器件,诸如,只读存储器(ROM)、诸如动态随机存取存储器(DRAM)与静态随机存取存储器(SRAM)的随机存取存储器(RAM)、可擦除可编程只读存储器(EPROM)、闪存、电可擦除可编程只读存储器(EEPROM);磁卡或光卡;或适用于存储电子指令的任何其他类型的介质。

[0198] 虽然已参照有限数量的实施例描述了本发明,但是本领域技术人员将从中领会很

多修改和变型。所附权利要求旨在涵盖落入本发明的真实精神与范围的所有此类修改与变型。

100

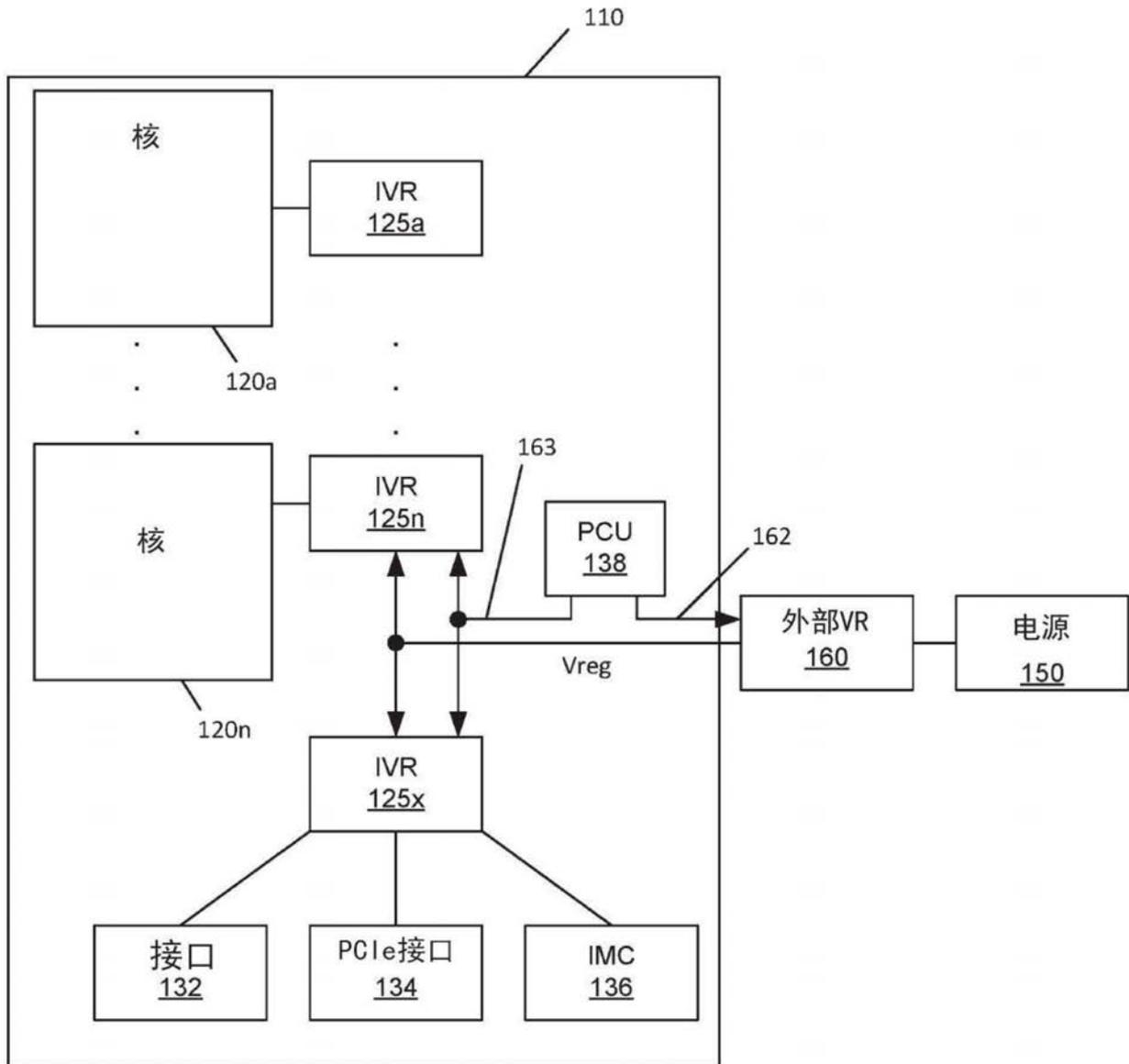
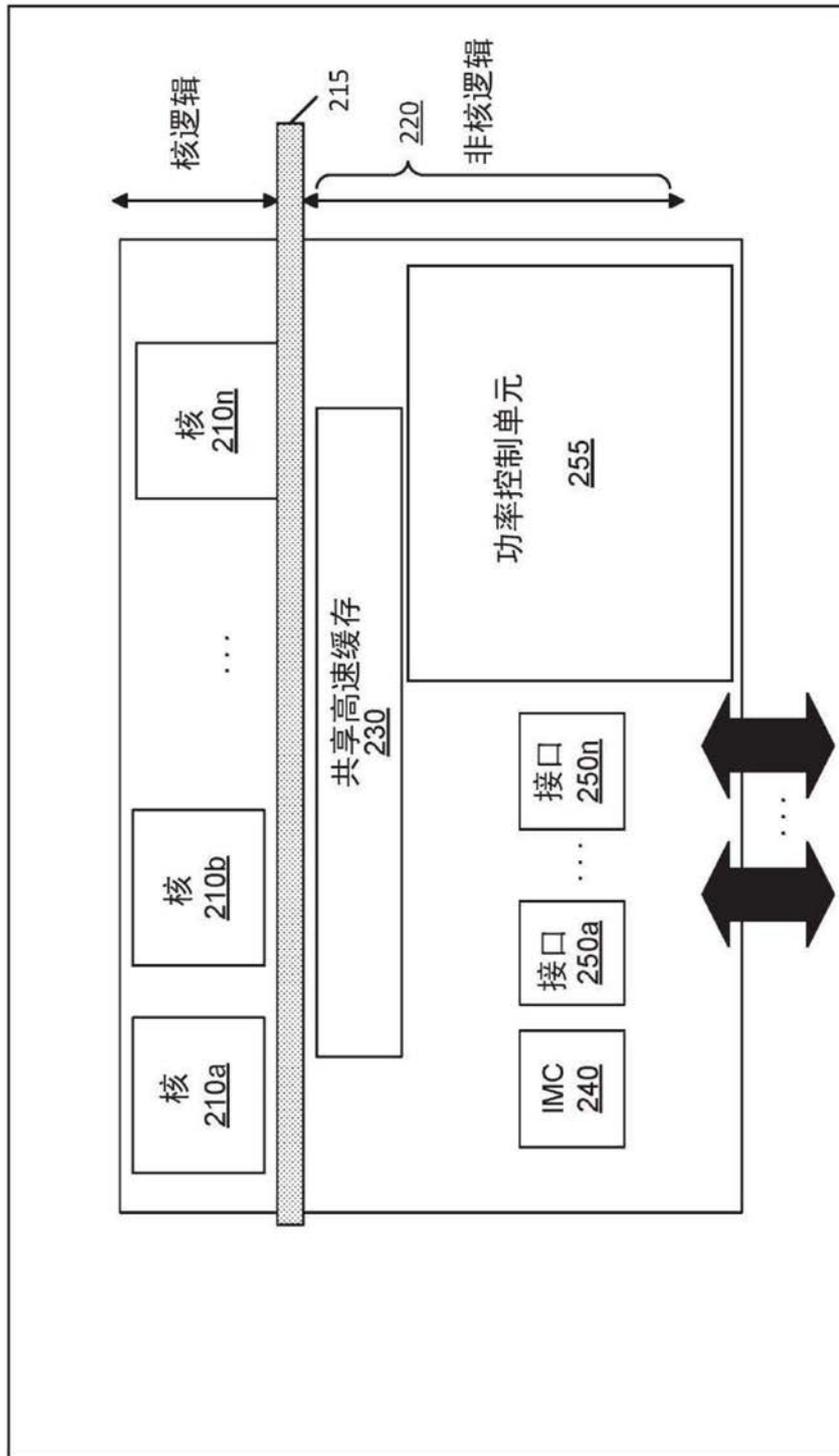


