



(12) 发明专利申请

(10) 申请公布号 CN 118535228 A

(43) 申请公布日 2024. 08. 23

(21) 申请号 202410698163.5

(22) 申请日 2019.11.19

(30) 优先权数据

16/231,050 2018.12.21 US

(62) 分案原申请数据

201911135320.7 2019.11.19

(71) 申请人 英特尔公司

地址 美国加利福尼亚州

(72) 发明人 A·F·海奈克 E·吉奥加纳斯

C·J·休斯 R·萨德 R·凡伦天

(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

(51) Int. Cl.

G06F 9/38 (2018.01)

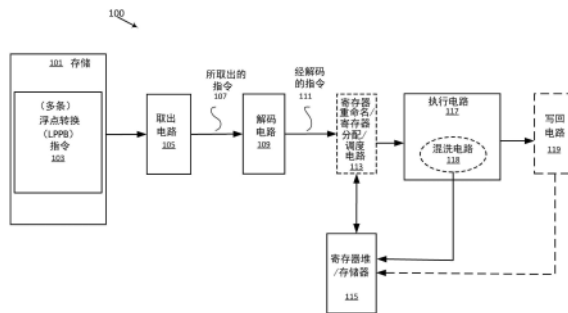
权利要求书2页 说明书30页 附图30页

(54) 发明名称

用于在从存储器加载时在进行中转置向量的系统和方法

(57) 摘要

本申请公开了用于在从存储器加载时在进行中转置向量的系统和方法。在一个示例中，一种处理器，包括：寄存器堆，存储器接口，取出电路，用于取出指令，解码电路，用于对所取出的指令解码，所取出的指令具有用于指定操作码、目的地向量寄存器、以及具有N组元素的源向量的字段，N为正整数，操作码用于指示处理器用于：取出源向量，生成包括一个或多个N元组的写入数据，每个N元组包括来自N组元素中的每一组的对应元素，并且将写入数据写入目的地向量寄存器，以及执行电路，用于根据操作码执行经解码的指令，执行电路具有放置在存储器与寄存器堆之间的混洗流水线，混洗流水线用于取出、解码、以及以一个指令每时钟周期执行指令的进一步的实例。



1. 一种处理器,包括:

寄存器堆,包括一个或多个向量寄存器;

存储器接口,用于从存储器读取多个数据元素;

取出电路,用于取出指令;

解码电路,用于对所述指令解码,所述指令包括用于指示操作码、用于广播的所述多个数据元素的子集、以及所述多个数据元素的位置的多个字段,所述多个数据元素被布置在对应的多个相对位置中,其中所述多个数据元素包括第一组数据元素和第二组数据元素;

执行电路,用于执行所述指令,所述执行电路用于根据所述指令执行置换操作和广播操作,其中所述广播操作用于使所述多个数据元素的所述子集被广播至与所述多个数据元素中的对应的多个其他子集相关联的多个所述相对位置,所述多个数据元素的所述子集用于替换在多个所述相对位置处的所述对应的多个其他子集。

2. 如权利要求1所述的处理器,其中,所述置换操作用于改变所述多个数据元素中的至少一部分数据元素的相对位置。

3. 如权利要求1或2所述的处理器,其中,所述执行电路用于基于所述指令的所述多个字段中的字段执行所述置换操作。

4. 如权利要求1至3中任一项所述的处理器,其中,所述多个数据元素被布置在存储器中的所述对应的多个相对位置中,并且其中所述执行电路用于执行所述置换操作和所述广播操作以将所述多个数据元素加载至所述一个或多个向量寄存器。

5. 如权利要求4所述的处理器,其中,所述执行电路用于将所述多个数据元素作为2元组加载至所述一个或多个向量寄存器,每个2元组包括所述第一组数据元素中的一个数据元素和所述第二组数据元素中的一个数据元素。

6. 如权利要求1至5中任一项所述的处理器,其中,所述指令的所述多个字段中的一个或多个字段用于进一步指示所述第一组数据元素和所述第二组数据元素的宽度,所述宽度为32位、64位、128位、256位、512位和1024位中的一者。

7. 如权利要求6所述的处理器,其中,所述指令的所述多个字段中的一个或多个字段用于进一步指示所述第一组数据元素和所述第二组数据元素中的每个数据元素的尺寸,所述尺寸为2位、4位、8位、16位、32位、64位和128位中的一者。

8. 如权利要求4所述的处理器,其中,所述第一组数据元素和所述第二组数据元素被连续地布置在所述存储器中。

9. 如权利要求4所述的处理器,其中,所述指令的所述多个字段中的一个或多个字段用于进一步指示所述多个数据元素被存储的一个或多个存储器位置。

10. 如权利要求1至9中任一项所述的处理器,其中,所述执行电路用于根据所述指令的所述多个字段中的一个或多个字段执行所述置换操作和所述广播操作。

11. 一种方法,包括:

取出指令;

对所述指令解码,所述指令包括用于指示操作码、用于广播的多个数据元素的子集、以及所述多个数据元素的位置的多个字段,所述多个数据元素从存储器被读取并且被布置在对应的多个相对位置中,其中所述多个数据元素包括第一组数据元素和第二组数据元素;

执行所述指令以根据所述指令执行置换操作和广播操作,其中所述广播操作用于使所

述多个数据元素的所述子集被广播至与所述多个数据元素中的对应的多个其他子集相关联的多个所述相对位置,所述多个数据元素的所述子集用于替换在多个所述相对位置处的所述对应的多个其他子集。

12. 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11所述的方法。

13. 一种计算机可读介质,具有存储于其上的如权利要求12所述的计算机程序产品。

14. 一种处理器,包括:

a) 前端单元,所述前端单元包括:用于从存储器中取出指令的指令取出单元,和用于将所述指令解码为微操作的解码单元;

b) 执行引擎单元,所述执行引擎单元包括:重命名/分配器单元,用于分配每个微操作需要的机器缓冲器和资源以及将多个逻辑寄存器重命名为寄存器堆中的条目;调度器单元,用于对分配端口进行仲裁以调度微操作以便执行;执行集群,包括执行单元和存储器访问单元;以及

c) 存储器单元,其耦合至所述前端单元和所述执行引擎单元。

15. 一种芯片上系统,包括:

应用处理器,所述应用处理器包括一个或多个核和共享高速缓存单元;

协处理器;以及

互连单元,其耦合所述协处理器和所述应用处理器。

16. 一种方法,包括:

使用x86编译器编译采用高级程序语言的程序,以生成由具有至少一个x86指令集核的第一处理器原生执行的x86二进制代码;

使用指令转换器,将所述x86二进制代码转换成能够由不具有x86指令集核的第二处理器原生执行的替代二进制代码。

用于在从存储器加载时在进行中转置向量的系统和方法

本发明专利申请是2019年11月19日提交的申请号为201911135320.7,名称为“用于在从存储器加载时在进行中转置向量的系统和方法”的发明专利申请的分案申请。

技术领域

[0001] 本发明的领域一般涉及计算机处理器架构,并且更具体地涉及用于在从存储器加载时转置向量的系统和方法。

背景技术

[0002] 指令集或指令集架构 (ISA) 是计算机架构中与编程有关的部分,并且可包括原生数据类型、指令、寄存器架构、寻址模式、存储器架构、中断和异常处置以及外部输入和输出 (I/O)。指令集包括一种或多种指令格式。给定的指令格式定义各种字段(位的数目、位的位置)以指定将要被执行的操作以及将要对其执行那个操作的(多个)操作数,等等。给定的指令使用给定的指令格式来表达,并且指定操作和操作数。指令流是特定的指令序列,其中,该序列中的每条指令是指令按指令格式的出现。

[0003] N元组频繁地用于计算应用的情境中,诸如语音识别、机器学习和深度学习。如本文中所使用,N元组是N个整数元素的有序序列。对N不存在上限,并且其中N等于1、2、3、4或5的N元组可以分别称为单个一组 (monad)、两个一组 (duad)、三个一组 (triad)、四个一组 (tetrad) 和五个一组 (pentad)。

[0004] N元组的一组M个实例可以被表示为紧缩数据向量,或者将M个实例中的每一个的元素一个接一个地交错,或者将M个实例的元素分离到N个组中,每个组具有M个元素。

[0005] 应用经常需要通过置换紧缩数据向量的元素将N元组的一组M个实例从分离表示转置为交错表示。有时,应用需要将N元组的M个实例中的一个、两个或更多个从分离表示转置为交错表示,并且将一个、两个或更多个N元组跨整个向量寄存器广播。尽管此类转置和广播可以使用现有向量指令来实现,但是这样做通常是缓慢的,导致大的代码尺寸,并且频繁地需要复杂的软件调节。

附图说明

[0006] 图1A是图示根据实施例的用于执行可能具有广播的加载置换 (LPPB) 指令的处理组件的框图;

[0010] 图1B是图示根据实施例的可能具有广播的加载置换 (LPPB) 指令的执行的框图;

[0011] 图2A是图示根据实施例的具有一个周期每指令的吞吐量的混洗电路的框图;

[0012] 图2B是图示根据实施例的可能具有广播的加载置换 (LPPB) 指令的执行的框图;

[0013] 图2C是图示根据实施例的可能具有广播的加载置换 (LPPB) 指令的执行的框图;

[0014] 图2D是图示根据实施例的可能具有广播的加载置换 (LPPB) 指令的执行的框图;

[0015] 图2E是图示根据实施例的加载广播指令的执行的框图;

[0016] 图2F是图示根据实施例的加载广播指令的执行的框图;

- [0017] 图2G是图示根据实施例的加载广播指令的执行的框图；
- [0018] 图2H是图示根据实施例的加载广播指令的执行的框图；
- [0019] 图2I是图示根据实施例的加载广播指令的执行的框图；
- [0020] 图2J是图示根据实施例的加载广播指令的执行的框图；
- [0021] 图2K是图示根据实施例的加载广播指令的执行的框图；
- [0022] 图3是图示根据实施例的处理器对可能具有广播的加载置换 (LPPB) 指令作出响应的流程图；
- [0023] 图4A是图示根据实施例的可能具有广播的加载置换 (LPPB) 指令的格式的框图；
- [0024] 图4B是图示根据实施例的加载置换和广播指令的格式的框图；
- [0025] 图5A-图5B是图示根据本发明的一些实施例的通用向量友好指令格式及其指令模板的框图；
- [0026] 图5A是图示根据本发明的一些实施例的通用向量友好指令格式及其A类指令模板的框图；
- [0027] 图5B是图示根据本发明的一些实施例的通用向量友好指令格式及其B类指令模板的框图；
- [0028] 图6A是图示根据本发明的一些实施例的示例性专用向量友好指令格式的框图；
- [0029] 图6B是图示根据一个实施例的构成完整操作码字段的具有专用向量友好指令格式的字段的框图；
- [0030] 图6C是图示根据一个实施例的构成寄存器索引字段的具有专用向量友好指令格式的字段的框图；
- [0031] 图6D是图示根据一个实施例的构成扩充操作字段的具有专用向量友好指令格式的字段的框图；
- [0032] 图7是根据一个实施例的寄存器架构的框图；
- [0033] 图8A是图示根据一些实施例的示例性有序流水线以及示例性寄存器重命名的乱序发布/执行流水线两者的框图；
- [0034] 图8B是图示根据一些实施例的要包括在处理器中的有序架构核的示例性实施例和示例性的寄存器重命名的乱序发布/执行架构核两者的框图；
- [0035] 图9A-图9B图示更具体的示例性有序核架构的框图,该核将是芯片中的若干逻辑块之一(包括相同类型和/或不同类型的其他核)；
- [0036] 图9A是根据一些实施例的单个处理器核以及它与管芯上互连网络的连接及其第二级 (L2) 高速缓存的本地子集的框图；
- [0037] 图9B是根据一些实施例的图9A中的处理器核的一部分的展开图；
- [0038] 图10是根据一些实施例的可具有多于一个的核、可具有集成存储器控制器、并且可具有集成图形的处理器的框图；
- [0039] 图11-图14是示例性计算机架构的框图；
- [0040] 图11示出根据一些实施例的系统的框图；
- [0041] 图12是根据一些实施例的更具体的第一示例性系统的框图；
- [0042] 图13是根据一些实施例的更具体的第二示例性系统的框图；
- [0043] 图14是根据一些实施例的芯片上系统 (SoC) 的框图；以及

[0044] 图15是根据一些实施例的对照使用软件指令转换器将源指令集中的二进制指令转换成目标指令集中的二进制指令的框图。

具体实施方式

[0045] 在下列描述中,阐述了众多特定细节。然而,应该理解,一些实施例可在没有这些特定细节的情况下实施。在其他实例中,未详细示出公知的电路、结构和技术,以免使对本描述的理解模糊。

[0046] 说明书中对“一个实施例”、“实施例”、“示例实施例”等的引用表明所描述的实施例可以包括特征、结构或特性,但是每个实施例可能不一定都包括该特征、结构或特性。此外,此类短语不一定是指同一个实施例。此外,当关于实施例描述特征、结构或特性时,认为影响关于如果被明确描述的其他实施例的此类特征、结构或特性是在本领域技术人员的知识范围内的。

[0047] 如上文所提及,n元组是N个有序集合,每个集合具有M个整数。如本文所使用,n元组向量可以在存储器中由包含N组元素的向量表示,每个组具有M个元素。N组元素可以使n元组向量的元素交错或者根据元素的组使它们分离。本文所公开的是从分离格式变换为交错格式的指令族,该变换是机器学习和训练应用在后向传播过程的若干阶段中所需要的。因此,机器学习和训练应用对转置的性能是敏感的并且受益于所公开的指令。

[0048] 尽管转置可以使用现有向量指令来实现,但是这样做通常是缓慢的,通常导致大的代码尺寸,并且频繁地需要复杂的软件调节。

[0049] 因此,系统和方法可以利用本文所公开的实施例的优势,其描述了在将经置换的向量数据放置到向量寄存器堆中之前对接收自存储器的向量数据进行置换的混洗网络。在一些实施例中,混洗网络用于在目的地向量寄存器的整个宽度上广播经置换的向量的部分。此外,通过在存储器读取端口与寄存器堆之间放置混洗网络,所公开的实施例实现“在进行中”变换,从而以每周期一个指令的速率处理指令,并且允许应用基本上“无成本地”受益于变换而不引发任何附加的等待时间或性能损失。

[0050] 图1A是图示根据一些实施例的用于执行加载置换或加载置换和广播指令的处理组件的框图。如所示,系统100包括存储101以存储要执行的(多个)可能具有广播的加载置换(LPPB)指令103。在一些实施例中,计算系统100是SIMD处理器,用于并发地处理诸如矩阵的紧缩数据向量的多个元素。

[0051] 在操作中,由取出电路105从存储101取出(多个)可能具有广播的加载置换(LPPB)指令103。所取出的可能具有广播的加载置换(LPPB)指令107由解码电路109解码。至少相对于图4A-4B、图5A-5B和图6A-6D进一步示出和描述的可能具有广播的加载置换(LPPB)指令格式具有用于指定操作码、目的地向量寄存器、以及具有N组元素的源向量的存储器位置的字段(此处未示出),N为正整数。操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。

[0052] 在一些实施例中,该解码包括:生成将由执行电路(诸如,执行电路117)执行的多个微操作。解码电路109还对指令后缀和前缀进行解码(如果使用)。在一些实施例中,执行电路117包括混洗电路118,混洗电路118可被包括在执行电路117外部或被并入执行电路

117中,在该程度上而言,混洗电路118是可选的。

[0053] 在一些实施例中,寄存器重命名、寄存器分配和/或调度电路113提供用于以下一项或多项的功能:1)将逻辑操作数值重命名为物理操作数值(例如,一些实施例中的寄存器别名表);2)将状态位和标志分配给经解码的指令;和3)(例如,在一些实施例中,使用预留站)调度经解码的可能具有广播的加载置换(LPPB)指令111供在指令池外部的执行电路117上执行。

[0054] 计算系统100进一步包括寄存器堆和存储器115,寄存器堆和存储器115耦合至寄存器重命名/寄存器分配/调度电路113、执行电路117、混洗电路118和写回电路119。

[0055] 寄存器重命名/寄存器分配/调度电路113和写回电路119表示可在不同时刻发生或可根本不发生的操作,在该程度上而言,它们是可选的,如由其虚线边界所指示。混洗电路118也被示出具有虚线边界,表示其可以被实现在执行电路117的外部,或者被并入执行电路117的数据路径和控制电路中。

[0056] 下面至少相对于图1B、图2A-2I、图3、图8A-8B和图9A-9B进一步描述和图示执行电路117。

[0057] 图1B是图示根据实施例的可能具有广播的加载置换(LPPB)指令的执行的框图。如所示,LPPB指令150A包括用于指定操作码(“LPPB*”)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数,操作码用于指示处理器用于:取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。

[0058] LPPB指令150A将由系统150执行,系统150包括源151(在存储器中)、目的地154(在寄存器堆的向量寄存器中)、以及执行电路152,执行电路152包括混洗电路153。

[0059] 此处的操作码助记符包括星号(*),星号用于指示操作码可以包括可以以其他方式被指定为指令字段的各种可选的修饰符,此处各种可选的修饰符包括源向量长度、源数据格式、组数N、每组的元素数量M、多位的掩码{k},多位的掩码的每个位用于控制是否允许对对应的目的地向量元素的修改、或者该对应元素是否将被掩码。指令还可以可选地指定{z}参数以控制经掩码的目的地向量元素将被归零还是合并。

1个周期每指令的LPPB指令吞吐量

[0060] 图2A是图示根据实施例的具有一个指令每周期的吞吐量的混洗流水线的框图。如所示,包括混洗流水线201的混洗电路200(其展开版本也在图2A中图示)用于从存储器202接收包含N个组的紧缩向量数据,每个组具有M个元素。例如,存储器202可以是具有用于输出高速缓存行的读取端口的第一级数据高速缓存,在一些实施例中,该高速缓存行由64字节的数据组成。

[0061] 在操作中,混洗电路200用于通过将一个或多个N元组置换并且可能广播到目的地向量来生成写入数据。取决于执行电路的执行能力,此类操作可以在一个时钟周期中发生。但是,在一些实施例中,混洗电路花费若干周期来执行那些操作,并且使用放置在存储器202与寄存器堆204之间的混洗流水线201来进行那些操作的执行以便允许每一周期接收一个新指令。如所示,混洗流水线201包括第一级201A,用于从存储器(例如从数据高速缓存的读取端口)读取包含N组元素的向量。混洗流水线201在所示出的接下来两个级201B和201C处确定元素尺寸和向量长度。此类确定可以以若干方式中的任一种作出,若干方式包括:作

为指令字段,或作为指示那些尺度的操作码前缀/后缀,如相对于图4A-4B、图5A-5B和图6A-6D所图示和描述的。元素尺寸和向量长度还可以由除指令自身之外的来指示;例如,软件可以将型号专属寄存器(MSR)编程为指示那些尺度,或者那些尺度甚至可以采取架构默认值。在接下来的级201D和201E处进行置换和广播。最后,如所示,混洗流水线在级201F处将所生成的写入数据写入目的地向量寄存器。

[0062] 在一些实施例中,故意将混洗流水线201放置在存储器读取端口与寄存器堆204之间。那样,混洗网络可以拦截来自存储器的传入的向量数据并且在将所生成的写入数据写入目的地向量寄存器之前“在进行中”执行其置换和广播操作。在一些实施例中,混洗流水线201采取多个周期来执行其置换和广播功能,但是通过在每一周期使新的LPPB指令能被接收、解码和执行的执行流水线将那些指令的执行流水线化。

[0063] 应当注意,混洗流水线级201A-201F示出为具有虚线边界以强调它们的可选性质。流水线级201A-201F中的每一个可以与其他级组合。所公开实施例不对混洗流水线的长度设置任何要求或限制,只要混洗流水线能实现和维持一个指令每周期的吞吐量。

[0064] 一周期的吞吐量或一个指令每周期的执行速率相对于缺少所公开实施例并且必须替代地将置换和广播功能执行为分开的指令的处理器提供改善的性能。鉴于所支持的一个指令每时钟周期的吞吐量,所公开实施例基本上“无成本地”提供置换和广播功能,有益地提供功能而不引发任何性能损失或附加的代码尺寸。

[0065] 图2B是图示根据实施例的可能具有广播的加载置换(LPPB)指令的执行的框图。如所示,可能具有广播的加载置换(LPPB)指令205A包括用于指定操作码(诸如LPPBWN2M16W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。