图1



200

图2

300

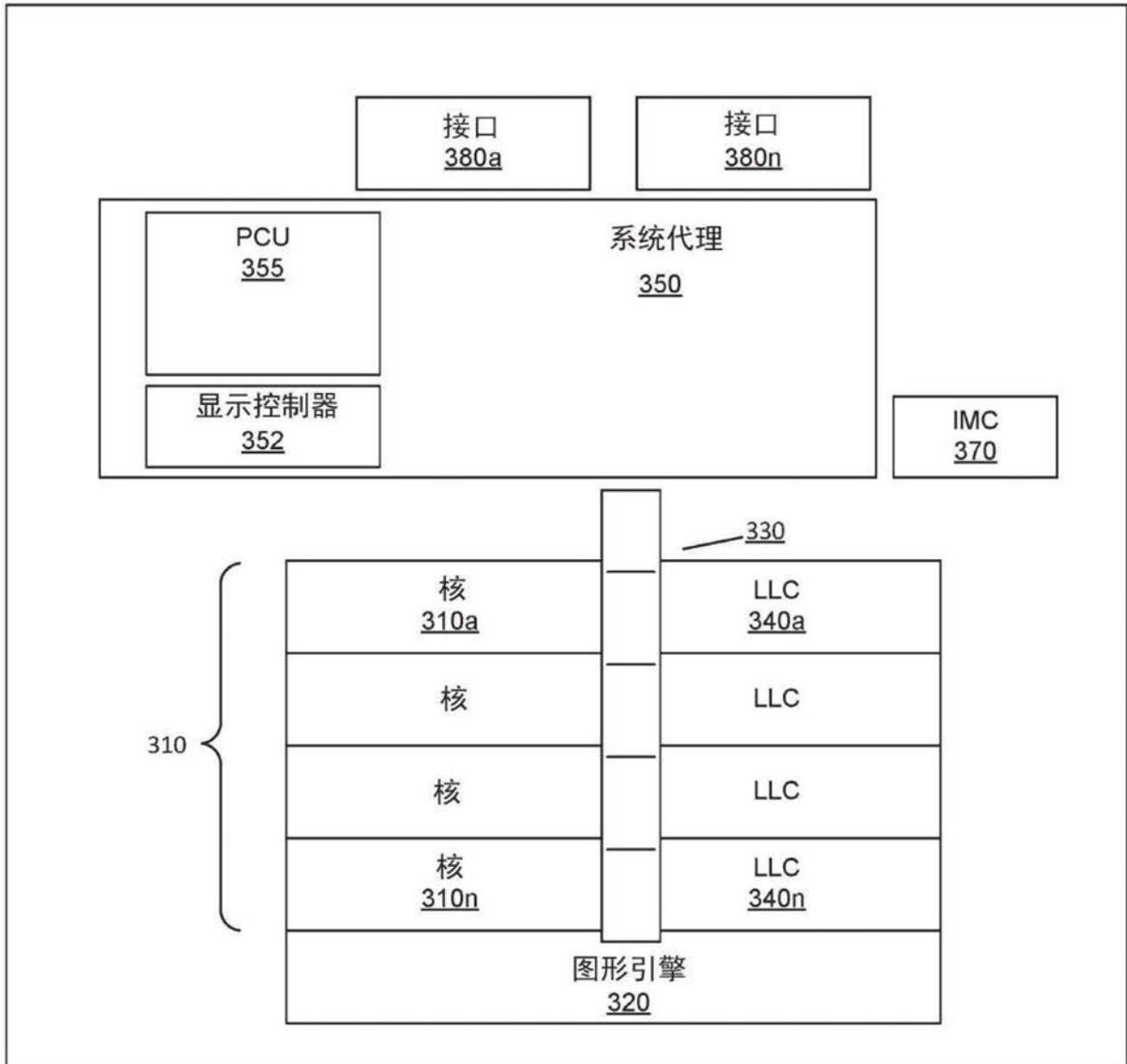


图3

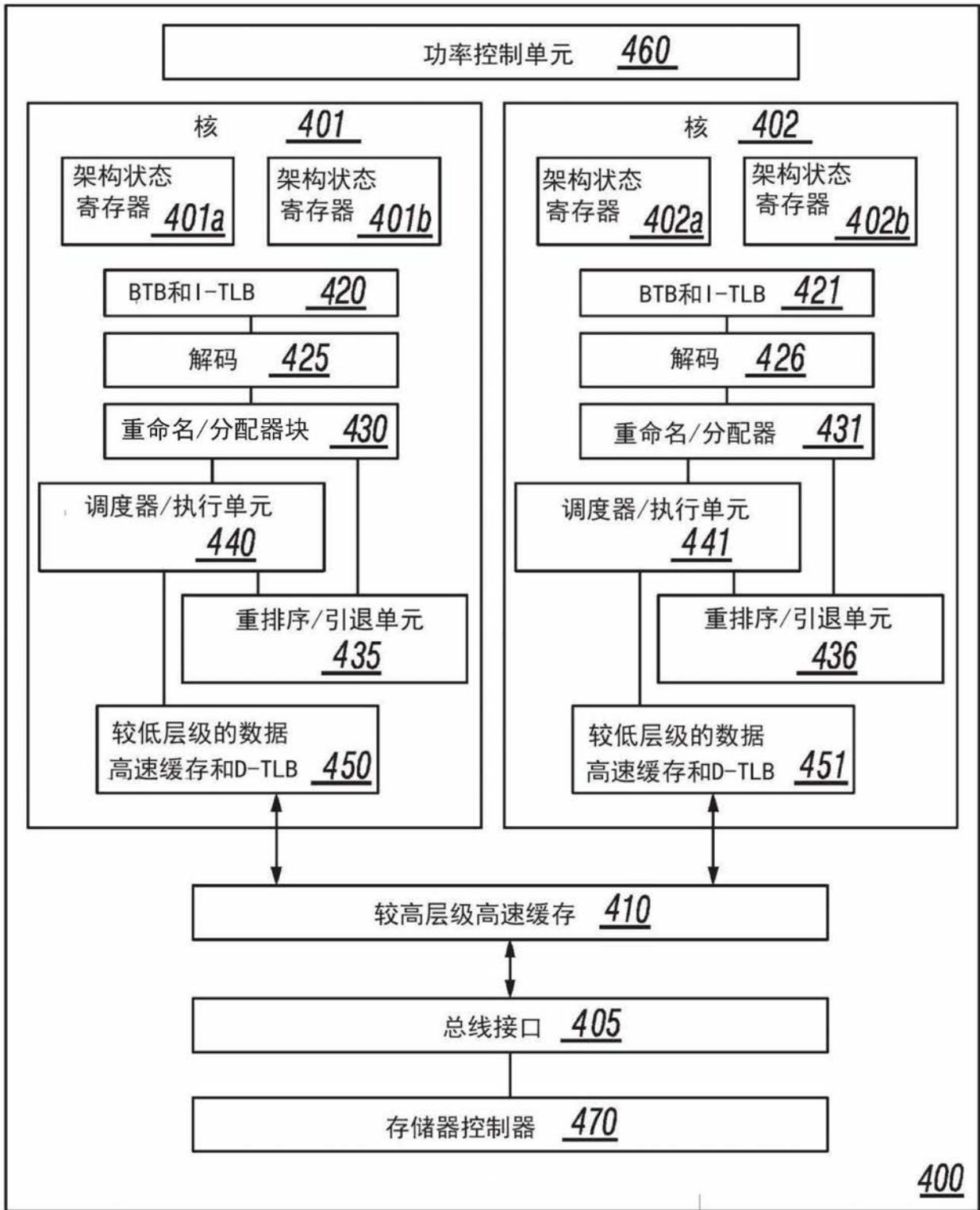