[0066] 此处,操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。指令205A将由系统205执行,系统205包括源206、目的地向量寄存器209和执行电路207,执行电路207包括混洗网络208。

[0067] 此处,操作码助记符(LPPBWN2M16W512)包括用于指示指令行为的后缀。“LPPBW”指示处理器用于执行将2组(“N2”)源元素中的每一组的16(“M16”)个元素置换到对应的目的地向量寄存器209而没有任何广播。“W”指示源向量的元素为16位或字尺寸元素。因此,源206被示出为包括两组元素A和B,其中每一组包含16个元素。“W512”指示目的地向量寄存器具有512位,其可以保存 $N \times M$ 个元素($N=2 \times M=16 \times 16$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、32位、64位和128位中的任一个。

[0068] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令205A的格式。

[0069] 至少相对于图1、图2A、图2C-2K、图3、图8A-8B和图9A-9B进一步示出和描述系统205和执行电路207。

[0070] 图2C是图示根据实施例的可能具有广播的加载置换(LPPB)指令的执行的框图。如所示,可能具有广播的加载置换(LPPB)指令210A包括用于指定操作码(诸如

LPPBBN4M8W256)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。此处,操作码指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。此处,不需要广播,因为将生成M=32个N元组。操作码进一步指示处理器用于生成M个元组并且将M个元组写入目的地向量寄存器的对应元素,M个元组中的每一个用于包括N组元素的N个对应元素。指令210A将由系统210执行,系统210包括源211、执行电路212和目的地向量寄存器214,执行电路212包括混洗网络213。

[0071] 此处,操作码助记符(LPPBBN4M8W256)包括用于指示指令行为的后缀。不具有“cast(投射)”修饰符的“LPPBB”指示处理器用于执行将N=4组元素中的每一组的M=8个元素中的每一个置换到对应的目的地向量寄存器214但没有广播。“B”指示源向量的元素为8位元素。因此,源211被示出为包括四组元素A、B、C和D,其中每一组包含8个元素。“W256”指示目的地向量寄存器具有256位($N=4 \times M=8 \times 8$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、512位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、16位、32位、64位和128位中的任一个。

[0072] 在操作中,包括混洗网络213的执行电路212用于以一个指令每周期的速率执行LPPB指令210A的多个实例。

[0073] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令210A的格式。

[0074] 至少相对于图1、图2A-2B、图2D-2K、图3、图8A-8B和图9A-9B进一步示出和描述系统210和执行电路212。

[0075] 图2D是图示根据实施例的可能具有广播的加载置换(LPPB)指令的执行的框图。如图所示,可能具有广播的加载置换(LPPB)指令215A包括用于指定操作码(诸如LPPBBN2M32W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。操作码指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。此处,不需要广播,因为将生成M=32个N元组。此处,操作码进一步指示处理器用于生成M个元组并且将M个元组写入目的地寄存器的对应元素,M个元组中的每一个用于包括N组元素的N个对应元素。指令215A将由系统215执行,系统215包括源216、执行电路217和目的地向量寄存器219,执行电路217包括混洗网络218。

[0076] 此处,操作码助记符(LPPBBN2M32W512)包括用于指示指令行为的后缀。不具有“cast”修饰符的“LPPB”指示处理器用于执行将2(“N2”)组元素中的每一组的M=32个元素中的每一个置换到对应的目的地向量寄存器219但没有广播。“B”指示源向量的元素为8位元素。因此,源216被示出为包括两组元素A和B,其中每一组包含32个元素。“W512”指示目的地向量寄存器具有512位($N=2 \times M=32 \times 8$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、16位、32位、64位和128位中的任一个。

[0077] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令215A的格式。

[0078] 至少相对于图1、图2A-2C、图2E-2K、图3、图8A-8B和图9A-9B进一步示出和描述系

统215和执行电路217。

[0079] 图2E是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的加载置换(LPPB)指令220A包括用于指定操作码(诸如LPPBCastW7N2M16W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。操作码(LPPBCastW7N2M16W512)用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。此处,操作码进一步指示所生成的写入数据用于包括基于N个源向量的所选择的元素(“W7”)生成的单个元组,所生成的单个元组将被广播到目的地向量寄存器的整个宽度。指令220A将由系统220执行,系统220包括源221、所指定的索引222(此处被设置为7)、目的地向量寄存器224和执行电路223。

[0080] 此处,操作码助记符(LPPBCastW7N2M16W512)包括用于指示指令行为的后缀。“LPPBCastW7”指示处理器用于执行将2(“N2”)组16(“M16”)个元素中的每一组的元素7置换并且广播到对应的目的地向量寄存器224。“W”指示源向量的元素为16位或字尺寸元素。因此,源221被示出为包括两组元素A和B,其中每一组包含16个元素。“W512”指示目的地向量寄存器具有512位,其可以保存 $N \times M$ 个元素($N=2 \times M=16 \times 16$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、32位、64位和128位中的任一个。

[0081] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令220A的格式。

[0082] 至少相对于图1、图2A-2D、图2F-2K、图3、图8A-8B和图9A-9B进一步示出和描述系统220和执行电路223。

[0083] 图2F是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的加载置换(LPPB)指令225A包括用于指定操作码(诸如LPPBDualCastWN2M16W512)、目的地向量寄存器、包括N组元素的源向量的存储器位置、以及要广播的索引(设置为7)的字段,N为正整数。此处,操作码指示处理器用于取出源向量,然后生成对应于源向量的M个元素中的两个元素的两个N元组并且将该两个N元组广播到目的地向量寄存器229的整个宽度。指令225A将由系统225执行,系统225包括源226、所指定的索引227、目的地向量寄存器229和执行电路228。

[0084] 此处,操作码助记符(LPPBDualCastWN2M16W512)包括用于指示指令行为的后缀。“LPPBDualCast”指示处理器用于执行将2(“N2”)组元素中的每一组的2个所选择的元素置换并且广播到对应的目的地向量寄存器229。“W”指示源向量的元素为16位或字尺寸元素。因此,源226被示出为包括两组元素A和B,其中每一组包含16个元素。“W512”指示目的地向量寄存器具有512位($N=2 \times M=16 \times 16$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、16位、32位、64位和128位中的任一个。

[0085] 至少参考图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令225A的格式。

[0086] 至少相对于图1、图2A-2E、图2G-2K、图3、图8A-8B和图9A-9B进一步示出和描述系

统225和执行电路228。

[0087] 图2G是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的加载置换(LPPB)指令230A包括用于指定操作码(诸如LPPBCastWN2M16W512)、目的地向量寄存器、包括N组元素的源向量的存储器位置、以及要广播的索引(=7)的字段,N为正整数。此处,操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器。指令230A将由系统230执行,系统230包括源231、所指定的要选择的索引232(此处被设置为8)、目的地向量寄存器234和执行电路233。

[0088] 此处,操作码助记符(LPPBCastWN2M16W512)包括用于指示指令行为的后缀。“LPPBCast”指示处理器用于执行将2(“N2”)组元素中的每一组的元素7(如由“要广播的索引”字段所指定)置换并且广播到对应的目的地向量寄存器234。“W”指示源向量的元素为16位或字尺寸元素。因此,源231被示出为包括两组元素A和B,其中每一组包含16个元素。“W512”指示目的地向量寄存器具有512位($N=2 \times M=16 \times 16$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、16位、32位、64位和128位中的任一个。

[0089] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令230A的格式。

[0090] 至少相对于图1、图2A-2F、图2H-2K、图3、图8A-8B和图9A-9B进一步示出和描述系统230和执行电路233。

[0091] 图2H是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的加载置换(LPPB)指令235A包括用于指定操作码(诸如LPPBCastWA11N2M16W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器,并且将由系统235执行,系统235包括源236、目的地向量寄存器239A-D、以及执行电路237,执行电路237包括混洗网络238。

[0092] 此处,操作码助记符(LPPBCastWA11N2M16W512)包括用于指示指令行为的后缀。“LPPBCastA11”指示处理器用于执行将2(“N2”)组元素中的每一组的所有元素置换并且广播到对应的目的地向量寄存器239A-D。“W”指示源向量的元素为16位或字尺寸元素。因此,源236被示出为包括四组元素A、B、C和D,其中每一组包含8个元素。“W512”指示目的地向量寄存器具有512位($N=4 \times M=8 \times 16$ 位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位、512位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、16位、32位、64位和128位中的任一个。

[0093] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令235A的格式。

[0094] 至少相对于图1、图2A-2G、图2I-2K、图3、图8A-8B和图9A-9B进一步示出和描述系统235和执行电路237。

[0095] 图2I是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的

加载置换 (LPPB) 指令240A包括用于指定操作码 (诸如LPPBCastW5N4M8W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器,并且将由系统240执行,系统240包括源241、所指定的索引242 (此处设置为5)、目的地244、以及执行电路243。

[0096] 此处,操作码助记符 (LPPBCastW5N4M8W512) 包括用于指示指令行为的后缀。“LPPBCast”指示处理器用于执行对源向量的元素5 (由“W5”指定) 的置换以及广播。“W”指示源向量的元素为16位或字尺寸元素。“N4”指示源向量包括N=4个组,其中每个组具有M=8个元素。因此,源241被示出为包括四组元素A、B、C和D,其中每一组包含8个元素。“W512”指示目的地向量寄存器具有512位 (N=4x M=8x 16位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、246位、512位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、16位、32位、64位和128位中的一个。

[0097] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换 (LPPB) 指令240A的格式。

[0098] 至少相对于图1、图2A-2H、图2J-2K、图3、图8A-8B和图9A-9B进一步示出和描述系统240和执行电路243。

[0099] 图2J是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的加载置换 (LPPB) 指令245A包括用于指定操作码 (诸如LPPBCastWN4M8W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器,并且将由系统245执行,系统245包括源246、目的地249、以及执行电路248、以及所指定的要广播的索引247 (为5)。

[0100] 此处,操作码助记符 (LPPBCastWN4M8W512) 包括用于指示指令行为的后缀。“LPPBCast”指示处理器用于执行对4 (“N4”) 组元素中的每一组的8 (“M8”) 个元素中的元素5 (要广播的索引) 的置换以及广播。“W”指示源向量的元素为16位或字尺寸元素。因此,源246被示出为包括四组元素A、B、C和D,其中每一组包含8个元素。“W512”指示目的地向量寄存器具有512位 (N=4x M=8x 16位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位、512位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、16位、32位、64位和128位中的任一个。

[0101] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换 (LPPB) 指令245A的格式。

[0102] 至少相对于图1、图2A-2I、图2K、图3、图8A-8B和图9A-9B进一步示出和描述系统245和执行电路247。

[0103] 图2K是图示根据实施例的加载广播指令的执行的框图。如所示,可能具有广播的加载置换 (LPPB) 指令250A包括用于指定操作码 (诸如LPPBCastWA11N4M8W512)、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数。操作码用于指示处理器用于取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中

的每一组的对应元素,并且将所生成的写入数据写入目的地向量寄存器,并且将由系统250执行,系统250包括源251、目的地254、以及执行电路252,执行电路252包括混洗网络253。

[0104] 此处,操作码助记符(LPPBCastWA11N4M8W512)包括用于指示指令行为的后缀。“LPPBCastA11”指示处理器用于执行对源向量的所有元素的置换以及广播。“W”指示源向量的元素为16位或字尺寸元素。“N4”指示源向量包括N=4个组,其中每个组具有“M8”个元素。因此,源251被示出为包括四组元素A、B、C和D,其中每一组包含8个元素。“W512”指示目的地向量寄存器具有512位(N=4x M=8x 16位)。在其他实施例中,目的地向量寄存器的宽度可以被指定为32位、64位、128位、256位、512位和1024位中的一个。在其他实施例中,元素可以具有各种不同尺寸中的任一个,包括2位、4位、8位、16位、32位、64位和128位中的一个。

[0105] 至少相对于图4A-4B、图5A-5B和图6A-6D进一步图示和描述可能具有广播的加载置换(LPPB)指令250A的格式。

[0106] 至少相对于图1、图2A-2J、图3、图8A-8B和图9A-9B进一步示出和描述系统250和执行电路252。

[0107] 图3是图示根据实施例的处理器对可能具有广播的加载置换(LPPB)指令作出响应的流程图。如所示,流程300将由处理器响应于可能具有广播的加载置换(LPPB)指令310执行,LPPB指令310包括用于指定操作码(诸如LPPB)、目的地向量寄存器以及源向量位置的字段。

[0108] 在操作301处,处理器用于使用取出电路取出指令。在操作303处,处理器用于使用解码电路对所取出的指令解码,所取出的指令具有用于指定操作码、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数,操作码用于指示处理器用于:使用存储器接口读取源向量;生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素;以及将所生成的写入数据写入目的地向量寄存器。在一些实施例中,在操作305处,处理器用于调度经解码的指令的执行。在操作307处,处理器用于使用执行电路根据操作码执行经解码的指令,执行电路具有放置在存储器与寄存器堆之间的混洗流水线,用于取出、解码、以及以一个指令每时钟周期的速率执行指令的进一步的实例。相对于图2A图示和描述此类混洗流水线的示例。在一些实施例中,在操作309处,处理器用于提交所执行的指令的结果。操作305和309各自可在不同时刻发生或可根本不发生,在该程度上而言,它们是可选的,如由其虚线边界所指示。

[0109] 图4A是图示根据实施例的可能具有广播的加载置换(LPPB)指令的格式的框图。如所示,LPPB指令400包括用于指定操作码(LPPB*) 402以及源406向量和目的地404向量的位置的字段。在指令的上下文中,源向量用于包括N组元素,N为正整数,并且操作码402用于指示处理器用于:从源位置406(存储器位置)取出源向量;生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素;以及将所生成的写入数据写入目的地向量位置404(寄存器堆中的向量寄存器)。LPPB指令400调用与图2B-2D所示的处理器行为类似的处理器行为。

[0110] LPPB指令400包括可选的若干行为修饰符,如由其虚线边界所指示。这些行为修饰符包括:源向量宽度408(即,32位、64位、128位、256位、512位、1024位等等中的一个);源元素尺寸410(即,2位、4位、8位、16位、32位、64位、128位等等);元组的数量N 412(例如,1、2、4、8或更多);以及每元组的元素数量M 414。

[0111] LPPB指令400可以进一步包括用于指定多位的掩码 {k} 416的字段,其中每个位用于控制是否允许对对应的目的地向量元素的修改、或者该对应元素是否将被掩码。指令还可以可选地指定 {z} 418参数以控制经掩码的目的地向量元素将被归零还是合并。

[0112] 操作码402的助记符被示出为包括星号(*),星号用于指示操作码可以包括可以以其他方式被指定为指令字段408、410、412、414、416和418中的一个或多个的各种可选修饰符作为前缀或后缀。

[0113] 应当注意,可选指令字段408、410、412、414、416和418中的任何一个或多个如果未被指定则可以采用默认值,或者在发布指令之前可以由软件编程,例如通过对架构型号专属寄存器(MSR)进行编程。

[0114] 相对于图4B、图5A-5B和图6A-6D进一步示出和描述LPPB指令400的格式。

[0115] 图4B是图示根据实施例的可能具有广播的加载置换指令的格式的框图。图4A与图4B之间的一个区别是,与参考图2E-2K图示和描述的执行流程类似,后一指令调用广播到目的地向量。如图所示,LPPB指令450包括用于指定操作码452(例如,由助记符LPPBCast*表示)以及源456向量和目的地454向量的位置的字段。在指令的上下文中,源向量用于包括N组元素,N为正整数,并且操作码452用于指示处理器用于:从源位置456(诸如存储器位置)取出源向量;生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素;以及将所生成的写入数据写入目的地向量位置454(诸如寄存器堆中的向量寄存器)。

[0116] LPPB指令450包括可选的若干行为修饰符,如由其虚线边界所指示。这些行为修饰符包括用于投射的索引457(即,要广播到目的地的组索引,如图所示,例如,在图2E、图2F和图2G中)、源向量宽度458(即,32位、64位、128位、256位、512位或更多中的一个)、源元素尺寸460(即,2位、4位、8位、16位、32位、64位、128位等等)、元组的数量N 462(例如,1、2、4、8或更多)、以及每元组的元素数量M 464。

[0117] LPPB指令450可以进一步包括用于指定多位的掩码 {k} 466的字段,其中每个位用于控制是否允许对对应的目的地向量元素的修改、或者该对应元素是否将被掩码。指令还可以可选地指定 {z} 468参数以控制经掩码的目的地向量元素将被归零还是合并。

[0118] 操作码452被示出为包括星号(*),星号用于指示操作码可以包括可以以其他方式被指定为指令字段457、458、460、462、464、466和468中的一个或多个的各种可选修饰符作为前缀或后缀。

[0119] 应当注意,可选指令字段457、458、460、462、464、466和468中的任何一个或多个如果未被指定则可以采用默认值,或者在发布指令之前可以由软件编程,例如通过对架构型号专属寄存器(MSR)进行编程。

[0120] 参考图4B、图5A-5B和图6A-6D进一步示出和描述LPPB指令400的格式。

指令集

[0121] 指令集可包括一种或多种指令格式。给定的指令格式可定义各种字段(例如,位的数量、位的位置)以指定将要执行的操作(例如,操作码)以及将对其执行该操作的(多个)操作数和/或(多个)其他数据字段(例如,掩码),等等。通过指令模板(或子格式)的定义来进一步分解一些指令格式。例如,可将给定指令格式的指令模板定义为具有该指令格式的字段(所包括的字段通常按照相同顺序,但是至少一些字段具有不同的位的位置,因为较少的

字段被包括)的不同子集,和/或定义为具有以不同方式进行解释的给定字段。由此,ISA的每一条指令使用给定的指令格式(并且如果经定义,则按照该指令格式的指令模板中的给定的一个指令模板)来表达,并包括用于指定操作和操作数的字段。例如,示例性ADD(加法)指令具有特定的操作码和指令格式,该特定的指令格式包括用于指定该操作码的操作码字段和用于选择操作数(源1/目的地以及源2)的操作数字段;并且该ADD指令在指令流中出现将使得在操作数字段中具有选择特定操作数的特定的内容。已经推出和/或发布了被称为高级向量扩展(AVX)(AVX1和AVX2)和利用向量扩展(VEX)编码方案的SIMD扩展集(参见例如2014年9月的英特尔®64和IA-32架构软件开发手册;并且参见2014年10月的英特尔®高级向量扩展编程参考)。

示例性指令格式

[0122] 本文中所述的(多条)指令的实施例能以不同的格式体现。另外,在下文中详述示例性系统、架构和流水线。(多条)指令的实施例可在此类系统、架构和流水线上执行,但是不限于详述的那些系统、架构和流水线。

通用向量友好指令格式

[0123] 向量友好指令格式是适于向量指令(例如,存在专用于向量操作的特定字段)的指令格式。尽管描述了其中通过向量友好指令格式支持向量和标量操作两者的实施例,但是替代实施例仅使用通过向量友好指令格式的向量操作。

[0124] 图5A-图5B是图示根据本发明的一些实施例的通用向量友好指令格式及其指令模板的框图。图5A是图示根据本发明的一些实施例的通用向量友好指令格式及其A类指令模板的框图;而图5B是图示根据本发明的一些实施例的通用向量友好指令格式及其B类指令模板的框图。具体地,针对通用向量友好指令格式500定义A类和B类指令模板,这两者都包括无存储器访问505的指令模板和存储器访问520的指令模板。在向量友好指令格式的上下文中的术语“通用”是指不束缚于任何特定指令集的指令格式。

[0125] 尽管将描述其中向量友好指令格式支持以下情况的本发明的实施例:64字节向量操作数长度(或尺寸)与32位(4字节)或64位(8字节)数据元素宽度(或尺寸)(并且由此,64字节向量由16个双字尺寸的元素组成,或者替代地由8个四字尺寸的元素组成);64字节向量操作数长度(或尺寸)与16位(2字节)或8位(1字节)数据元素宽度(或尺寸);32字节向量操作数长度(或尺寸)与32位(4字节)、64位(8字节)、16位(2字节)或8位(1字节)数据元素宽度(或尺寸);以及16字节向量操作数长度(或尺寸)与32位(4字节)、64位(8字节)、16位(2字节)、或8位(1字节)数据元素宽度(或尺寸);但是替代实施例可支持更大、更小和/或不同的向量操作数尺寸(例如,256字节向量操作数)与更大、更小或不同的数据元素宽度(例如,128位(16字节)数据元素宽度)。