图4

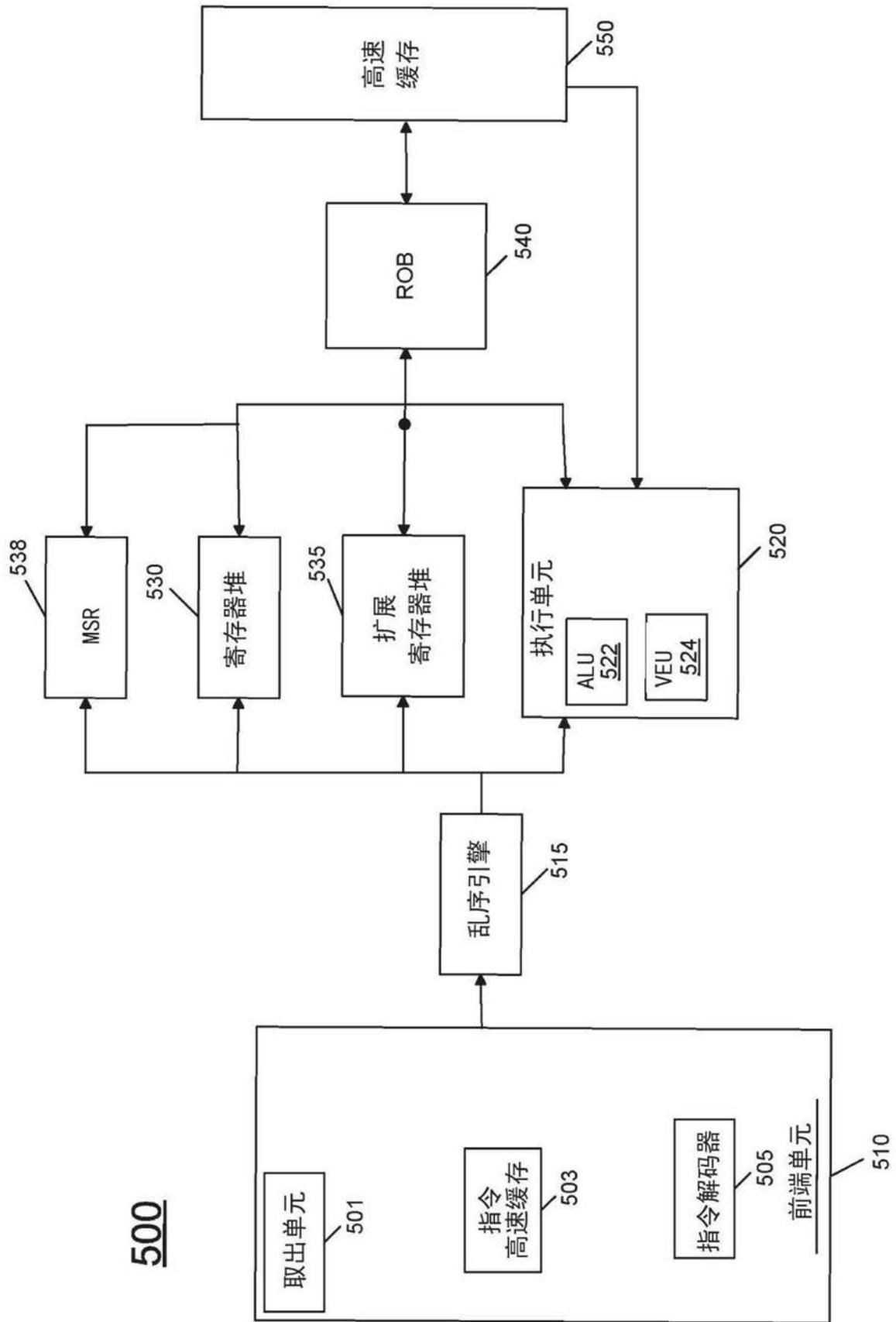


图5

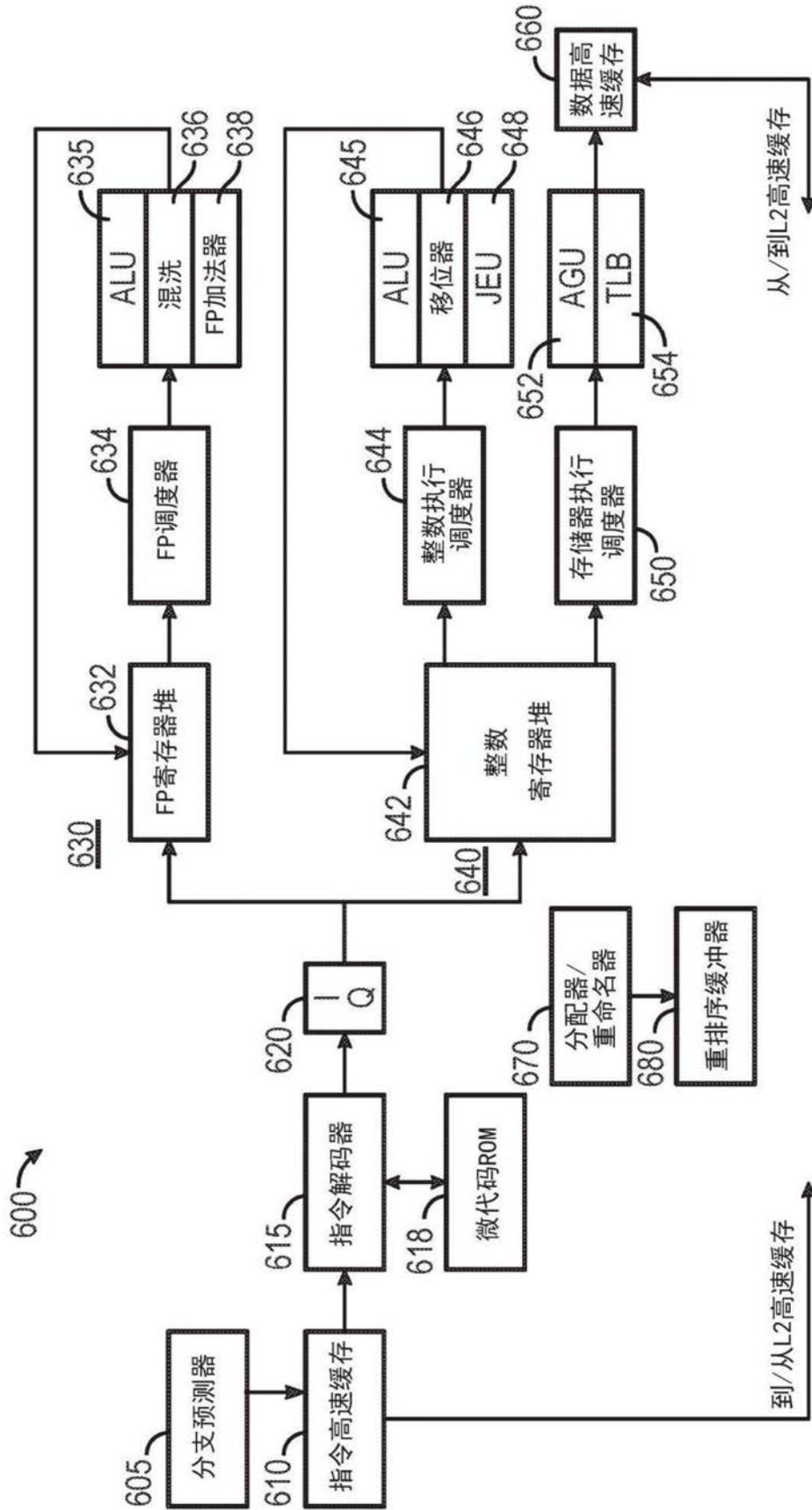


图6

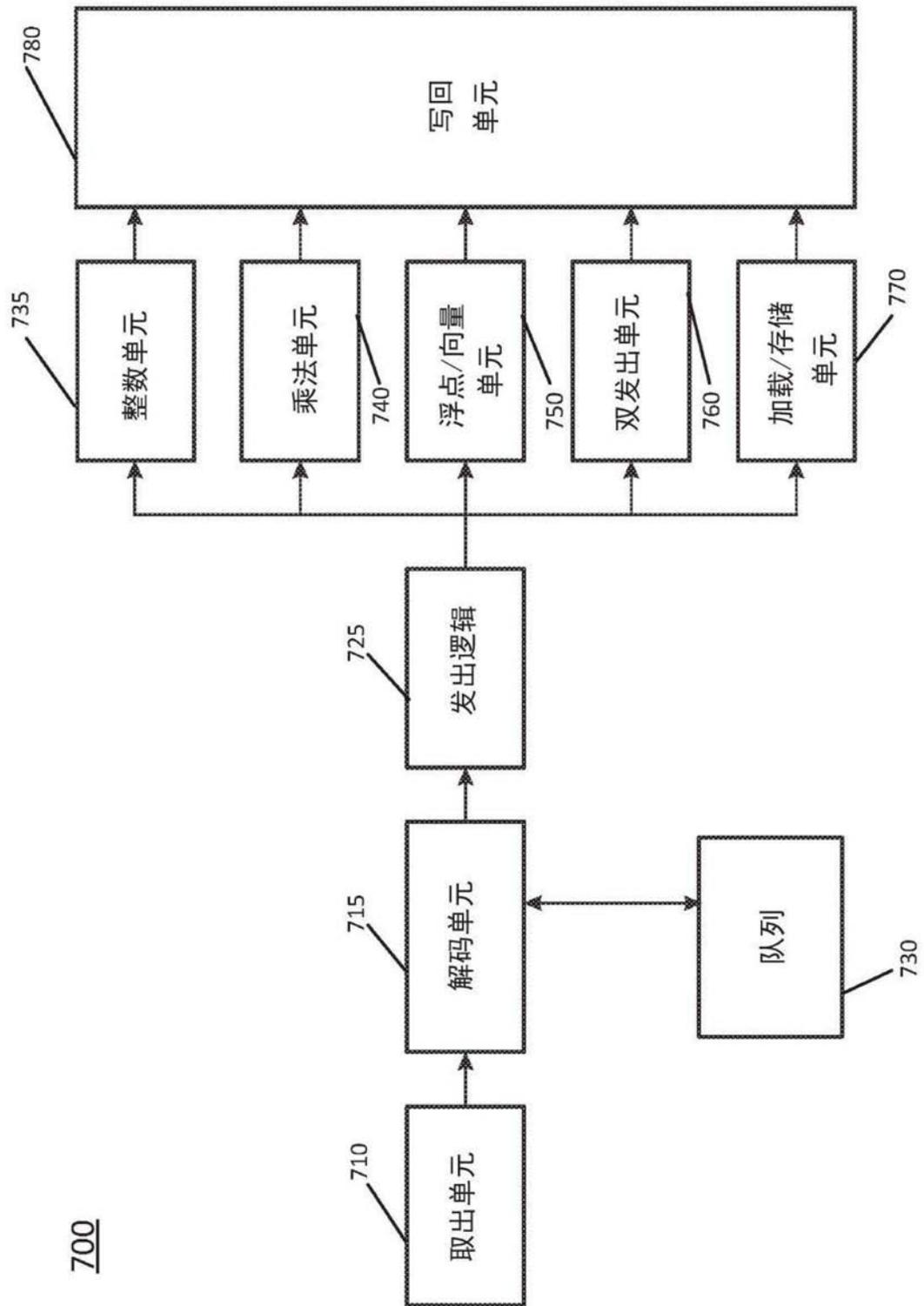


图7

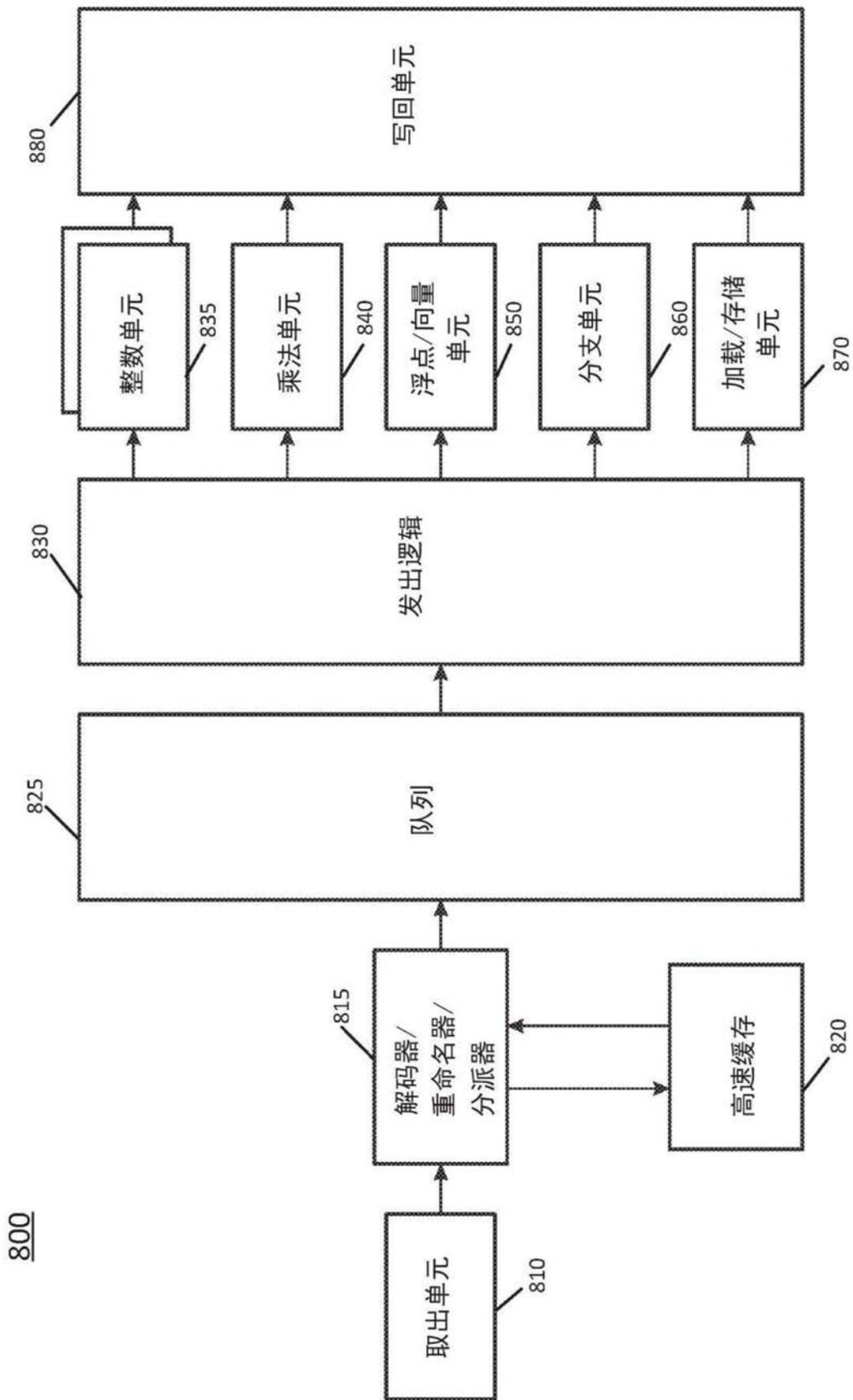


图8

900



图9

1000