[0126] 图5A中的A类指令模板包括:1)在无存储器访问505的指令模板内,示出无存储器访问的完全舍入控制型操作510的指令模板、以及无存储器访问的数据变换型操作515的指令模板;以及2)在存储器访问520的指令模板内,示出存储器访问的时效性525的指令模板和存储器访问的非时效性530的指令模板。图5B中的B类指令模板包括:1)在无存储器访问505的指令模板内,示出无存储器访问的写掩码控制的部分舍入控制型操作512的指令模板以及无存储器访问的写掩码控制的vsize型操作517的指令模板;以及2)在存储器访问520的指令模板内,示出存储器访问的写掩码控制527的指令模板。

[0127] 通用向量友好指令格式500包括以下列出的按照在图5A-5B中图示的顺序的如下字段。

[0128] 格式字段540——该字段中的特定值(指令格式标识符值)唯一地标识向量友好指令格式,并且由此标识指令在指令流中以向量友好指令格式出现。由此,该字段对于仅具有通用向量友好指令格式的指令集是不需要的,在这个意义上该字段是任选的。

[0129] 基础操作字段542——其内容区分不同的基础操作。

[0130] 寄存器索引字段544——其内容直接或者通过地址生成来指定源或目的地操作数在寄存器中或者在存储器中的位置。这些字段包括足够数量的位以从PxQ(例如,32x512、16x128、32x1024、64x1024)寄存器堆中选择N个寄存器。尽管在一个实施例中N可多达三个源寄存器和一个目的地寄存器,但是替代实施例可支持更多或更少的源和目的地寄存器(例如,可支持多达两个源,其中这些源中的一个源还用作目的地;可支持多达三个源,其中这些源中的一个源还用作目的地;可支持多达两个源和一个目的地)。

[0131] 修饰符(modifier)字段546——其内容将指定存储器访问的以通用向量指令格式出现的指令与不指定存储器访问的以通用向量指令格式出现的指令区分开;即在无存储器访问505的指令模板与存储器访问520的指令模板之间进行区分。存储器访问操作读取和/或写入到存储器层次(在一些情况下,使用寄存器中的值来指定源和/或目的地地址),而非存储器访问操作不这样(例如,源和目的地是寄存器)。尽管在一个实施例中,该字段还在三种不同的方式之间选择以执行存储器地址计算,但是替代实施例可支持更多、更少或不同的方式来执行存储器地址计算。

[0132] 扩充操作字段550——其内容区分除基础操作以外还要执行各种不同操作中的哪一个操作。该字段是针对上下文的。在一些实施例中,该字段被分成类字段568、 α 字段552和 β 字段554。扩充操作字段550允许在单条指令而非2条、3条或4条指令中执行多组共同的操作。

[0133] 比例字段560——其内容允许用于存储器地址生成(例如,用于使用($2^{\text{比例}} \times \text{索引} + \text{基址}$)的地址生成)的索引字段的内容的按比例缩放。

[0134] 位移字段562A——其内容用作存储器地址生成的一部分(例如,用于使用($2^{\text{比例}} \times \text{索引} + \text{基址} + \text{位移}$)的地址生成)。

[0135] 位移因数数字段562B(注意,位移字段562A直接在位移因数数字段562B上的并置指示使用一个或另一个)——其内容用作地址生成的一部分;它指定将按比例缩放存储器访问的尺寸(N)的位移因数——其中N是存储器访问中的字节数量(例如,用于使用($2^{\text{比例}} \times \text{索引} + \text{基址} + \text{按比例缩放的位移}$)的地址生成)。忽略冗余的低阶位,并且因此将位移因数数字段的内容乘以存储器操作数总尺寸(N)以生成将在计算有效地址中使用的最终位移。N的值由处理器硬件在运行时基于完整操作码字段574(稍后在本文中描述)和数据操纵字段554C确定。位移字段562A和位移因数数字段562B不用于无存储器访问505的指令模板和/或不同的实施例可实现这两者中的仅一个或不实现这两者中的任一个,在这个意义上,位移字段562A和位移因数数字段562B是任选的。

[0136] 数据元素宽度字段564——其内容区分将使用多个数据元素宽度中的哪一个(在一些实施例中用于所有指令;在其他实施例中只用于指令中的一些指令)。如果支持仅一个数据元素宽度和/或使用操作码的某一方面来支持数据元素宽度,则该字段是不需要的,在

这个意义上,该字段是任选的。

[0137] 写掩码字段570——其内容逐数据元素位置地控制目的地向量操作数中的数据元素位置是否反映基础操作和扩充操作的结果。A类指令模板支持合并-写掩码,而B类指令模板支持合并-写掩码和归零-写掩码两者。当合并时,向量掩码允许在执行(由基础操作和扩充操作指定的)任何操作期间保护目的地中的任何元素集免于更新;在另一实施例中,保持其中对应掩码位具有0的目的地的每一元素的旧值。相反,当归零时,向量掩码允许在执行(由基础操作和扩充操作指定的)任何操作期间使目的地中的任何元素集归零;在一个实施例中,目的地的元素在对应掩码位具有0值时被设为0。该功能的子集是控制正被执行的操作的向量长度的能力(即,从第一个到最后一个正被修改的元素的跨度),然而,被修改的元素不一定要是连续的。由此,写掩码字段570允许部分向量操作,这包括加载、存储、算术、逻辑等。尽管描述了其中写掩码字段570的内容选择了多个写掩码寄存器中的包含要使用的写掩码的一个写掩码寄存器(并且由此,写掩码字段570的内容间接地标识要执行的掩码)的本发明的实施例,但是替代实施例替代地或附加地允许掩码写字段570的内容直接指定要执行的掩码。

[0138] 立即数字段572——其内容允许对立即数的指定。该字段在实现不支持立即数的通用向量友好格式中不存在且在不使用立即数的指令中不存在,在这个意义上,该字段是任选的。

[0139] 类字段568——其内容在不同类的指令之间进行区分。参考图5A-图5B,该字段的内容在A类和B类指令之间进行选择。在图5A-图5B中,圆角方形用于指示特定的值存在于字段中(例如,在图5A-图5B中分别用于类字段568的A类568A和B类568B)。

A类指令模板

[0140] 在A类非存储器访问505的指令模板的情况下, α 字段552被解释为其内容区分要执行不同扩充操作类型中的哪一种(例如,针对无存储器访问的舍入型操作510和无存储器访问的数据变换型操作515的指令模板分别指定舍入552A.1和数据变换552A.2)的RS字段552A,而 β 字段554区分要执行所指定类型的操作中的哪一种。在无存储器访问505的指令模板中,比例字段560、位移字段562A和位移比例字段562B不存在。

无存储器访问的指令模板——完全舍入控制型操作

[0141] 在无存储器访问的完全舍入控制型操作510的指令模板中, β 字段554被解释为其(多个)内容提供静态舍入的舍入控制字段554A。尽管在本发明的所述实施例中舍入控制字段554A包括抑制所有浮点异常(SAE)字段556和舍入操作控制字段558,但是替代实施例可支持这两个概念,可将这两个概念编码为同一字段,或仅具有这些概念/字段中的一个或另一个(例如,可仅具有舍入操作控制字段558)。

[0142] SAE字段556——其内容区分是否禁用异常事件报告;当SAE字段556的内容指示启用抑制时,给定的指令不报告任何种类的浮点异常标志,并且不唤起任何浮点异常处置程序。

[0143] 舍入操作控制字段558——其内容区分要执行一组舍入操作中的哪一个(例如,向上舍入、向下舍入、向零舍入以及就近舍入)。由此,舍入操作控制字段558允许逐指令地改变舍入模式。在其中处理器包括用于指定舍入模式的控制寄存器的一些实施例中,舍入操作控制字段550的内容覆盖(override)该寄存器值。

无存储器访问的指令模板—数据变换型操作

[0144] 在无存储器访问的数据变换型操作515的指令模板中,β字段554被解释为数据变换字段554B,其内容区分要执行多个数据变换中的哪一个(例如,无数据变换、混合、广播)。

[0145] 在A类存储器访问520的指令模板的情况下,α字段552被解释为驱逐提示字段552B,其内容区分要使用驱逐提示中的哪一个(在图5A中,对于存储器访问时效性525的指令模板和存储器访问非时效性530的指令模板分别指定时效性的552B.1和非时效性的552B.2),而β字段554被解释为数据操纵字段554C,其内容区分要执行多个数据操纵操作(也称为基元(primitive))中的哪一个(例如,无操纵、广播、源的向上转换以及目的地的向下转换)。存储器访问520的指令模板包括比例字段560,并任选地包括位移字段562A或位移比例字段562B。

[0146] 向量存储器指令使用转换支持来执行来自存储器的向量加载以及向存储器的向量存储。如同寻常的向量指令,向量存储器指令以数据元素式的方式从/向存储器传输数据,其中实际被传输的元素由被选为写掩码的向量掩码的内容规定。

存储器访问的指令模板—时效性的

[0147] 时效性的数据是可能足够快地被重新使用以从高速缓存操作受益的数据。然而,这是提示,并且不同的处理器能以不同的方式实现它,包括完全忽略该提示。

存储器访问的指令模板—非时效性的

[0148] 非时效性的数据是不太可能足够快地被重新使用以从第一级高速缓存中的高速缓存操作受益且应当被给予驱逐优先级的数据。然而,这是提示,并且不同的处理器能以不同的方式实现它,包括完全忽略该提示。

B类指令模板

[0149] 在B类指令模板的情况下,α字段552被解释为写掩码控制(Z)字段552C,其内容区分由写掩码字段570控制的写掩码应当是合并还是归零。

[0150] 在B类非存储器访问505的指令模板的情况下,β字段554的一部分被解释为RL字段557A,其内容区分要执行不同扩充操作类型中的哪一种(例如,针对无存储器访问的写掩码控制部分舍入控制类型操作512的指令模板和无存储器访问的写掩码控制VSIZE型操作517的指令模板分别指定舍入557A.1和向量长度(VSIZE)557A.2),而β字段的其余部分区分要执行所指定类型的操作中的哪一种。在无存储器访问505的指令模板中,比例字段560、位移字段562A和位移比例字段562B不存在。

[0151] 在无存储器访问的写掩码控制部分舍入控制型操作510的指令模板中,β字段的其余部分被解释为舍入操作字段559A,并且禁用异常事件报告(给定的指令不报告任何种类的浮点异常标志,并且不唤起任何浮点异常处置程序)。

[0152] 舍入操作控制字段559A——正如舍入操作控制字段558,其内容区分要执行一组舍入操作中的哪一个(例如,向上舍入、向下舍入、向零舍入以及就近舍入)。由此,舍入操作控制字段559A允许逐指令地改变舍入模式。在其中处理器包括用于指定舍入模式的控制寄存器的一些实施例中,舍入操作控制字段550的内容覆盖该寄存器值。

[0153] 在无存储器访问的写掩码控制VSIZE型操作517的指令模板中,β字段的其余部分被解释为向量长度字段559B,其内容区分要执行多个数据向量长度中的哪一个(例如,128字节、256字节或512字节)。

[0154] 在B类存储器访问520的指令模板的情况下,β字段554的一部分被解释为广播字段557B,其内容区分是否要执行广播型数据操纵操作,而β字段554的其余部分被解释为向量长度字段559B。存储器访问520的指令模板包括比例字段560,并任选地包括位移字段562A或位移比例字段562B。

[0155] 针对通用向量友好指令格式500,示出完整操作码字段574包括格式字段540、基础操作字段542和数据元素宽度字段564。尽管示出了其中完整操作码字段574包括所有这些字段的一个实施例,但是在不支持所有这些字段的实施例中,完整操作码字段574包括少于所有的这些字段。完整操作码字段574提供操作代码(操作码)。

[0156] 扩充操作字段550、数据元素宽度字段564和写掩码字段570允许逐指令地以通用向量友好指令格式指定这些特征。

[0157] 写掩码字段和数据元素宽度字段的组合创建各种类型的指令,因为这些指令允许基于不同的数据元素宽度应用该掩码。

[0158] 在A类和B类内出现的各种指令模板在不同的情形下是有益的。在本发明的一些实施例中,不同处理器或处理器内的不同核可支持仅A类、仅B类、或者可支持这两类。举例而言,旨在用于通用计算的高性能通用乱序核可仅支持B类,旨在主要用于图形和/或科学(吞吐量)计算的核可仅支持A类,并且旨在用于通用计算和图形和/或科学(吞吐量)计算两者的核可支持A类和B类两者(当然,具有来自这两类的模板和指令的一些混合、但是并非来自这两类的所有模板和指令的核在本发明的范围内)。同样,单个处理器可包括多个核,这多个核全部都支持相同的类,或者其中不同的核支持不同的类。举例而言,在具有单独的图形核和通用核的处理器中,图形核中的旨在主要用于图形和/或科学计算的一个核可仅支持A类,而通用核中的一个或多个可以是具有旨在用于通用计算的仅支持B类的乱序执行和寄存器重命名的高性能通用核。不具有单独的图形核的另一处理器可包括既支持A类又支持B类的一个或多个通用有序或乱序核。当然,在本发明的不同实施例中,来自一类的特征也可在其他类中实现。将使以高级语言编写的程序成为(例如,及时编译或静态编译)各种不同的可执行形式,这些可执行形式包括:1) 仅具有由用于执行的目标处理器支持的(多个)类的指令的形式;或者2) 具有替代例程并具有控制流代码的形式,该替代例程使用所有类的指令的不同组合来编写,该控制流代码选择这些例程以基于由当前正在执行代码的处理器支持的指令来执行。

示例性专用向量友好指令格式

[0159] 图6A是图示根据本发明的一些实施例的示例性专用向量友好指令格式的框图。图6A示出专用向量友好指令格式600,其指定各字段的位置、尺寸、解释和次序、以及那些字段中的一些字段的值,在这个意义上,该专用向量友好指令格式600是专用的。专用向量友好指令格式600可用于扩展x86指令集,并且由此字段中的一些字段与如在现有的x86指令集及其扩展(例如,AVX)中所使用的那些字段类似或相同。该格式保持与具有扩展的现有x86指令集的前缀编码字段、实操作码字节字段、MOD R/M字段、SIB字段、位移字段和立即数字段一致。图示来自图5的字段,来自图6A的字段映射到来自图5的字段。

[0160] 应当理解,虽然出于说明的目的在通用向量友好指令格式500的上下文中参考专用向量友好指令格式600描述了本发明的实施例,但是本发明不限于专用向量友好指令格式600,除非另有声明。例如,通用向量友好指令格式500构想了各种字段的各种可能的尺

寸,而专用向量友好指令格式600示出为具有特定尺寸的字段。作为具体示例,尽管在专用向量友好指令格式600中数据元素宽度字段564被图示为一位字段,但是本发明不限于此(即,通用向量友好指令格式500构想数据元素宽度字段564的其他尺寸)。

[0161] 通用向量友好指令格式500包括以下列出的按照图6A中图示的顺序的如下字段。

[0162] EVEX前缀(字节0-3)602——以四字节形式进行编码。

[0163] 格式字段540(EVEX字节0,位[7:0])——第一字节(EVEX字节0)是格式字段540,并且它包含0x62(在一些实施例中,为用于区分向量友好指令格式的唯一值)。

[0164] 第二—第四字节(EVEX字节1-3)包括提供专用能力的多个位字段。

[0165] REX字段605(EVEX字节1,位[7-5])——由EVEX.R位字段(EVEX字节1,位[7]-R)、EVEX.X位字段(EVEX字节1,位[6]-X)以及(557BEX字节1,位[5]-B)组成。EVEX.R、EVEX.X和EVEX.B位字段提供与对应的VEX位字段相同的功能,并且使用1补码的形式进行编码,即ZMM0被编码为1111B,ZMM15被编码为0000B。这些指令的其他字段对如在本领域中已知的寄存器索引的较低三个位(rrr、xxx和bbb)进行编码,由此可通过对EVEX.R、EVEX.X和EVEX.B相加来形成Rrrr、Xxxx和Bbbb。

[0166] REX' 字段610A——这是REX' 字段610的第一部分,并且是用于对扩展的32个寄存器集合的较高16个或较低16个寄存器进行编码的EVEX.R' 位字段(EVEX字节1,位[4]-R')。在一些实施例中,该位与以下指示的其他位一起以位反转的格式存储以(在公知x86的32位模式下)与BOUND指令进行区分,该BOUND指令的实操作码字节是62,但是在MOD R/M字段(在下文中描述)中不接受MOD字段中的值11;本发明的替代实施例不以反转的格式存储该指示的位以及以下其他指示的位。值1用于对较低16个寄存器进行编码。换句话说,通过组合EVEX.R'、EVEX.R以及来自其他字段的其他RRR来形成R' Rrrr。

[0167] 操作码映射字段615(EVEX字节1,位[3:0]-mmmm)——其内容对隐含的前导操作码字节(0F、0F 38或0F 3)进行编码。

[0168] 数据元素宽度字段564(EVEX字节2,位[7]-W)——由记号EVEX.W表示。EVEX.W用于定义数据类型(32位数据元素或64位数据元素)的粒度(尺寸)。

[0169] EVEX.vvvv 620(EVEX字节2,位[6:3]-vvvv)——EVEX.vvvv的作用可包括如下:1)EVEX.vvvv对以反转(1补码)形式指定的第一源寄存器操作数进行编码,并且对具有两个或更多个源操作数的指令有效;2)EVEX.vvvv对针对特定向量位移以1补码的形式指定的目的地寄存器操作数进行编码;或者3)EVEX.vvvv不对任何操作数进行编码,该字段被预留,并且应当包含1111b。由此,EVEX.vvvv字段620对以反转(1补码)的形式存储的第一源寄存器指定符的4个低阶位进行编码。取决于该指令,额外不同的EVEX位字段用于将指定符尺寸扩展到32个寄存器。

[0170] EVEX.U 568类字段(EVEX字节2,位[2]-U)——如果EVEX.U=0,则它指示A类或EVEX.U0;如果EVEX.U=1,则它指示B类或EVEX.U1。

[0171] 前缀编码字段625(EVEX字节2,位[1:0]-pp)——提供了用于基础操作字段的附加位。除了对以EVEX前缀格式的传统SSE指令提供支持以外,这也具有压缩SIMD前缀的益处(EVEX前缀仅需要2位,而不是需要字节来表达SIMD前缀)。在一个实施例中,为了支持使用以传统格式和以EVEX前缀格式两者的SIMD前缀(66H、F2H、F3H)的传统SSE指令,将这些传统SIMD前缀编码成SIMD前缀编码字段;并且在运行时在被提供给解码器的PLA之前被扩展成

传统SIMD前缀(因此,在无需修改的情况下,PLA既可执行传统格式的这些传统指令又可执行EVEX格式的这些传统指令)。虽然较新的指令可将EVEX前缀编码字段的内容直接用作操作码扩展,但是为了一致性,特定实施例以类似的方式扩展,但允许由这些传统SIMD前缀指定的不同含义。替代实施例可重新设计PLA以支持2位SIMD前缀编码,并且由此不需要扩展。

[0172] α 字段552 (EVEX字节3,位[7]-EH,也称为EVEX.EH、EVEX.rs、EVEX.RL、EVEX.写掩码控制、以及EVEX.N;也以 α 图示)——如先前所述,该字段是针对上下文的。

[0173] β 字段554 (EVEX字节3,位[6:4]-SSS,也称为EVEX.s₂₋₀、EVEX.r₂₋₀、EVEX.rr1、EVEX.LL0、EVEX.LLB,还以 $\beta\beta\beta$ 图示)——如前所述,此字段是针对上下文的。

[0174] REX' 字段610B——这是REX' 字段610的其余部分,并且是可用于对扩展的32个寄存器集合的较高16个或较低16个寄存器进行编码的EVEX.V' 位字段 (EVEX字节3,位[3]-V')。该位以位反转的格式存储。值1用于对较低16个寄存器进行编码。换句话说,通过组合EVEX.V'、EVEX.vvvv来形成V' VVVV。

[0175] 写掩码字段570 (EVEX字节3,位[2:0]-kkk)——其内容指定写掩码寄存器中的寄存器的索引,如先前所述。在一些实施例中,特定值EVEX.kkk=000具有暗示没有写掩码用于特定指令的特殊行为(这能以各种方式实现,包括使用硬连线到所有对象的写掩码或绕过掩码硬件的硬件来实现)。

[0176] 实操作码字段630 (字节4) 还被称为操作码字节。操作码的一部分在该字段中被指定。

[0177] MOD R/M字段640 (字节5) 包括MOD字段642、Reg字段644和R/M字段646。如先前所述的,MOD字段642的内容将存储器访问操作和非存储器访问操作区分开。Reg字段644的作用可被归结为两种情形:对目的地寄存器操作数或源寄存器操作数进行编码;或者被视为操作码扩展,并且不用于对任何指令操作数进行编码。R/M字段646的作用可包括如下:对引用存储器地址的指令操作数进行编码;或者对目的地寄存器操作数或源寄存器操作数进行编码。

[0178] 比例、索引、基址 (SIB) 字节 (字节6) ——如先前所述的,比例字段550的内容用于存储器地址生成。SIB.xxx 654和SIB.bbb 656——先前已经针对寄存器索引Xxxx和Bbbb提及了这些字段的内容。

[0179] 位移字段562A (字节7-10) ——当MOD字段642包含10时,字节7-10是位移字段562A,并且它与传统32位位移 (disp32) 一样地工作,并且以字节粒度工作。

[0180] 位移因数字段562B (字节7) ——当MOD字段642包含01时,字节7是位移因数字段562B。该字段的位置与以字节粒度工作的传统x86指令集8位位移 (disp8) 的位置相同。由于disp8是符号扩展的,因此它仅能在-128和127字节偏移之间寻址;在64字节高速缓存行的方面,disp8使用可被设为仅四个真正有用的值-128、-64、0和64的8位;由于常常需要更大的范围,所以使用disp32;然而,disp32需要4个字节。与disp8和disp32对比,位移因数字段562B是disp8的重新解释;当使用位移因数字段562B时,通过将位移因数字段的内容乘以存储器操作数访问的尺寸(N)来确定实际位移。该类型的位移被称为disp8*N。这减小了平均指令长度(单个字节用于位移,但具有大得多的范围)。此类经压缩的位移基于有效位移是存储器访问的粒度的倍数的假设,并且由此地址偏移的冗余低阶位不需要被编码。换句话说,位移因数字段562B替代传统x86指令集8位位移。由此,位移因数字段562B以与x86指令

集8位位移相同的方式被编码(因此,在ModRM/SIB编码规则中没有变化),唯一的不同在于,将disp8超载至disp8*N。换句话说,在编码规则或编码长度方面没有变化,而仅在在有硬件对位移值的解释方面有变化(这需要将位移按比例缩放存储器操作数的尺寸以获得字节式地址偏移)。立即数字段572如先前所述地操作。

完整操作码字段

[0181] 图6B是图示根据一些实施例的构成完整操作码字段574的具有专用向量友好指令格式600的字段的框图。具体地,完整操作码字段574包括格式字段540、基础操作字段542和数据元素宽度(W)字段564。基础操作字段542包括前缀编码字段625、操作码映射字段615和实操作码字段630。

寄存器索引字段

[0182] 图6C是图示根据一些实施例的构成寄存器索引字段544的具有专用向量友好指令格式600的字段的框图。具体地,寄存器索引字段544包括REX字段605、REX' 字段610、MODR/M.reg字段644、MODR/M.r/m字段646、VVVV字段620、xxx字段654和bbb字段656。

扩充操作字段

[0183] 图6D是图示根据一些实施例的构成扩充操作字段550的具有专用向量友好指令格式600的字段的框图。当类(U)字段568包含0时,它表明EVEX.U0(A类568A);当它包含1时,它表明EVEX.U1(B类568B)。当U=0且MOD字段642包含11(表明无存储器访问操作)时, α 字段552(EVEX字节3,位[7]-EH)被解释为rs字段552A。当rs字段552A包含1(舍入552A.1)时, β 字段554(EVEX字节3,位[6:4]-SSS)被解释为舍入控制字段554A。舍入控制字段554A包括一位SAE字段556和两位舍入操作字段558。当rs字段552A包含0(数据变换552A.2)时, β 字段554(EVEX字节3,位[6:4]-SSS)被解释为三位数据变换字段554B。当U=0且MOD字段642包含00、01或10(表明存储器访问操作)时, α 字段552(EVEX字节3,位[7]-EH)被解释为驱逐提示(EH)字段552B,并且 β 字段554(EVEX字节3,位[6:4]-SSS)被解释为三位数据操纵字段554C。

[0184] 当U=1时, α 字段552(EVEX字节3,位[7]-EH)被解释为写掩码控制(Z)字段552C。当U=1且MOD字段642包含11(表明无存储器访问操作)时, β 字段554的一部分(EVEX字节3,位[4]-S₀)被解释为RL字段557A;当它包含1(舍入557A.1)时, β 字段554的其余部分(EVEX字节3,位[6-5]-S₂₋₁)被解释为舍入操作字段559A,而当RL字段557A包含0(VSIZE 557.A2)时, β 字段554的其余部分(EVEX字节3,位[6-5]-S₂₋₁)被解释为向量长度字段559B(EVEX字节3,位[6-5]-L₁₋₀)。当U=1且MOD字段642包含00、01或10(表明存储器访问操作)时, β 字段554(EVEX字节3,位[6:4]-SSS)被解释为向量长度字段559B(EVEX字节3,位[6-5]-L₁₋₀)和广播字段557B(EVEX字节3,位[4]-B)。

示例性寄存器架构

[0185] 图7是根据一些实施例的寄存器架构700的框图。在所图示的实施例中,有32个512位宽的向量寄存器710;这些寄存器被引用为zmm0到zmm31。较低的16个zmm寄存器的较低阶256个位覆盖(overlay)在寄存器ymm0-16上。较低的16个zmm寄存器的较低阶128个位(ymm寄存器的较低阶128个位)覆盖在寄存器xmm0-15上。专用向量友好指令格式600对这些被覆盖的寄存器堆操作,如在以下表格中所图示。

可调节向量长度	类	操作	寄存器
不包括向量长度字段 559B 的指令模板	A (图 5A; U=0)	510、515、525、530	zmm 寄存器(向量长度是 64 字节)
	B (图 5B; U=1)	512	zmm 寄存器(向量长度是 64 字节)
包括向量长度字段 559B 的指令模板	B (图 5B; U=1)	517、527	zmm、ymm、或 xmm 寄存器(向量长度是 64 字节、32 字节、或 16 字节), 取决于向量长度字段 559B

[0186] 换句话说,向量长度字段559B在最大长度与一个或多个其他较短长度之间进行选择,其中每一个此类较短长度是前一长度的一半,并且不具有向量长度字段559B的指令模板在最大向量长度上操作。此外,在一个实施例中,专用向量友好指令格式600的B类指令模板对紧缩或标量单/双精度浮点数据以及紧缩或标量整数数据操作。标量操作是对zmm/ymm/xmm寄存器中的最低阶数据元素位置执行的操作;取决于实施例,较高阶数据元素位置要么保持与在指令之前相同,要么归零。

[0187] 写掩码寄存器715——在所图示的实施例中,存在8个写掩码寄存器(k0至k7),每一写掩码寄存器的尺寸是64位。在替代实施例中,写掩码寄存器715的尺寸是16位。如先前所述,在一些实施例中,向量掩码寄存器k0无法用作写掩码;当将正常指示k0的编码用作写掩码时,它选择硬连线的写掩码0xFFFF,从而有效地禁止写掩码用于那条指令。

[0188] 通用寄存器725——在所示出的实施例中,有十六个64位通用寄存器,这些寄存器与现有的x86寻址模式一起使用以对存储器操作数寻址。这些寄存器通过名称RAX、RBX、RCX、RDX、RBP、RSI、RDI、RSP以及R8到R15来引用。

[0189] 标量浮点栈寄存器堆(x87栈)745,在其上面重叠了MMX紧缩整数平坦寄存器堆750——在所图示的实施例中,x87栈是用于使用x87指令集扩展来对32/64/80位浮点数据执行标量浮点操作的八元素栈;而使用MMX寄存器来对64位紧缩整数数据执行操作,以及为在MMX与XMM寄存器之间执行的一些操作保存操作数。

[0190] 替代实施例可以使用更宽的或更窄的寄存器。另外,替代实施例可以使用更多、更少或不同的寄存器堆和寄存器。

示范性核架构、处理器和计算机架构

[0191] 处理器核能以不同方式、出于不同的目的、在不同的处理器中实现。例如,此类核的实现可以包括:1)旨在用于通用计算的通用有序核;2)旨在用于通用计算的高性能通用乱序核;3)旨在主要用于图形和/或科学(吞吐量)计算的专用核。不同处理器的实现可包括:1)CPU,其包括旨在用于通用计算的一个或多个通用有序核和/或旨在用于通用计算的一个或多个通用乱序核;以及2)协处理器,其包括旨在主要用于图形和/或科学(吞吐量)的

一个或多个专用核。此类不同的处理器导致不同的计算机系统架构,这些计算机系统架构可包括:1)在与CPU分开的芯片上的协处理器;2)在与CPU相同的封装中但在分开的管芯上的协处理器;3)与CPU在相同管芯上的协处理器(在该情况下,此类协处理器有时被称为专用逻辑或被称为专用核,该专用逻辑诸如,集成图形和/或科学(吞吐量)逻辑);以及4)芯片上系统,其可以将所描述的CPU(有时被称为(多个)应用核或(多个)应用处理器)、以上描述的协处理器和附加功能包括在同一管芯上。接着描述示例性核架构,随后描述示例性处理器和计算机架构。

示例性核架构

有序和乱序核框图

[0192] 图8A是图示根据本发明的一些实施例的示例性有序流水线和示例性的寄存器重命名的乱序发布/执行流水线的框图。图8B是示出根据本发明的一些实施例的要包括在处理器中的有序架构核的示例性实施例和示例性的寄存器重命名的乱序发布/执行架构核的框图。图8A-图8B中的实线框图示有序流水线和有序核,而虚线框的任选增加图示寄存器重命名的、乱序发布/执行流水线和核。考虑到有序方面是乱序方面的子集,将描述乱序方面。

[0193] 在图8A中,处理器流水线800包括取出级802、长度解码级804、解码级806、分配级808、重命名级810、调度(也被称为分派或发布)级812、寄存器读取/存储器读取级814、执行级816、写回/存储器写入级818、异常处置级822和提交级824。

[0194] 图8B示出处理器核890,该处理器核890包括前端单元830,该前端单元830耦合到执行引擎单元850,并且前端单元830和执行引擎单元850两者都耦合到存储器单元870。核890可以是精简指令集计算(RISC)核、复杂指令集计算(CISC)核、超长指令字(VLIW)核、或混合或替代的核类型。作为又一选项,核890可以是专用核,诸如例如,网络或通信核、压缩引擎、协处理器核、通用计算图形处理单元(GPGPU)核、图形核,等等。

[0195] 前端单元830包括分支预测单元832,该分支预测单元832耦合到指令高速缓存单元834,该指令高速缓存单元834耦合到指令转换后备缓冲器(TLB)836,该指令转换后备缓冲器836耦合到指令取出单元838,该指令取出单元838耦合到解码单元840。解码单元840(或解码器)可对指令解码,并且生成从原始指令解码出的、或以其他方式反映原始指令的、或从原始指令导出的一个或多个微操作、微代码进入点、微指令、其他指令、或其他控制信号作为输出。解码单元840可使用各种不同的机制来实现。合适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。在一个实施例中,核890包括存储用于某些宏指令的微代码的微代码ROM或其他介质(例如,在解码单元840中,或以其他方式在前端单元830内)。解码单元840耦合到执行引擎单元850中的重命名/分配器单元852。

[0196] 执行引擎单元850包括重命名/分配器单元852,该重命名/分配器单元852耦合到引退单元854和一个或多个调度器单元的集合856。(多个)调度器单元856表示任何数量的不同调度器,包括预留站、中央指令窗等。(多个)调度器单元856耦合到(多个)物理寄存器堆单元858。(多个)物理寄存器堆单元858中的每一个物理寄存器堆单元表示一个或多个物理寄存器堆,其中不同的物理寄存器堆存储一种或多种不同的数据类型,诸如,标量整数、标量浮点、紧缩整数、紧缩浮点、向量整数、向量浮点,状态(例如,作为要执行的下一条指令的地址的指令指针)等等。在一个实施例中,(多个)物理寄存器堆单元858包括向量寄存器

单元、写掩码寄存器单元和标量寄存器单元。这些寄存器单元可以提供架构向量寄存器、向量掩码寄存器和通用寄存器。(多个)物理寄存器堆单元858由引退单元854重叠,以图示可实现寄存器重命名和乱序执行的各种方式(例如,使用(多个)重排序缓冲器和(多个)引退寄存器堆;使用(多个)未来文件、(多个)历史缓冲器、(多个)引退寄存器堆;使用寄存器映射和寄存器池,等等)。引退单元854和(多个)物理寄存器堆单元858耦合到(多个)执行集群860。(多个)执行集群860包括一个或多个执行单元的集合862以及一个或多个存储器访问单元的集合864。执行单元862可执行各种操作(例如,移位、加法、减法、乘法)并可对各种数据类型(例如,标量浮点、紧缩整数、紧缩浮点、向量整数、向量浮点)执行。尽管一些实施例可以包括专用于特定功能或功能集合的多个执行单元,但是其他实施例可包括仅一个执行单元或全都执行所有功能的多个执行单元。(多个)调度器单元856、(多个)物理寄存器堆单元858和(多个)执行集群860示出为可能有多,因为某些实施例为某些类型的数据/操作创建分开的流水线(例如,标量整数流水线、标量浮点/紧缩整数/紧缩浮点/向量整数/向量浮点流水线,和/或各自具有其自身的调度器单元、(多个)物理寄存器堆单元和/或执行集群的存储器访问流水线——并且在分开的存储器访问流水线的情况下,实现其中仅该流水线的执行集群具有(多个)存储器访问单元864的某些实施例)。还应当理解,在使用分开的流水线的情况下,这些流水线中的一个或多个可以是乱序发布/执行,并且其余流水线可以是有序的。

[0197] 存储器访问单元的集合864耦合到存储器单元870,该存储器单元870包括数据TLB单元872,该数据TLB单元872耦合到数据高速缓存单元874,该数据高速缓存单元874耦合到第二级(L2)高速缓存单元876。在一个示例性实施例中,存储器访问单元864可包括加载单元、存储地址单元和存储数据单元,其中的每一个均耦合到存储器单元870中的数据TLB单元872。指令高速缓存单元834还耦合到存储器单元870中的第二级(L2)高速缓存单元876。L2高速缓存单元876耦合到一个或多个其他级别的高速缓存,并最终耦合到主存储器。

[0198] 作为示例,示例性寄存器重命名的乱序发布/执行核架构可如下所述地实现流水线800:1)指令取出838执行取出级802和长度解码级804;2)解码单元840执行解码级806;3)重命名/分配器单元852执行分配级808和重命名级810;4)(多个)调度器单元856执行调度级812;5)(多个)物理寄存器堆单元858和存储器单元870执行寄存器读取/存储器读取级814;执行集群860执行执行级816;6)存储器单元870和(多个)物理寄存器堆单元858执行写回/存储器写入级818;7)各单元可牵涉到异常处置级822;以及8)引退单元854和(多个)物理寄存器堆单元858执行提交级824。

[0199] 核890可支持一个或多个指令集(例如,x86指令集(具有已与较新版本一起添加的一些扩展);加利福尼亚州桑尼维尔市的MIPS技术公司的MIPS指令集;加利福尼亚州桑尼维尔市的ARM控股公司的ARM指令集(具有诸如NEON的任选的附加扩展)),其中包括本文中描述的(多条)指令。在一个实施例中,核890包括用于支持紧缩数据指令集扩展(例如,AVX1、AVX2)的逻辑,由此允许使用紧缩数据来执行由许多多媒体应用使用的操作。

[0200] 应当理解,核可支持多线程化(执行两个或更多个并行的操作或线程的集合),并且可以按各种方式来完成该多线程化,各种方式包括时分多线程化、同时多线程化(其中单个物理核为物理核正在同时多线程化的线程中的每一个线程提供逻辑核)、或其组合(例如,时分取出和解码以及此后的诸如英特尔®超线程化技术中的同时多线程化)。

[0201] 尽管在乱序执行的上下文中描述了寄存器重命名,但应当理解,可以在有序架构中使用寄存器重命名。尽管所图示的处理器实施例还包括分开的指令和数据高速缓存单元834/874以及共享的L2高速缓存单元876,但是替代实施例可以具有用于指令和数据两者的单个内部高速缓存,诸如例如,第一级(L1)内部高速缓存或多个级别的内部高速缓存。在一些实施例中,该系统可包括内部高速缓存和在核和/或处理器外部的的外部高速缓存的组合。或者,所有高速缓存都可以在核和/或处理器的外部。

具体的示例性有序核架构

[0202] 图9A-图9B图示更具体的示例性有序核架构的框图,该核将是芯片中的若干逻辑块(包括相同类型和/或不同类型的其他核)中的一个逻辑块。取决于应用,逻辑块通过高带宽互连网络(例如,环形网络)与一些固定的功能逻辑、存储器I/O接口和其他必要的I/O逻辑进行通信。

[0203] 图9A是根据本发明的一些实施例的单个处理器核以及它至管芯上互连网络902的连接及其第二级(L2)高速缓存的本地子集904的框图。在一个实施例中,指令解码器900支持具有紧缩数据指令集扩展的x86指令集。L1高速缓存906允许对进入标量和向量单元中的、对高速缓存存储器的低等待时间访问。尽管在一个实施例中(为了简化设计),标量单元908和向量单元910使用分开的寄存器集合(分别为标量寄存器912和向量寄存器914),并且在这些寄存器之间传输的数据被写入到存储器,并随后从第一级(L1)高速缓存906读回,但是本发明的替代实施例可以使用不同的方法(例如,使用单个寄存器集合或包括允许数据在这两个寄存器堆之间传输而无需被写入和读回的通信路径)。

[0204] L2高速缓存的本地子集904是全局L2高速缓存的一部分,该全局L2高速缓存被划分成多个分开的本地子集,每个处理器核一个本地子集。每个处理器核具有到其自身的L2高速缓存的本地子集904的直接访问路径。由处理器核读取的数据被存储在其L2高速缓存子集904中,并且可以与其他处理器核访问其自身的本地L2高速缓存子集并行地被快速访问。由处理器核写入的数据被存储在其自身的L2高速缓存子集904中,并在必要的情况下从其他子集转储清除。环形网络确保共享数据的一致性。环形网络是双向的,以允许诸如处理器核、L2高速缓存和其他逻辑块之类的代理在芯片内彼此通信。每个环形数据路径为每个方向1012位宽。

[0205] 图9B是根据本发明的一些实施例的图9A中的处理器核的一部分的展开图。图9B包括L1高速缓存904的L1数据高速缓存906A部分,以及关于向量单元910和向量寄存器914的更多细节。具体地,向量单元910是16宽向量处理单元(VPU)(见16宽ALU 928),该单元执行整数、单精度浮点以及双精度浮点指令中的一个或多个。该VPU通过混合单元920支持对寄存器输入的混合,通过数值转换单元922A-B支持数值转换,并且通过复制单元924支持对存储器输入的复制。写掩码寄存器926允许掩蔽所得的向量写入。

[0206] 图10是根据本发明的一些实施例的可具有多于一个的核、可具有集成存储器控制器、以及可具有集成图形器件的处理器1000的框图。图10中的实线框图示具有单个核1002A、系统代理1010、一个或多个总线控制器单元的集合1016的处理器1000,而虚线框的任选增加图示具有多个核1002A-N、系统代理单元1010中的一个或多个集成存储器控制器单元的集合1014以及专用逻辑1008的替代处理器1000。

[0207] 因此,处理器1000的不同实现可包括:1)CPU,其中专用逻辑1008是集成图形和/或

科学(吞吐量)逻辑(其可包括一个或多个核),并且核1002A-N是一个或多个通用核(例如,通用有序核、通用乱序核、这两者的组合);2)协处理器,其中核1002A-N是旨在主要用于图形和/或科学(吞吐量)的大量专用核;以及3)协处理器,其中核1002A-N是大量通用有序核。因此,处理器1000可以是通用处理器、协处理器或专用处理器,诸如例如,网络或通信处理器、压缩引擎、图形处理器、GPGPU(通用图形处理单元)、高吞吐量的集成众核(MIC)协处理器(包括30个或更多核)、嵌入式处理器,等等。该处理器可以被实现在一个或多个芯片上。处理器1000可以是一个或多个基板的一部分,和/或可使用多种工艺技术(诸如例如,BiCMOS、CMOS、或NMOS)中的任何技术被实现在一个或多个基板上。