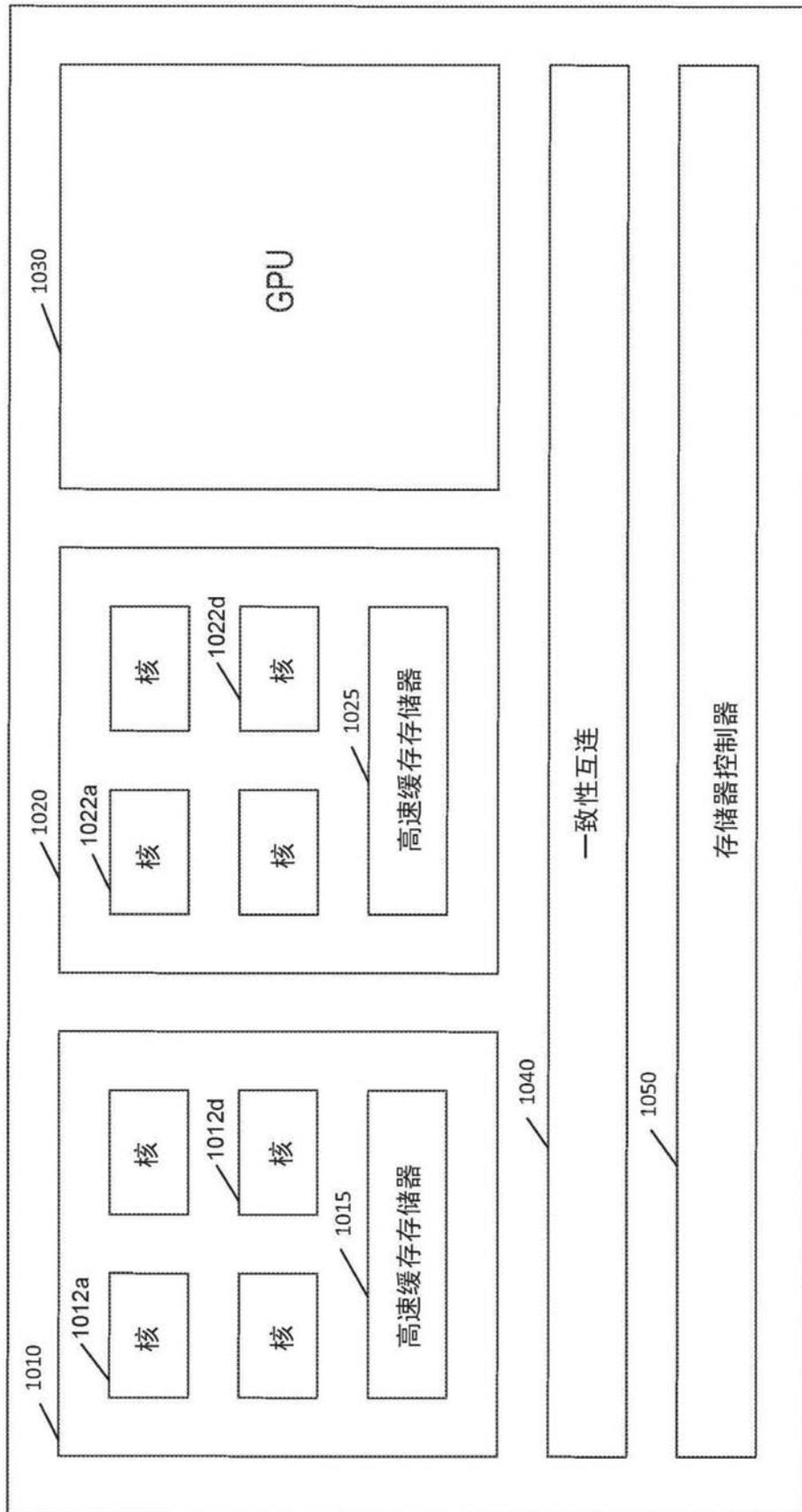


图10

1100

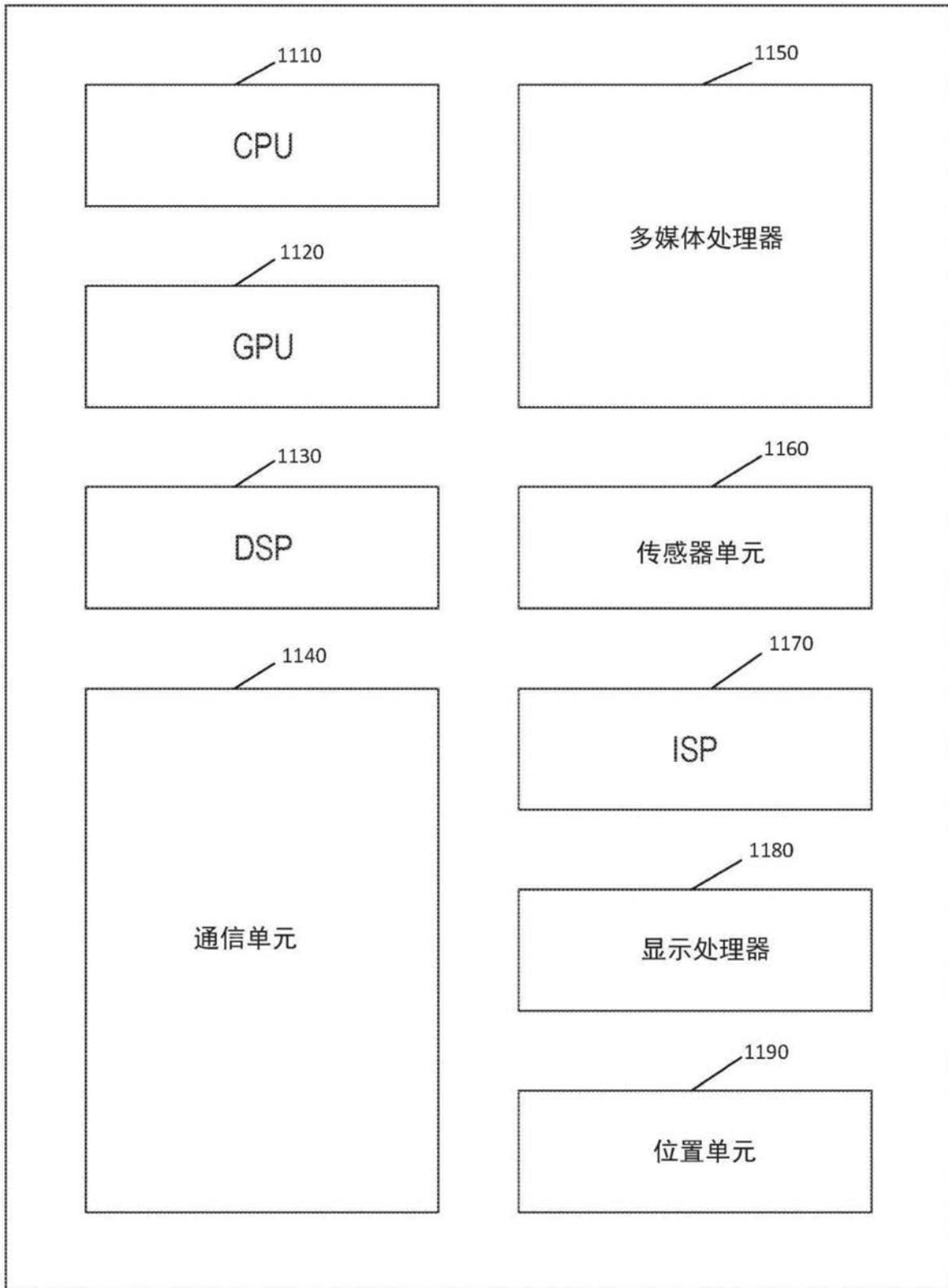


图11

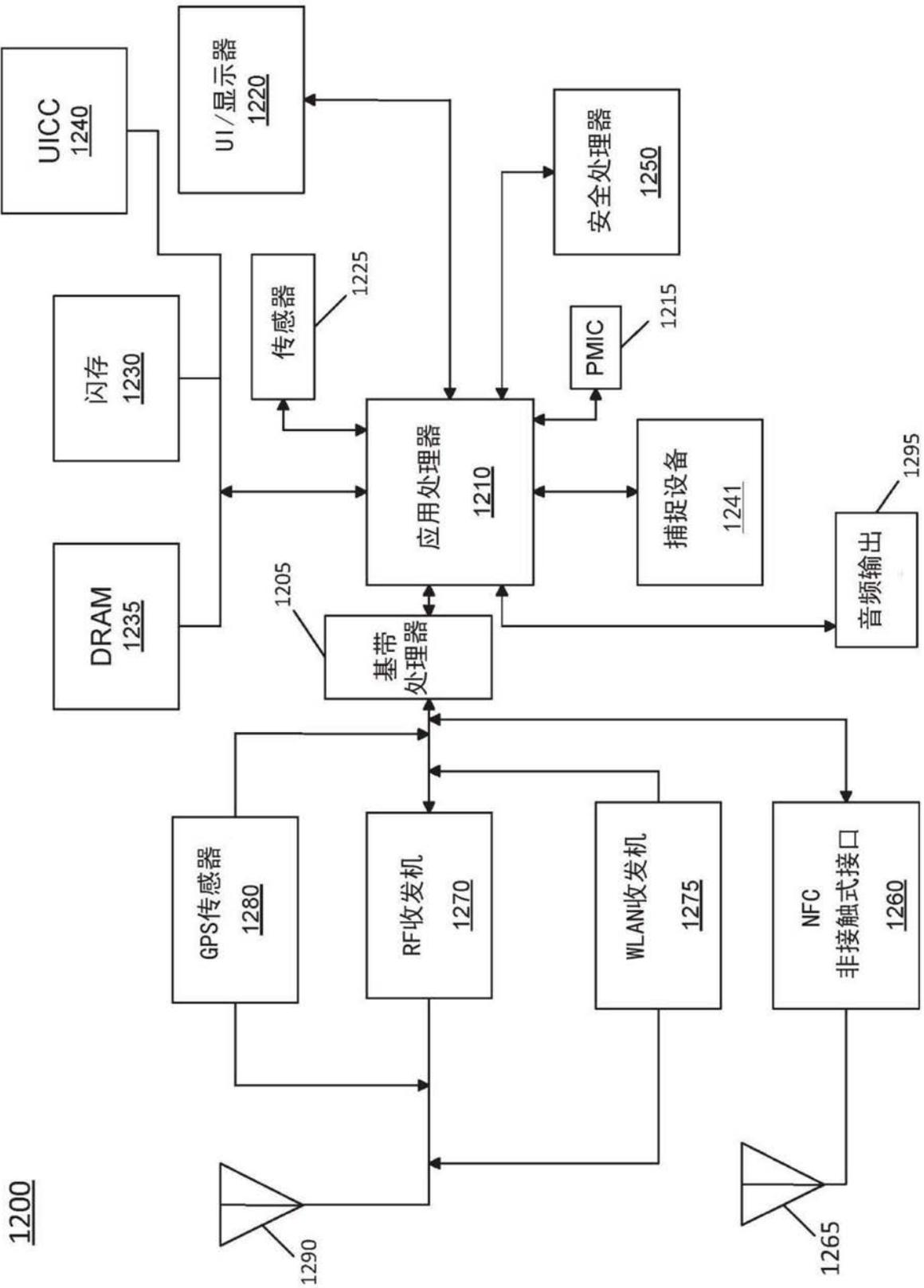