[0208] 存储器层次结构包括核内的一个或多个级别的高速缓存、一个或多个共享高速缓存单元的集合1006、以及耦合到集成存储器控制器单元的集合1014的外部存储器(未示出)。共享高速缓存单元的集合1006可包括一个或多个中间级别的高速缓存,诸如,第二级(L2)、第三级(L3)、第四级(L4)或其他级别的高速缓存、末级高速缓存(LLC)和/或以上各项的组合。虽然在一个实施例中,基于环的互连单元1012将集成图形逻辑1008(集成图形逻辑1008是专用逻辑的示例并且在本文中也称为专用逻辑)、共享高速缓存单元的集合1006以及系统代理单元1010/(多个)集成存储器控制器单元1014互连,但是替代实施例可使用任何数量的公知技术来互连此类单元。在一个实施例中,在一个或多个高速缓存单元1006与核1002A-N之间维持一致性。

[0209] 在一些实施例中,一个或多个核1002A-N能够实现多线程化。系统代理1010包括协调和操作核1002A-N的那些部件。系统代理单元1010可包括例如功率控制单元(PCU)和显示单元。PCU可以是对核1002A-N以及集成图形逻辑1008的功率状态进行调节所需的逻辑和部件,或可包括这些逻辑和部件。显示单元用于驱动一个或多个外部连接的显示器。

[0210] 核1002A-N在架构指令集方面可以是同构的或异构的;即,核1002A-N中的两个或更多个核可能能够执行相同的指令集,而其他核可能能够执行该指令集的仅仅子集或不同的指令集。

示例性计算机架构

[0211] 图11-14是示例性计算机架构的框图。本领域中已知的对膝上型设备、台式机、手持PC、个人数字助理、工程工作站、服务器、网络设备、网络集线器、交换机、嵌入式处理器、数字信号处理器(DSP)、图形设备、视频游戏设备、机顶盒、微控制器、蜂窝电话、便携式媒体播放器、手持设备以及各种其他电子设备的其他系统设计和配置也是合适的。一般地,能够包含如本文中所公开的处理器和/或其他执行逻辑的各种各样的系统或电子设备一般都是合适的。

[0212] 现在参考图11,所示出的是根据本发明一个实施例的系统1100的框图。系统1100可以包括一个或多个处理器1110、1115,这些处理器耦合到控制器中枢1120。在一个实施例中,控制器中枢1120包括图形存储器控制器中枢(GMCH)1190和输入/输出中枢(IOH)1150(其可以在分开的芯片上);GMCH 1190包括存储器和图形控制器,存储器1140和协处理器1145耦合到该存储器和图形控制器;IOH 1150将输入/输出(I/O)设备1160耦合到GMCH 1190。或者,存储器和图形控制器中的一个或这两者被集成在(如本文中所描述的)处理器内,存储器1140和协处理器1145直接耦合到处理器1110,并且控制器中枢1120与IOH 1150处于单个芯片中。

[0213] 附加的处理器1115的任选性在图11中通过虚线来表示。每一处理器1110、1115可包括本文中描述的处理核中的一个或多个,并且可以是处理器1000的某一版本。

[0214] 存储器1140可以是例如动态随机存取存储器 (DRAM)、相变存储器 (PCM) 或这两者的组合。对于至少一个实施例,控制器中枢1120经由诸如前端总线 (FSB) 之类的多分支总线、诸如快速路径互连 (QPI) 之类的点对点接口、或者类似的连接1195来与(多个)处理器1110、1115进行通信。

[0215] 在一个实施例中,协处理器1145是专用处理器,诸如例如,高吞吐量MIC处理器、网络或通信处理器、压缩引擎、图形处理器、GPGPU、嵌入式处理器,等等。在一个实施例中,控制器中枢1120可以包括集成图形加速器。

[0216] 在物理资源1110、1115之间可以存在包括架构、微架构、热、功耗特性等一系列品质度量方面的各种差异。

[0217] 在一个实施例中,处理器1110执行控制一般类型的数据处理操作的指令。嵌入在这些指令内的可以是协处理器指令。处理器1110将这些协处理器指令识别为具有应当由附连的协处理器1145执行的类型。因此,处理器1110在协处理器总线或者其他互连上将这些协处理器指令(或者表示协处理器指令的控制信号)发布到协处理器1145。(多个)协处理器1145接受并执行所接收的协处理器指令。

[0218] 现在参见图12,所示出的是根据本发明的实施例的第一更具体的示例性系统1200的框图。如图12中所示,多处理器系统1200是点对点互连系统,并且包括经由点对点互连1250耦合的第一处理器1270和第二处理器1280。处理器1270和1280中的每一个都可以是处理器1000的某一版本。在一些实施例中,处理器1270和1280分别是处理器1110和1115,而协处理器1238是协处理器1145。在另一实施例中,处理器1270和1280分别是处理器1110和协处理器1145。

[0219] 处理器1270和1280示出为分别包括集成存储器控制器 (IMC) 单元1272和1282。处理器1270还包括作为其总线控制器单元的一部分的点对点 (P-P) 接口1276和1278;类似地,第二处理器1280包括P-P接口1286和1288。处理器1270、1280可以经由使用点对点 (P-P) 接口电路1278、1288的P-P接口1250来交换信息。如图12中所示,IMC 1272和1282将处理器耦合到相应的存储器,即存储器1232和存储器1234,这些存储器可以是本地附连到相应处理器的主存储器的部分。

[0220] 处理器1270、1280可各自经由使用点对点接口电路1276、1294、1286、1298的各个P-P接口1252、1254来与芯片组1290交换信息。芯片组1290可以任选地经由高性能接口1292来与协处理器1238交换信息。在一个实施例中,协处理器1238是专用处理器,诸如例如,高吞吐量MIC处理器、网络或通信处理器、压缩引擎、图形处理器、GPGPU、嵌入式处理器,等等。

[0221] 共享高速缓存(未示出)可被包括在任一处理器中,或在这两个处理器的外部但经由P-P互连与这些处理器连接,使得如果处理器被置于低功率模式,则任一或这两个处理器的本地高速缓存信息可被存储在共享高速缓存中。

[0222] 芯片组1290可以经由接口1296耦合到第一总线1216。在一个实施例中,第一总线1216可以是外围部件互连 (PCI) 总线或诸如PCI快速总线或另一第三代I/O互连总线之类的总线,但是本发明的范围不限于此。

[0223] 如图12中所示,各种I/O设备1214可连同总线桥1218一起耦合到第一总线1216,该

总线桥1218将第一总线1216耦合到第二总线1220。在一个实施例中,诸如协处理器、高吞吐量MIC处理器、GPGPU、加速器(诸如例如,图形加速器或数字信号处理(DSP)单元)、现场可编程门阵列或任何其他处理器的一个或多个附加处理器1215耦合到第一总线1216。在一个实施例中,第二总线1220可以是低引脚数(LPC)总线。在一个实施例中,各种设备可耦合到第二总线1220,这些设备包括例如键盘和/或鼠标1222、通信设备1227以及存储单元1228,该存储单元1228诸如可包括指令/代码和数据1230的盘驱动器或者其他大容量存储设备。此外,音频I/O 1224可以被耦合到第二总线1220。注意,其他架构是可能的。例如,代替图12的点对点架构,系统可以实现多分支总线或其他此类架构。

[0224] 现在参考图13,示出的是根据本发明的实施例的第二更具体的示例性系统1300的框图。图12和13中的类似元件使用类似的附图标记,并且从图13中省略了图12的某些方面以避免混淆图13的其他方面。

[0225] 图13图示处理器1270、1280可分别包括集成存储器和I/O控制逻辑(“CL”)1372和1382。因此,CL 1372、1382包括集成存储器控制器单元,并包括I/O控制逻辑。图13图示不仅存储器1232、1234耦合到CL 1372、1382,而且I/O设备1314也耦合到控制逻辑1372、1382。传统I/O设备1315被耦合到芯片组1290。

[0226] 现在参考图14,示出的是根据本发明的实施例的SoC 1400的框图。图10中的类似要素使用类似的附图标记。另外,虚线框是更先进的SoC上的任选的特征。在图14中,(多个)互连单元1402被耦合到:应用处理器1410,其包括一个或多个核的集合1002A-N以及(多个)共享高速缓存单元1006,一个或多个核的集合1002A-N包括高速缓存单元1004A-N;系统代理单元1010;(多个)总线控制器单元1016;(多个)集成存储器控制器单元1014;一个或多个协处理器的集合1420,其可包括集成图形逻辑、图像处理器、音频处理器和视频处理器;静态随机存取存储器(SRAM)单元1430;直接存储器访问(DMA)单元1432;以及用于耦合到一个或多个外部显示器的显示单元1440。在一个实施例中,(多个)协处理器1420包括专用处理器,诸如例如,网络或通信处理器、压缩引擎、GPGPU、高吞吐量MIC处理器、或嵌入式处理器,等等。

[0227] 本文公开的机制的各实施例可以被实现在硬件、软件、固件或此类实现方式的组合中。本发明的实施例可实现为在可编程系统上执行的计算机程序或程序代码,该可编程系统包括至少一个处理器、存储系统(包括易失性和非易失性存储器和/或存储元件)、至少一个输入设备以及至少一个输出设备。

[0228] 可将程序代码(诸如,图12中图示的代码1230)应用于输入指令,以执行本文中描述的功能并生成输出信息。可以按已知方式将输出信息应用于一个或多个输出设备。为了本申请的目的,处理系统包括具有处理器的任何系统,该处理器诸如例如,数字信号处理器(DSP)、微控制器、专用集成电路(ASIC)或微处理器。

[0229] 程序代码可以用高级的面向过程的编程语言或面向对象的编程语言来实现,以便与处理系统通信。如果需要,也可用汇编语言或机器语言来实现程序代码。事实上,本文中描述的机制不限于任何特定的编程语言的范围。在任何情况下,该语言可以是编译语言或解释语言。

[0230] 至少一个实施例的一个或多个方面可以由存储在机器可读介质上的表示性指令来实现,该指令表示处理器中的各种逻辑,该指令在被机器读取时使得该机器制造用于执

行本文中所述的技术的逻辑。被称为“IP核”的此类表示可以被存储在有形的机器可读介质上,并可被供应给各个客户或生产设施以加载到实际制造该逻辑或处理器的制造机器中。

[0231] 此类机器可读存储介质可以包括但不限于通过机器或设备制造或形成的制品的非暂态、有形布置,其包括存储介质,诸如硬盘;任何其他类型的盘,包括软盘、光盘、紧致盘只读存储器(CD-ROM)、可重写紧致盘(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)和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);相变存储器(PCM);磁卡或光卡;或适于存储电子指令的任何其他类型的介质。

[0232] 因此,本发明的实施例还包括非暂态的有形机器可读介质,该介质包含指令或包含设计数据,诸如硬件描述语言(HDL),它定义本文中描述的结构、电路、装置、处理器和/或系统特征。这些实施例也被称为程序产品。

仿真(包括二进制变换、代码变形等)

[0233] 在一些情况下,指令转换器可用于将指令从源指令集转换至目标指令集。例如,指令转换器可以将指令变换(例如,使用静态二进制变换、包括动态编译的动态二进制变换)、变形、仿真或以其他方式转换成要由核处理的一条或多条其他指令。指令转换器可以用软件、硬件、固件、或其组合来实现。指令转换器可以在处理器上、在处理器外、或者部分在处理器上且部分在处理器外。

[0234] 图15是根据本发明的一些实施例的对照使用软件指令转换器将源指令集中的二进制指令转换成目标指令集中的二进制指令的框图。在所图示的实施例中,指令转换器是软件指令转换器,但替代地,该指令转换器可以用软件、固件、硬件或其各种组合来实现。图15示出可使用x86编译器1504来编译高级语言1502形式的程序,以生成可由具有至少一个x86指令集核的处理器1516原生执行的x86二进制代码1506。具有至少一个x86指令集核的处理器1516表示通过兼容地执行或以其他方式处理以下各项来执行与具有至少一个x86指令集核的英特尔处理器基本相同的功能的任何处理器:1) 英特尔x86指令集核的指令集的实质部分,或2) 目标为在具有至少一个x86指令集核的英特尔处理器上运行以便取得与具有至少一个x86指令集核的英特尔处理器基本相同的结果的应用或其他软件的目标代码版本。x86编译器1504表示可操作用于生成x86二进制代码1506(例如,目标代码)的编译器,该二进制代码可通过或不通过附加的链接处理在具有至少一个x86指令集核的处理器1516上执行。类似地,图15示出可以使用替代的指令集编译器1508来编译高级语言1502形式的程序,以生成可以由不具有至少一个x86指令集核的处理器1514(例如,具有执行加利福尼亚州桑尼维尔市的MIPS技术公司的MIPS指令集、和/或执行加利福尼亚州桑尼维尔市的ARM控股公司的ARM指令集的核的处理器)原生执行的替代的指令集二进制代码1510。指令转换器1512用于将x86二进制代码1506转换成可以由不具有x86指令集核的处理器1514原生执行的代码。该转换后的代码不大可能与替代的指令集二进制代码1510相同,因为能够这样做的指令转换器难以制造;然而,转换后的代码将完成一般操作,并且由来自替代指令集的指令构成。因此,指令转换器1512通过仿真、模拟或任何其他过程来表示允许不具有x86指令集处理器或核的处理器或其他电子设备执行x86二进制代码1506的软件、固件、硬件或其组合。

进一步的示例

[0235] 示例1提供一种处理器,包括:寄存器堆,包括多个向量寄存器;存储器接口,用于从存储器接收读取数据;取出电路,用于取出指令;解码电路,用于对所取出的指令解码,所取出的指令具有用于指定操作码、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数,操作码用于指示处理器用于:取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将写入数据写入目的地向量寄存器;以及执行电路,用于根据操作码执行经解码的指令,其中执行电路具有放置在存储器与寄存器堆之间的混洗流水线,混洗流水线用于取出、解码、以及以一个指令每时钟周期的速率执行指令的进一步的实例。

[0236] 示例2包括示例1的示例性处理器的实质内容,其中指令进一步指定源向量的宽度,宽度为32位、64位、128位、256位、512位、和1024位中的一个。

[0237] 示例3包括示例1的示例性处理器的实质内容,其中指令进一步指定源向量的每个元素的尺寸,尺寸为2位、4位、8位、16位、32位、64位、和128位中的一个。

[0238] 示例4包括示例1的示例性处理器的实质内容,其中源向量的N组元素中的每一组包括M个元素,其中目的地向量寄存器足够宽以存储N x M个元素,并且其中操作码进一步指示处理器用于:生成M个元组并且将M个元组写入目的地向量寄存器的对应元素,M个元组中的每一个用于包括N组元素的N个对应元素。

[0239] 示例5包括示例1的示例性处理器的实质内容,其中源向量的N组元素中的每一组包括M个元素,其中指令指定M个目的地向量寄存器,并且其中操作码进一步指示处理器用于:生成M个N元组并且将所生成的每个N元组广播到M个目的地向量寄存器中的一个的整个宽度。

[0240] 示例6包括示例1的示例性处理器的实质内容,其中源向量的N组元素被连续地布置在存储器中。

[0241] 示例7包括示例1的示例性处理器的实质内容,其中指令进一步指定一个或多个附加存储器位置,使得为N组元素中的每一组提供存储器位置。

[0242] 示例8包括示例1的示例性处理器的实质内容,其中源向量的N组元素中的每一组包括M个元素,并且其中操作码进一步指示处理器用于生成与源向量的M个元素中的两个对应的两个N元组并且将两个N元组广播到目的地向量寄存器的整个宽度。

[0243] 示例9包括示例1的示例性处理器的实质内容,其中操作码进一步用于指示:写入数据用于包括基于N个源向量的所选择的元素生成的单个元组,单个元组将被广播到目的地向量寄存器的整个宽度。

[0244] 示例10包括示例9的示例性处理器的实质内容,其中将被广播的单个元组将利用附加指令字段被选择。

[0245] 示例11提供一种用于由处理器执行的方法,处理器包括用于从存储器读取数据的存储器接口以及包括多个向量寄存器的寄存器堆,该方法包括:使用取出电路来取出指令;使用解码电路对所取出的指令解码,所取出的指令具有用于指定操作码、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数,操作码用于指示处理器用于:取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将写入数据写入目的地向量寄存器;以及使用执行电路根据操作码执行经解码的指令,其中执行电路具有放置在存储器与寄存器堆之间的混洗流水线,混

洗流水线用于取出、解码、以及以一个指令每时钟周期的速率执行指令的进一步的实例。

[0246] 示例12包括示例11的示例性方法的实质内容,其中指令进一步指定源向量的宽度,宽度为32位、64位、128位、256位、512位、和1024位中的一个。

[0247] 示例13包括示例11的示例性方法的实质内容,其中指令进一步指定源向量的每个元素的尺寸,尺寸为2位、4位、8位、16位、32位、64位、和128位中的一个。

[0248] 示例14包括示例11的示例性方法的实质内容,其中源向量的N组元素中的每一组包括M个元素,其中目的地向量寄存器足够宽以存储N x M个元素,并且其中操作码进一步指示处理器用于:生成M个元组并且将M个元组写入目的地向量寄存器的对应元素,M个元组中的每一个用于包括N组元素的N个对应元素。

[0249] 示例15包括示例11的示例性方法的实质内容,其中源向量的N组元素中的每一组包括M个元素,其中指令指定M个目的地向量寄存器,并且其中操作码进一步指示处理器用于:生成M个N元组并且将所生成的每个N元组广播到M个目的地向量寄存器中的一个的整个宽度。

[0250] 示例16包括示例11的示例性方法的实质内容,其中源向量的N组元素被连续地布置在存储器中。

[0251] 示例17包括示例11的示例性方法的实质内容,其中指令进一步指定一个或多个附加存储器位置,使得为N组元素中的每一组提供存储器位置。

[0252] 示例18包括示例11的示例性方法的实质内容,其中源向量的N组元素中的每一组包括M个元素,并且其中操作码进一步指示处理器用于生成与M个元素中的两个对应的两个N元组并且将两个N元组广播到目的地向量寄存器的整个宽度。

[0253] 示例19包括示例11的示例性方法的实质内容,其中操作码进一步用于指示:写入数据用于包括基于N个源向量的所选择的元素生成的单个元组,单个元组将被广播到目的地向量寄存器的整个宽度。

[0254] 示例20包括示例19的示例性方法的实质内容,其中将被广播的单个元组将利用附加指令字段被选择。

[0255] 示例21提供一种包含指令的非暂态机器可读介质,包括用于从存储器读取数据的存储器接口以及包括多个向量寄存器的寄存器堆的处理器用于通过以下操作对指令作出响应:使用取出电路来取出指令;使用解码电路对所取出的指令解码,所取出的指令具有用于指定操作码、目的地向量寄存器、以及包括N组元素的源向量的存储器位置的字段,N为正整数,操作码用于指示处理器用于:取出源向量,生成包括一个或多个N元组的写入数据,每个N元组包括来自N组元素中的每一组的对应元素,并且将写入数据写入目的地向量寄存器;以及使用执行电路根据操作码执行经解码的指令,其中执行电路包括放置在存储器与寄存器堆之间的混洗流水线,混洗流水线用于取出、解码、以及以一个指令每时钟周期的速率执行指令的进一步的实例。

[0256] 示例22包括示例21的示例性非暂态机器可读介质的实质内容,其中指令进一步指定源向量的宽度,宽度为32位、64位、128位、256位、512位、和1024位中的一个。

[0257] 示例23包括示例21的示例性非暂态机器可读介质的实质内容,其中指令进一步指定源向量的每个元素的尺寸,元素尺寸为2位、4位、8位、16位、32位、64位、和128位中的一个。

[0258] 示例24包括示例21的示例性非暂态机器可读介质的实质内容,其中源向量的N组元素中的每一组包括M个元素,其中目的地向量寄存器足够宽以存储 $N \times M$ 个元素,并且其中操作码进一步指示处理器用于:生成M个元组并且将M个元组写入目的地向量寄存器的对应元素,M个元组中的每一个用于包括N组元素的N个对应元素。

[0259] 示例25包括示例21的示例性非暂态机器可读介质的实质内容,其中源向量的N组元素中的每一组包括M个元素,其中指令指定M个目的地向量寄存器,并且其中操作码进一步指示处理器用于:生成M个N元组并且将所生成的每个N元组广播到M个目的地向量寄存器中的一个的整个宽度。

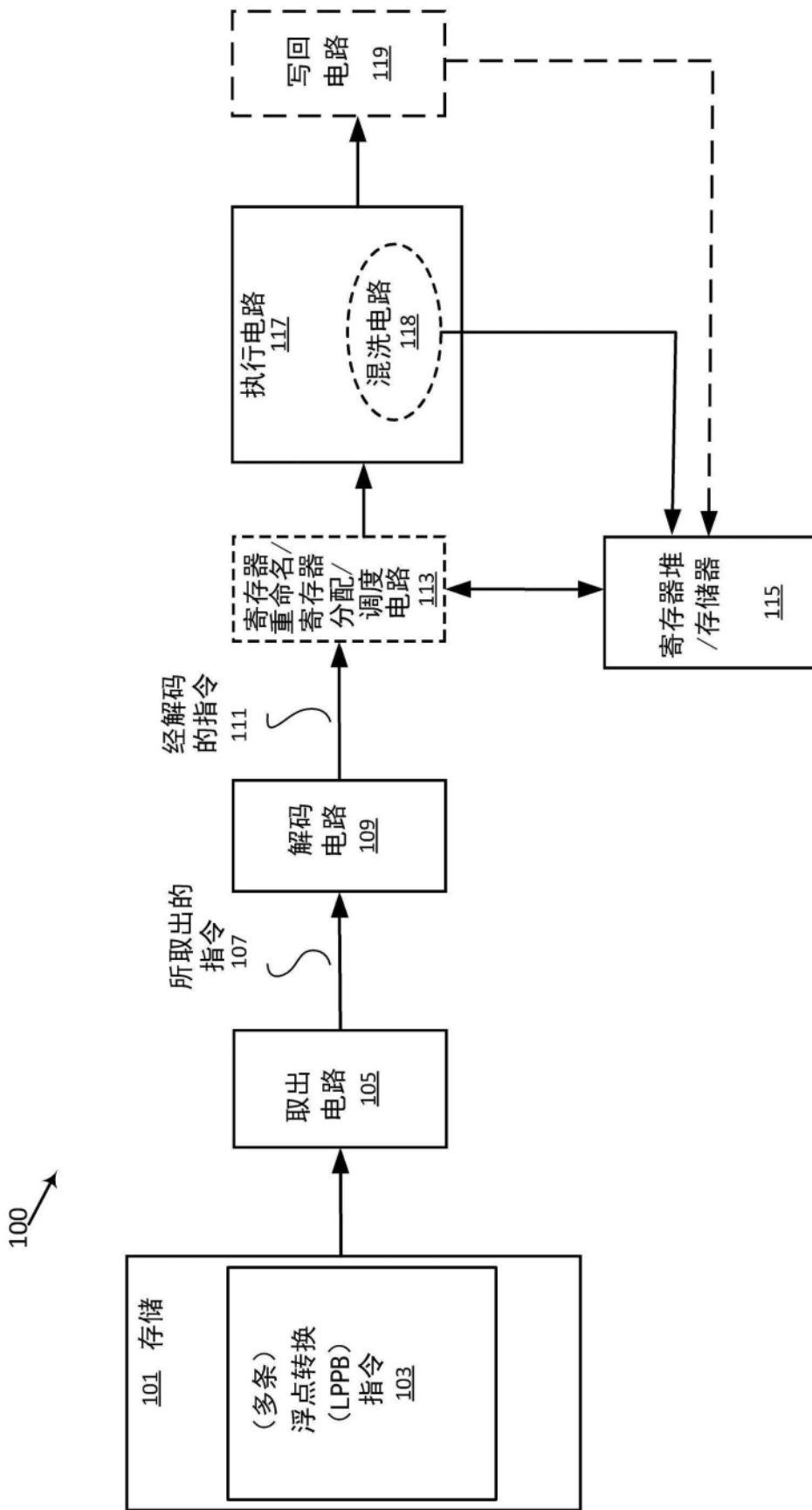


图1A

可能具有广播的加载置换 (LPPB) 指令 150A

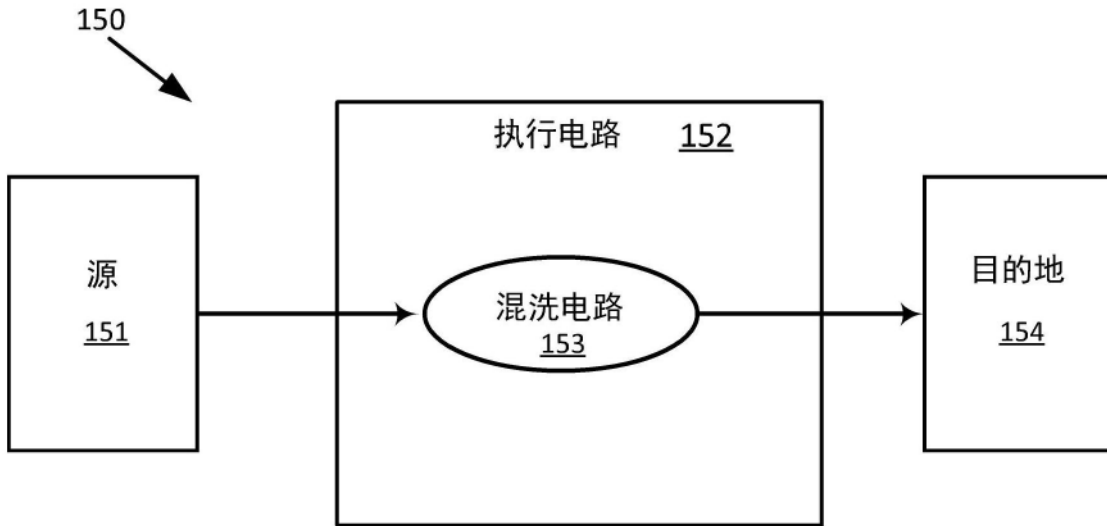


图1B

混洗流水线 201

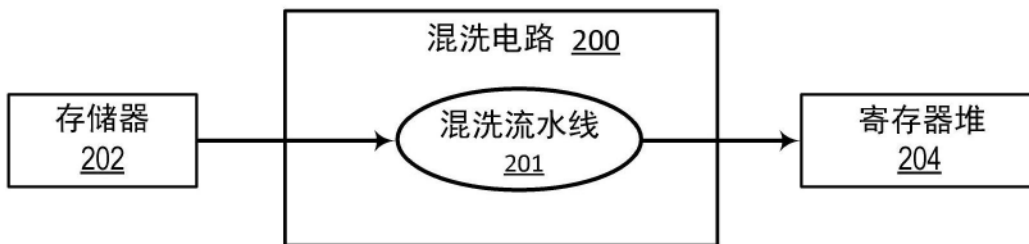


图2A

可能具有广播的加载置换 (LPPB) 指令

205A

<p>操作码 LPPBWN2M16W512</p>	<p>目的地 位置 (寄存器)</p>	<p>源位置 (存储器)</p>
-------------------------------	-----------------------------	----------------------

205

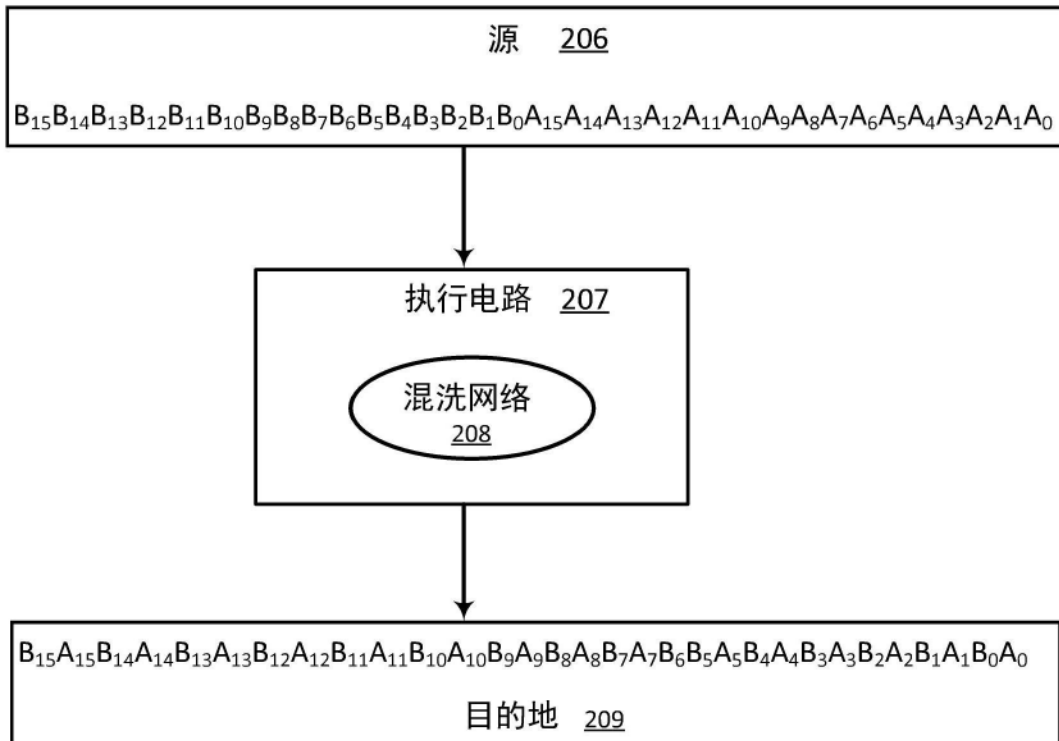


图2B

可能具有广播的加载置换 (LPPB) 指令 210A

<p>操作码 LPPBBN4M8W256</p>	<p>目的地 位置 (寄存器)</p>	<p>源位置 (存储器)</p>
------------------------------	-----------------------------	----------------------

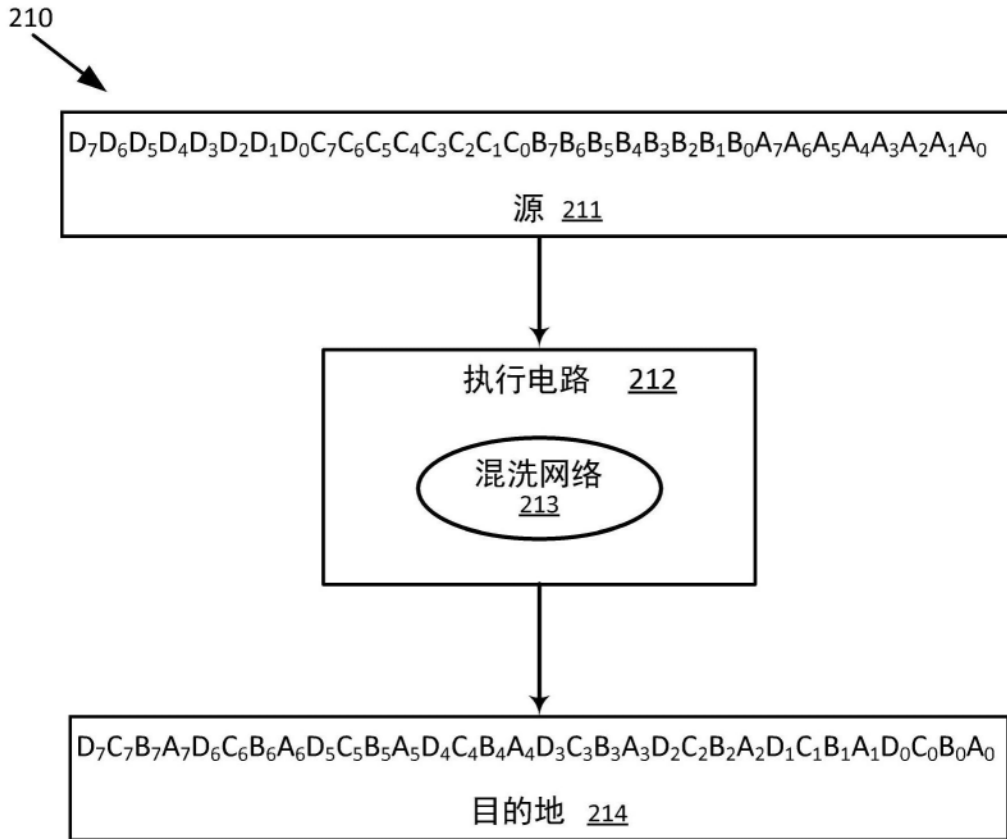


图2C

可能具有广播的加载置换 (LPPB) 指令 215A



215

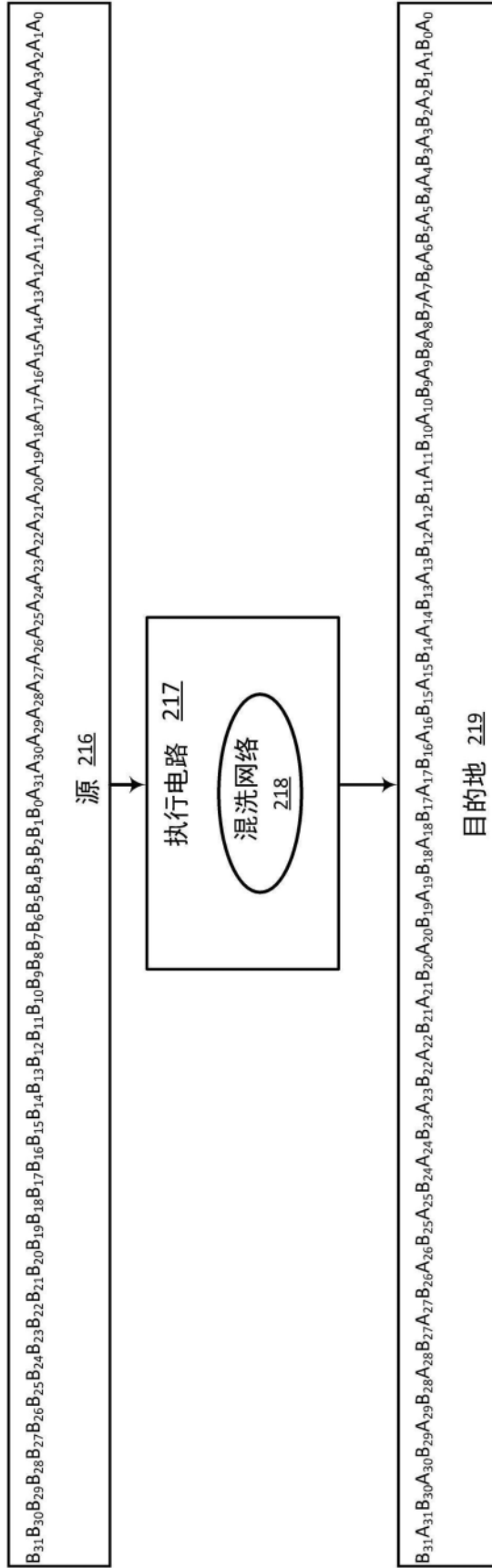


图2D

可能具有广播的加载置换 (LPPB) 指令 225A

操作码 LPPBDualCastWN2M16W512	目的地位置 (寄存器)	源位置 (存储器)	要广播的 索引=7
-------------------------------	----------------	--------------	--------------

225
↓

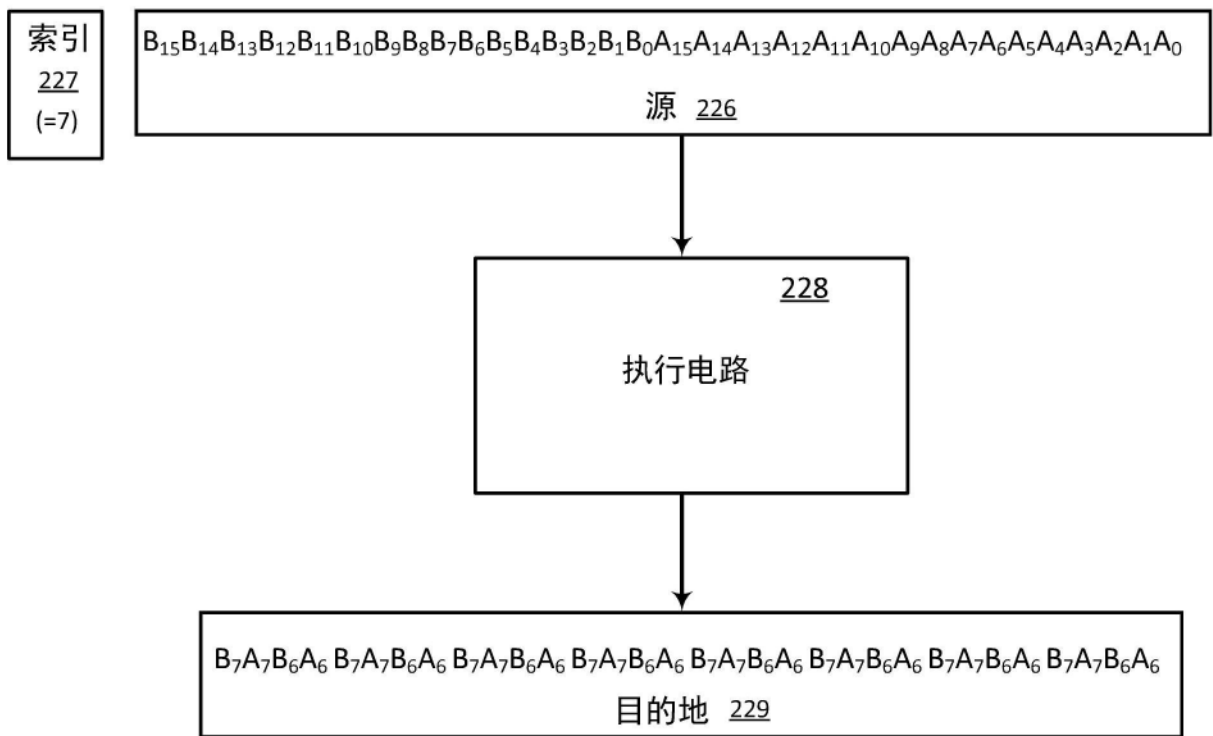


图2F

可能具有广播的加载置换 (LPPB) 指令 235A

操作码 LPPBCastWAIIN2M16W512	目的地位置 (寄存器)	源位置 (存储器)
------------------------------	----------------	--------------

235

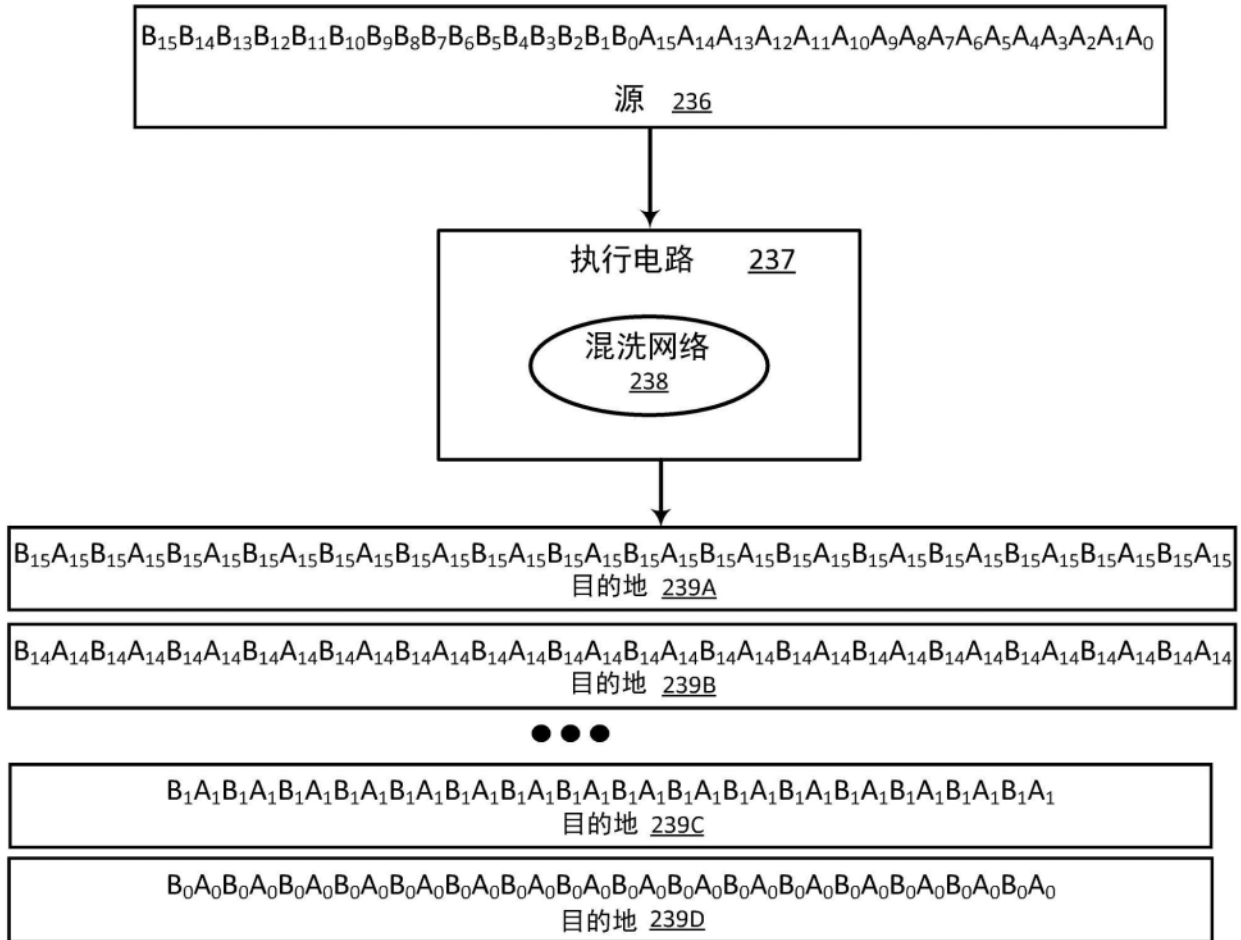


图2H

可能具有广播的加载置换 (LPPB) 指令 240A

操作码 LPPBCastW5N4M8W512	目的地 位置 (寄存器)	源位置 (存储器)
---------------------------	--------------------	--------------