图12

1300

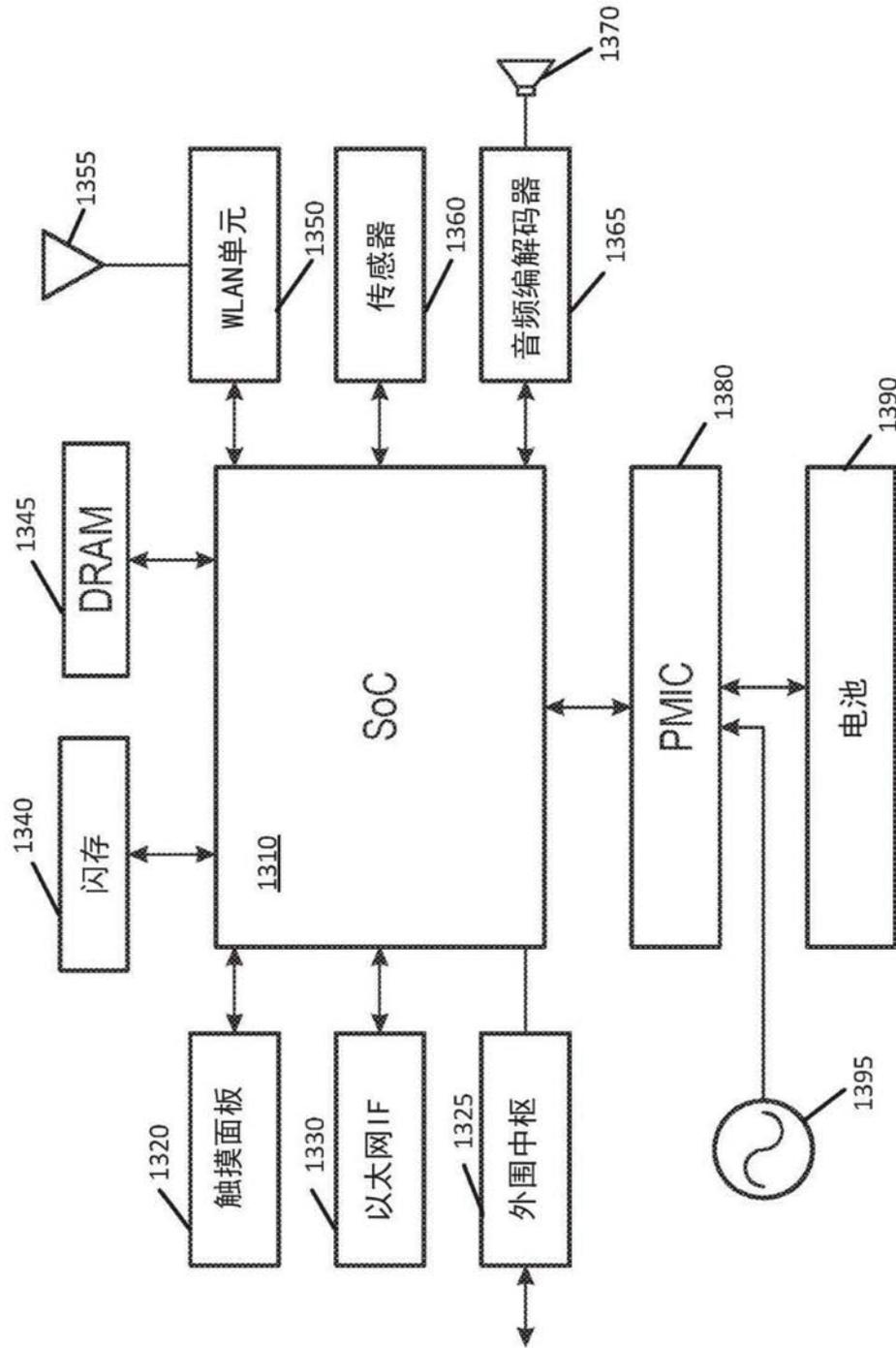


图13

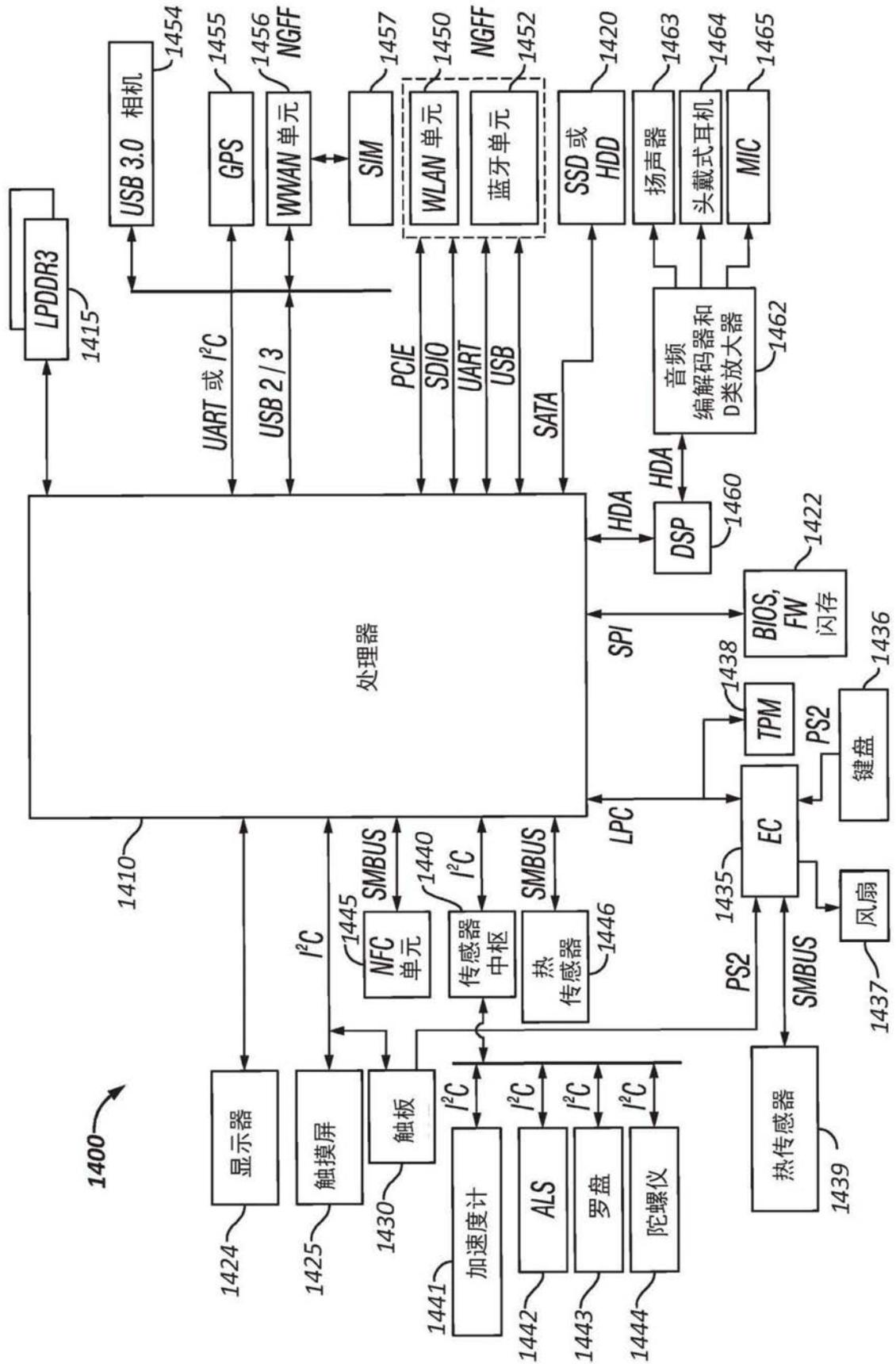