240
↓

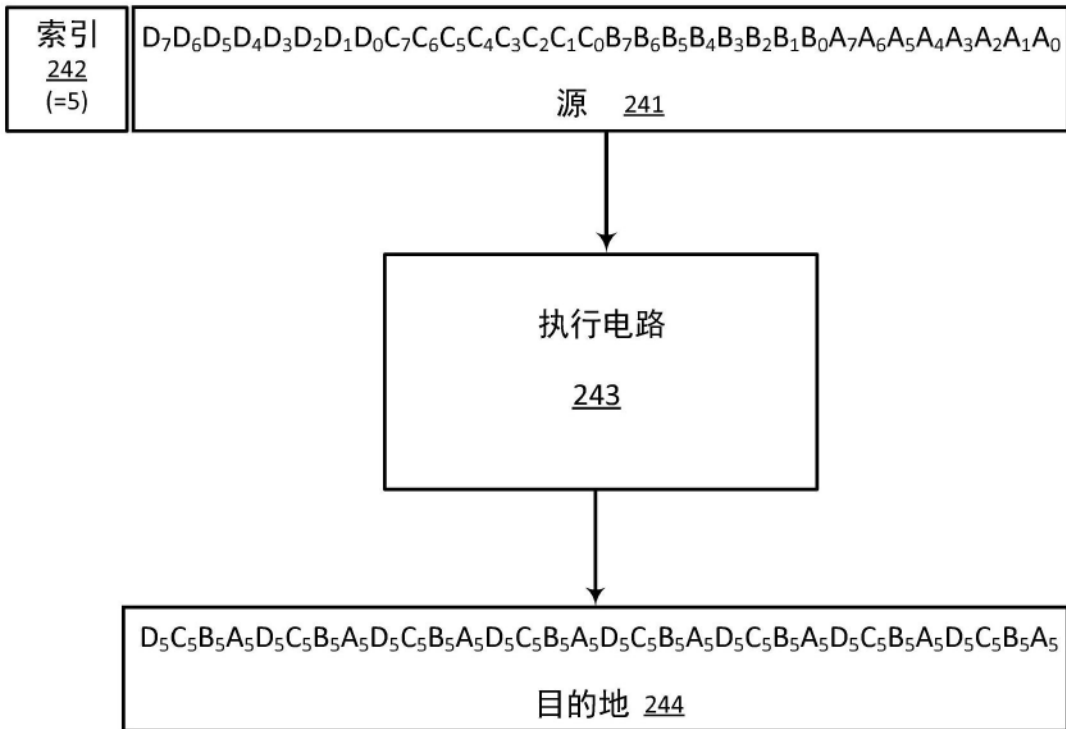


图2I

可能具有广播的加载置换 (LPPB) 指令 245A

操作码 LPPBCastWN4M8W512	目的地位置 (寄存器)	源位置 (存储器)	要广播的 索引=5
--------------------------	----------------	--------------	--------------

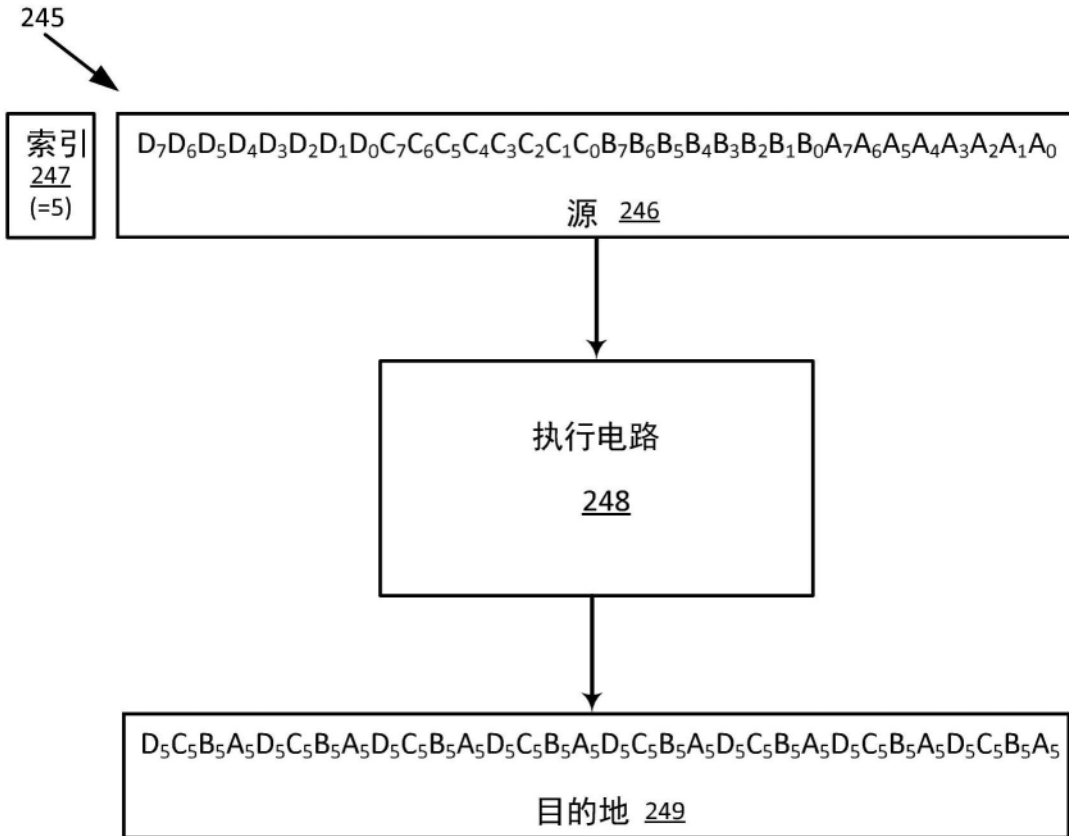


图2J

可能具有广播的加载置换 (LPPB) 指令 310

操作码 LPPB	目的地 位置 (寄存器)	源位置 (存储器)
-------------	--------------------	--------------

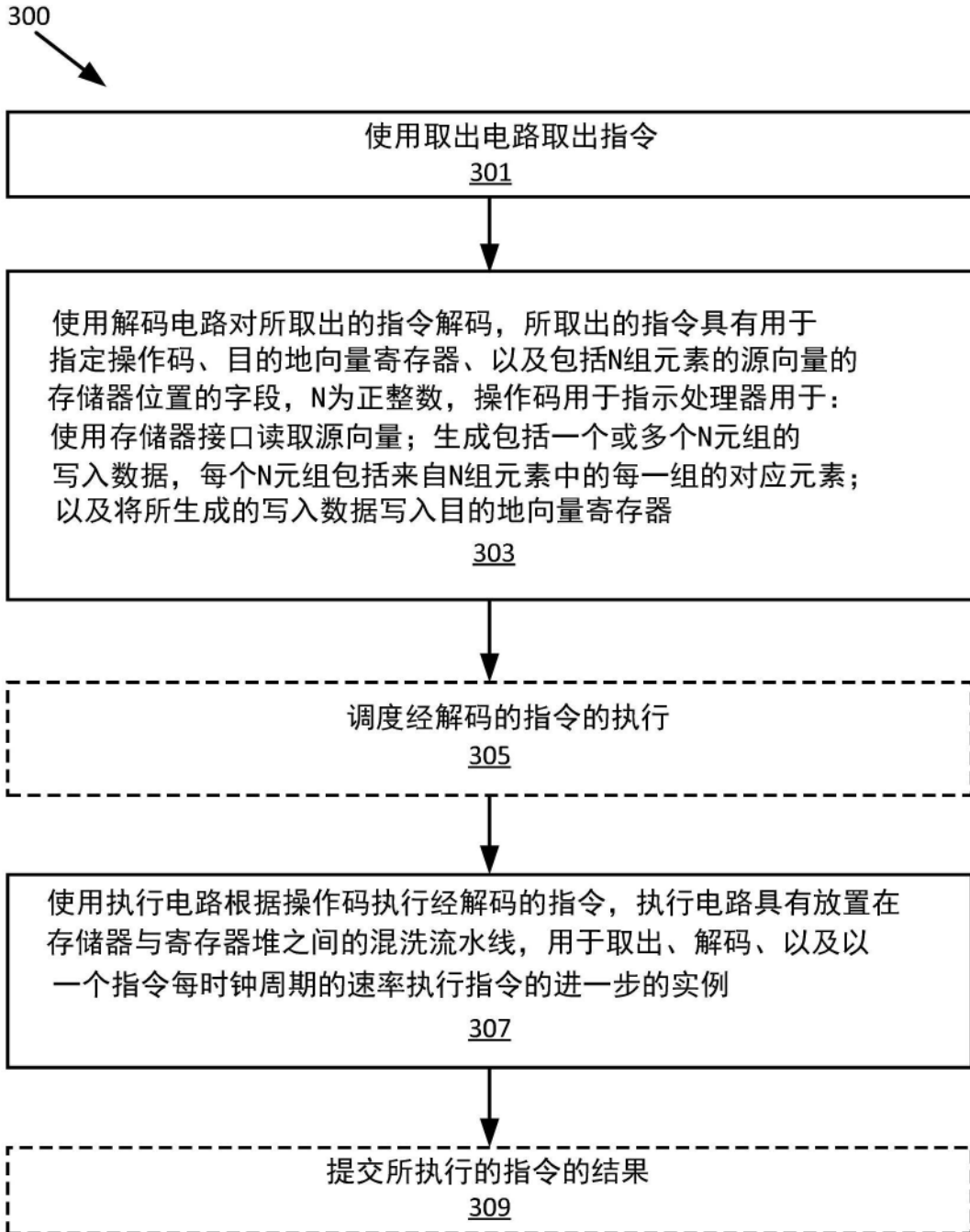


图3

可能具有广播的加载置换 (LPPB) 指令 400

操作码 LPPB* <u>402</u>	目的地位置 (寄存器) <u>404</u>	源位置 (存储器) <u>406</u>	源向量 宽度 <u>408</u>	源元素 尺寸 <u>410</u>	N (元组) <u>412</u>	M (每组的 数量) <u>414</u>	{k} <u>416</u>	{Z} <u>418</u>
----------------------------	------------------------------	----------------------------	-------------------------	-------------------------	-------------------------	--------------------------------	-------------------	-------------------

图4A

可能具有广播的加载置换 (LPPB) 指令 450

操作码 LPPBCast* <u>452</u>	目的地位置 (寄存器) <u>454</u>	源位置 (存储器) <u>456</u>	要投射 的索引 457	源向量 宽度 <u>458</u>	源元素 尺寸 <u>460</u>	N (元组) <u>462</u>	M (每组的 数量) <u>464</u>	{k} <u>466</u>	{Z} <u>468</u>
--------------------------------	------------------------------	----------------------------	-------------------	-------------------------	-------------------------	-------------------------	--------------------------------	-------------------	-------------------