图14

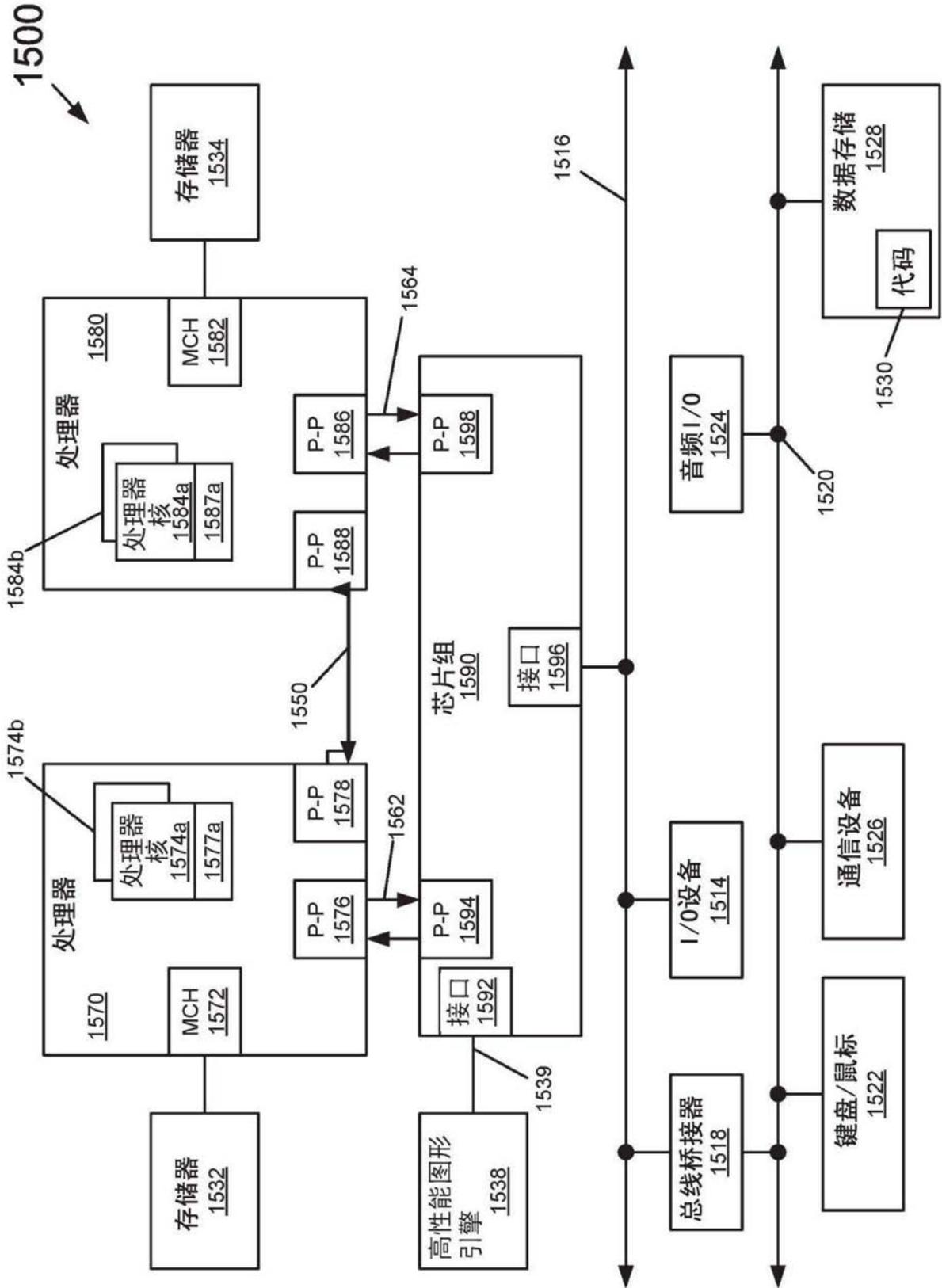


图15

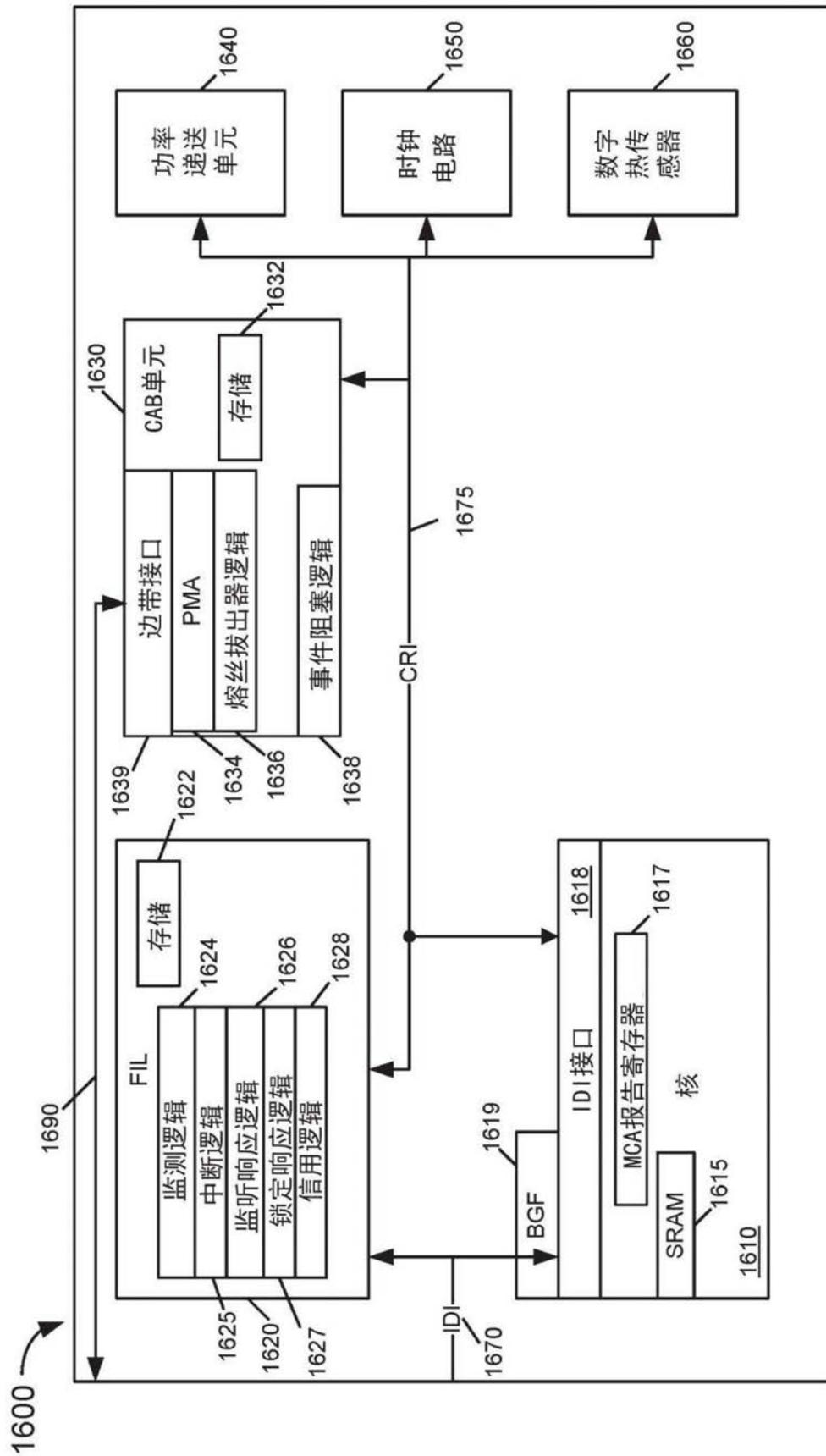


图16

1800

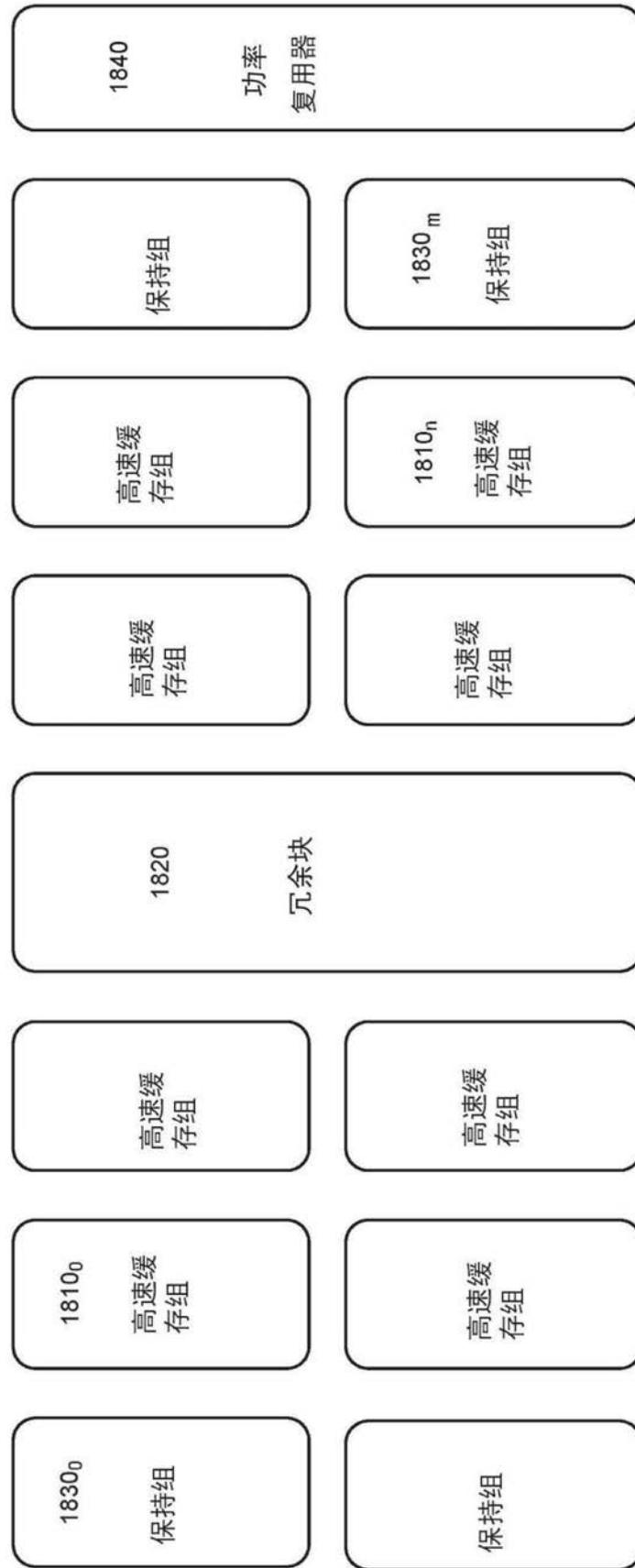


图18

2000

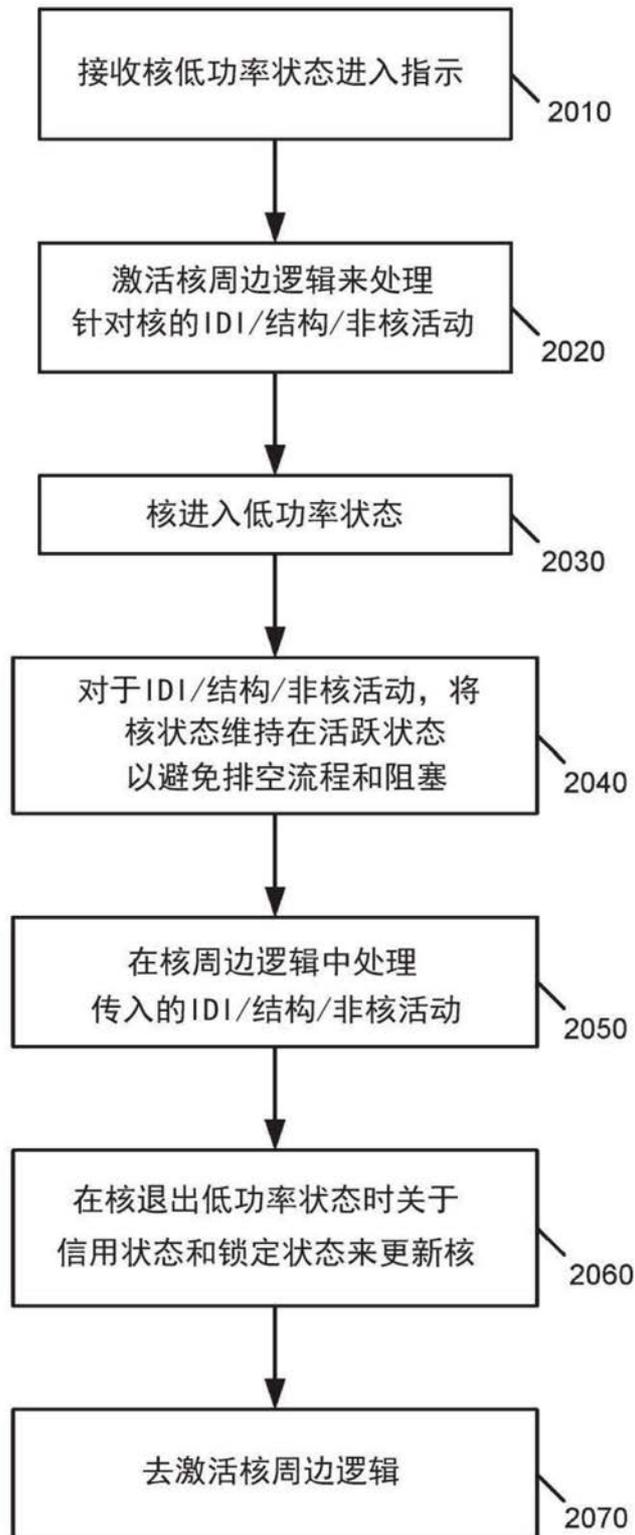


图20