图4B

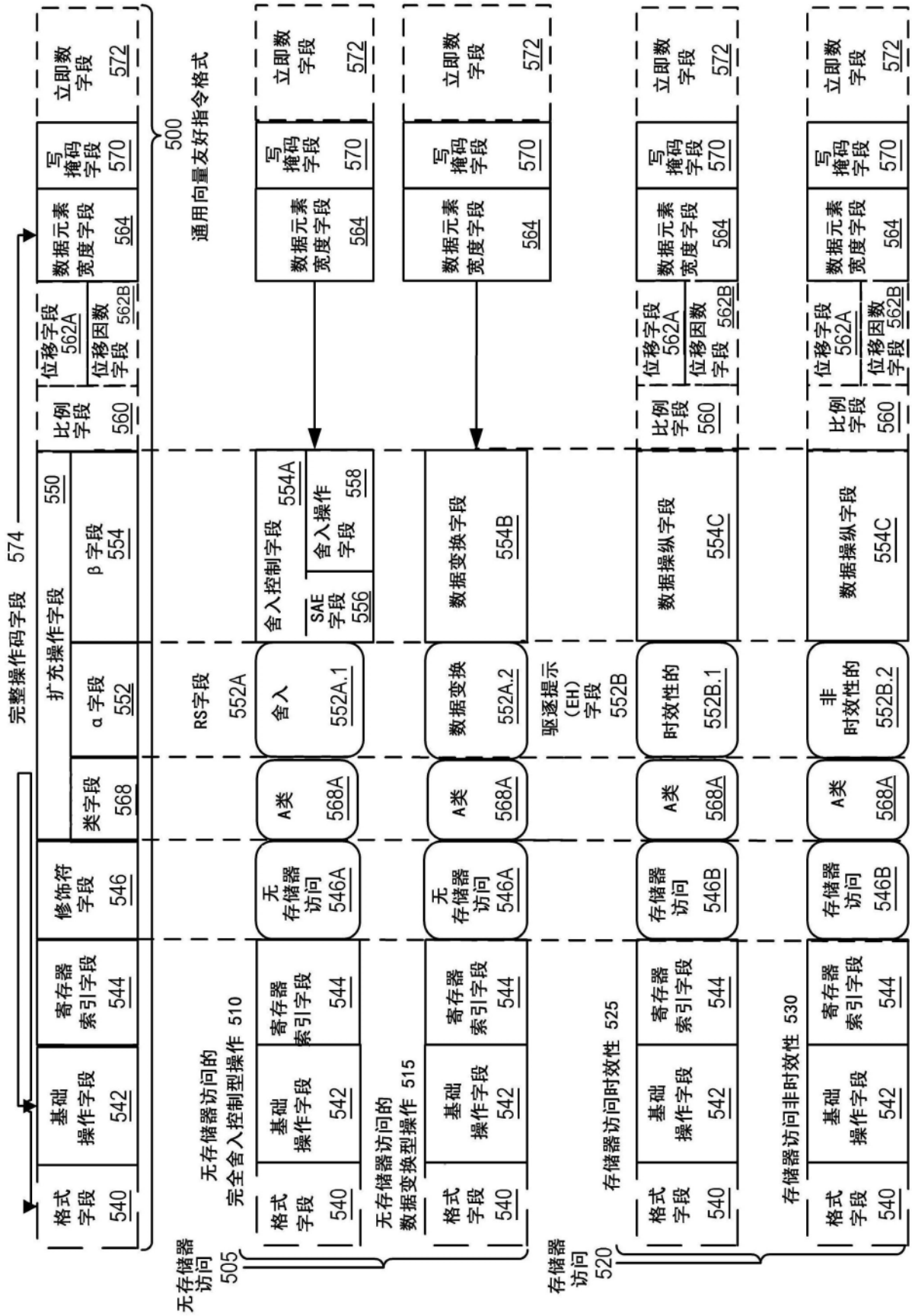


图5A

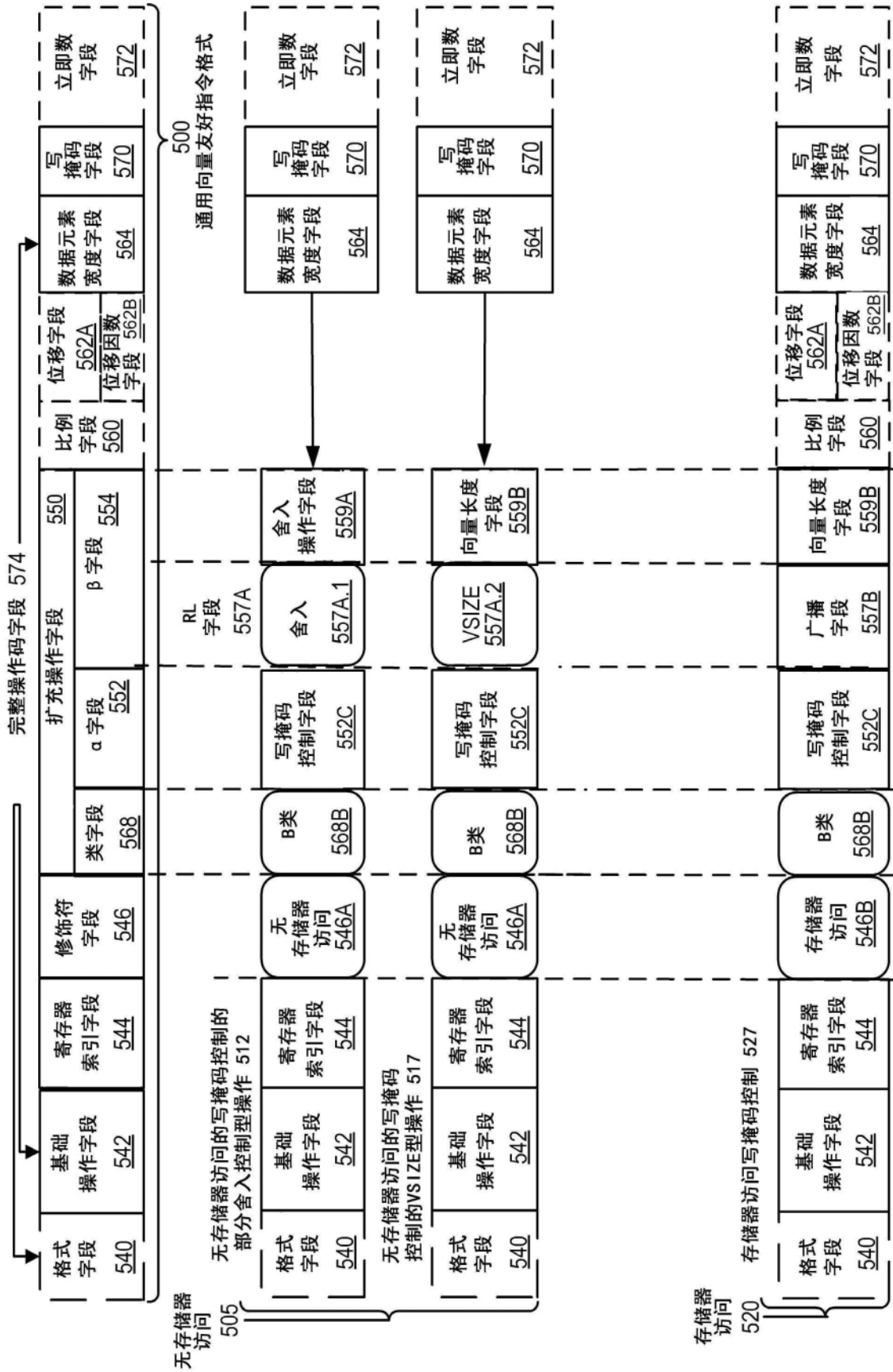


图5B

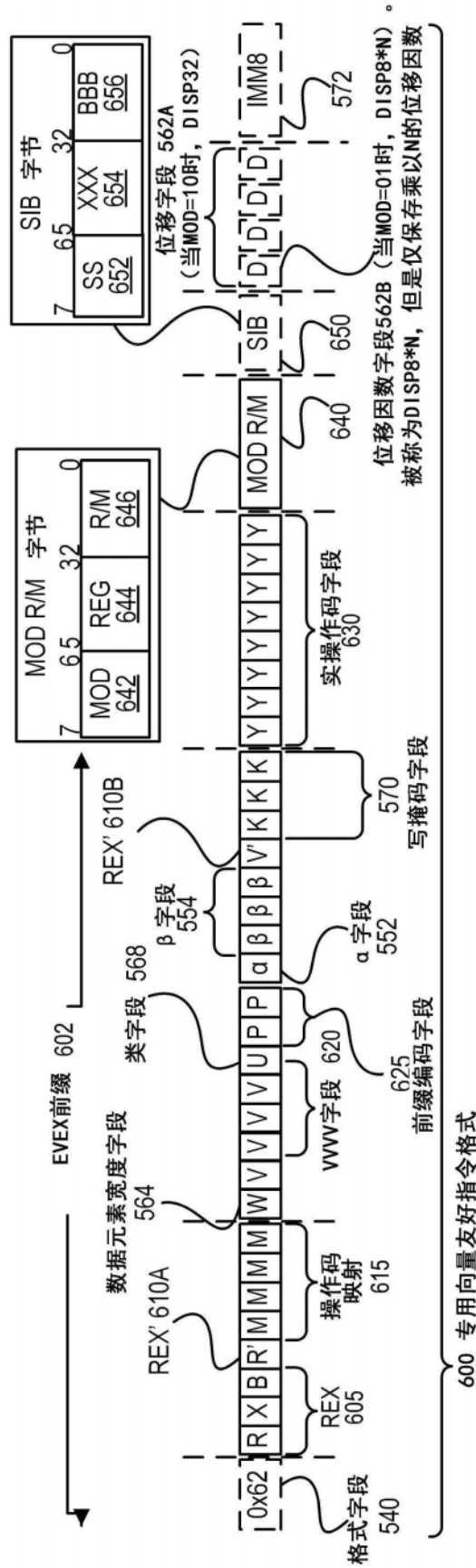


图6A

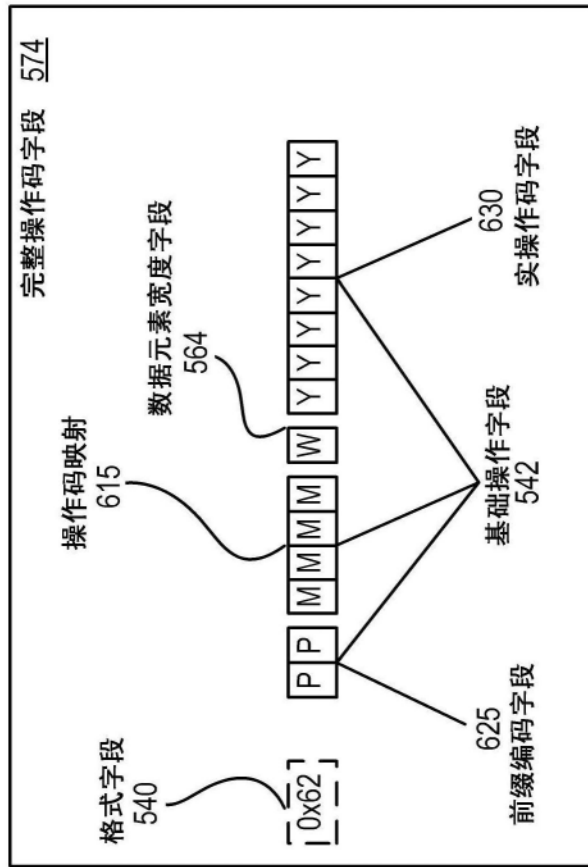


图6B

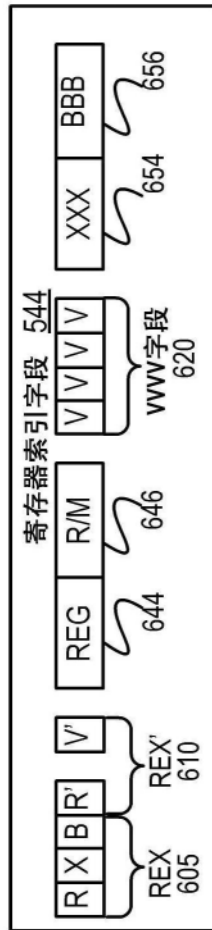


图6C

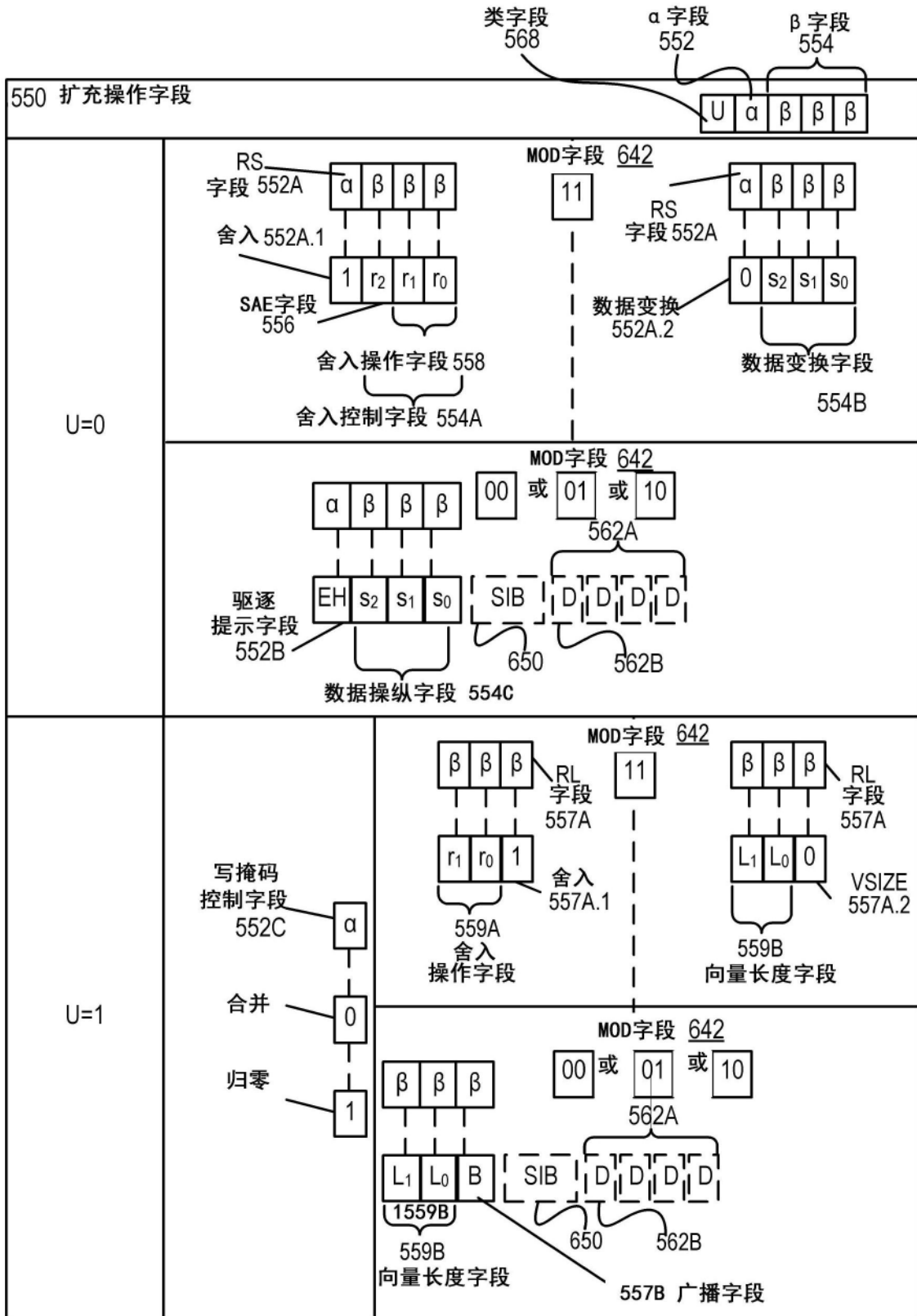


图6D

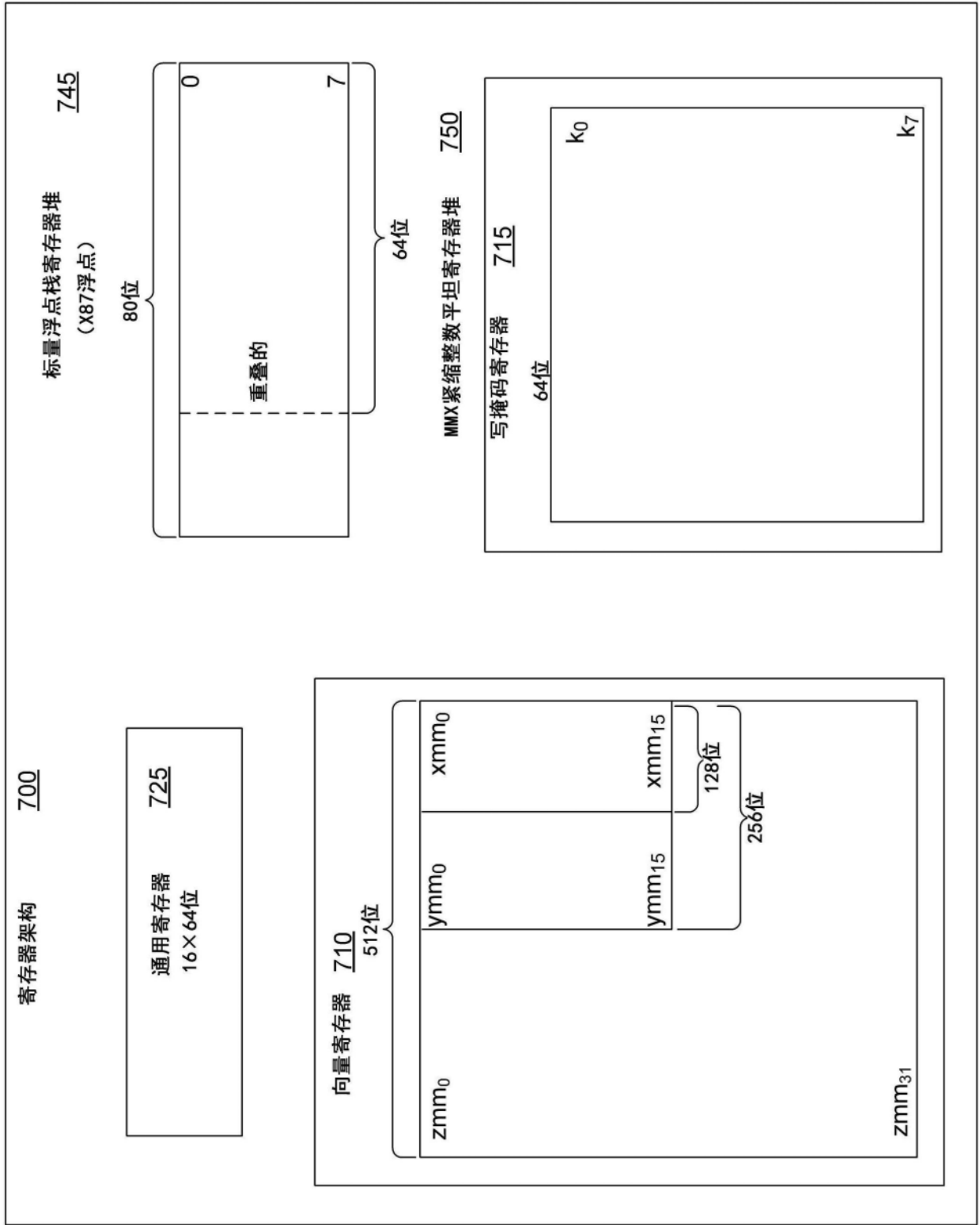


图7

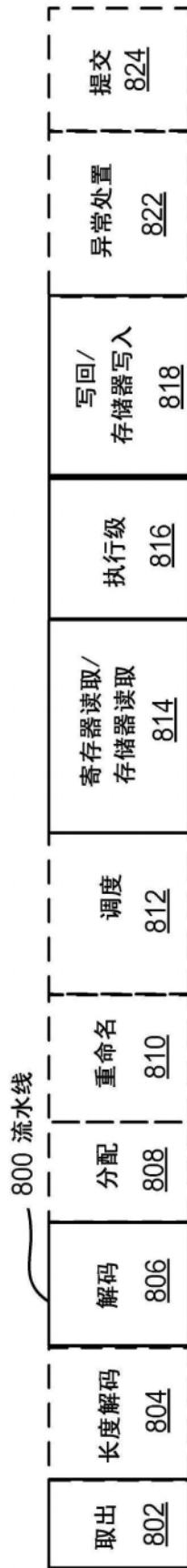


图8A

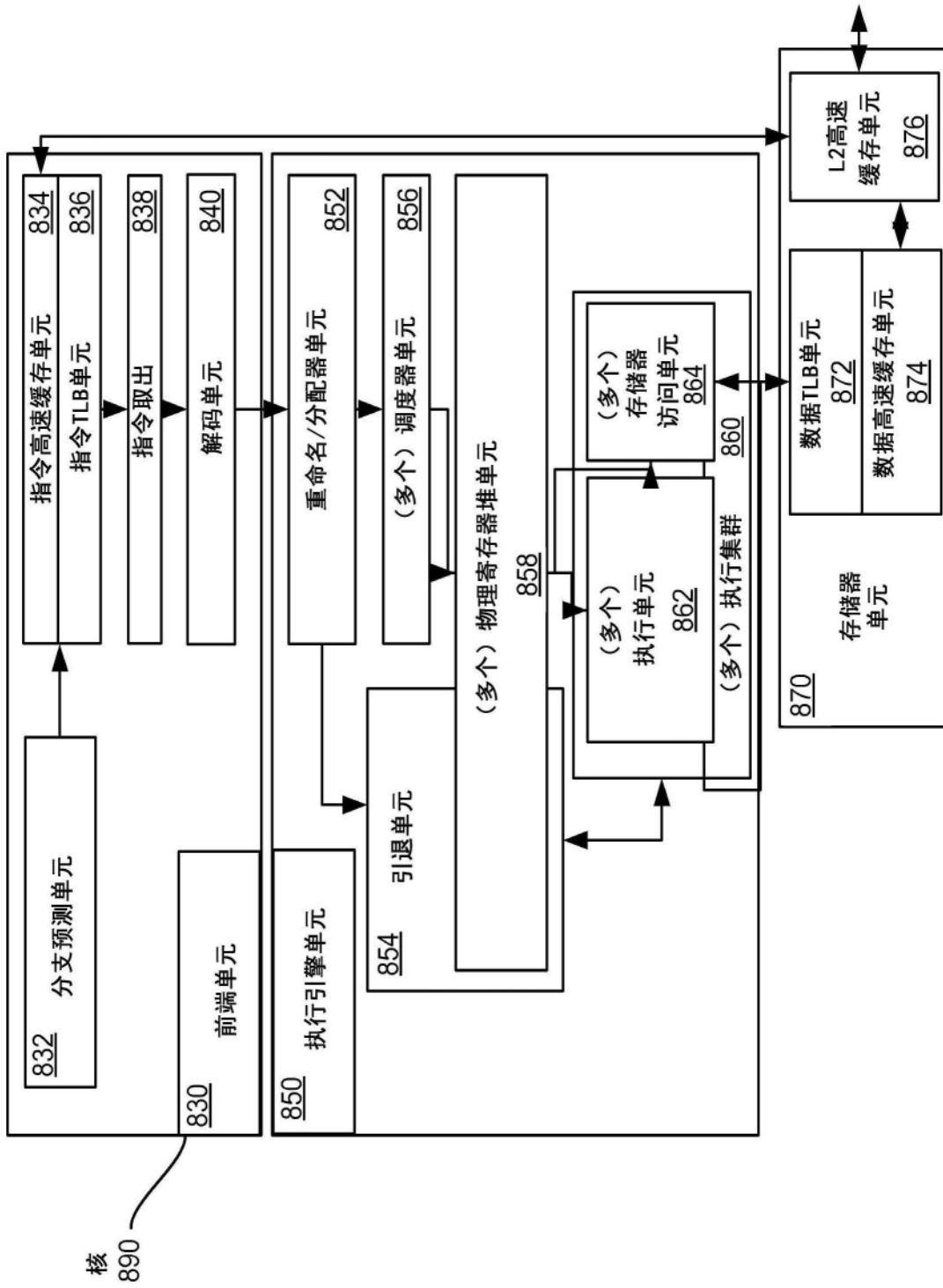


图8B

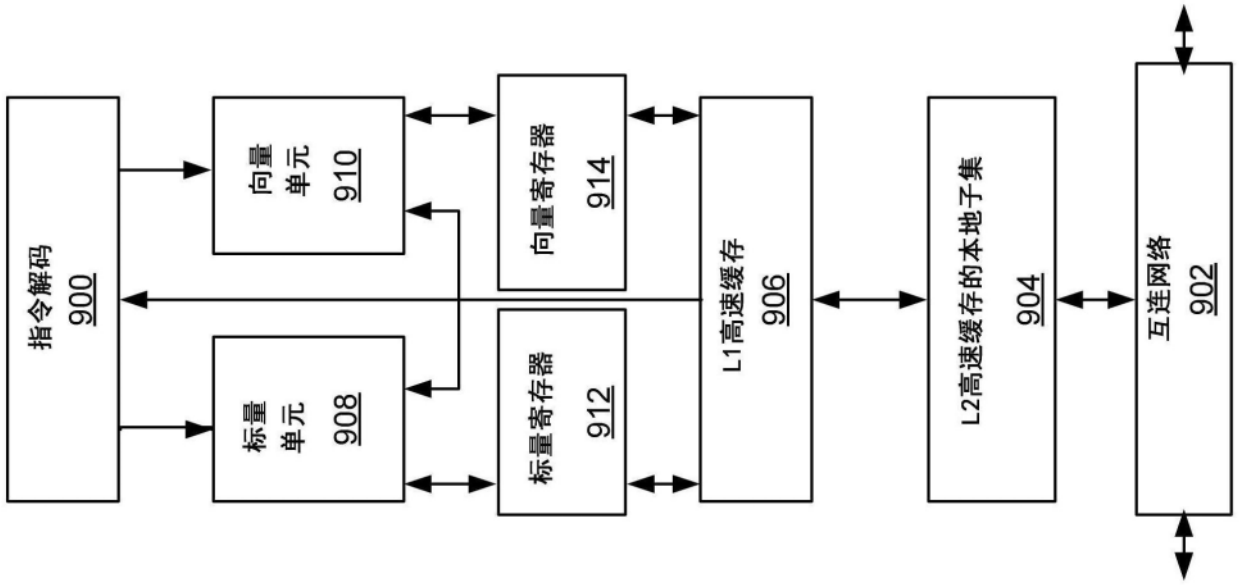


图9A

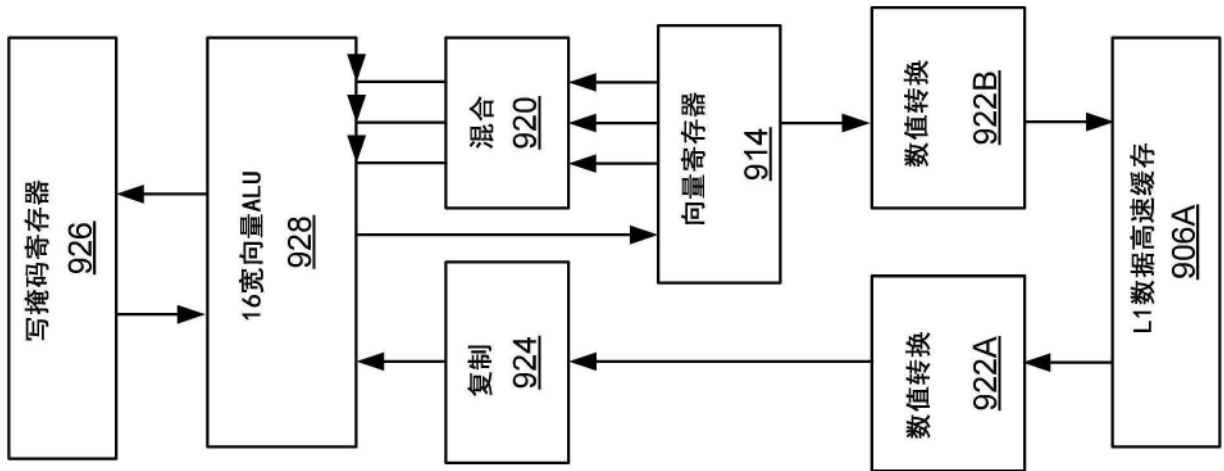


图9B

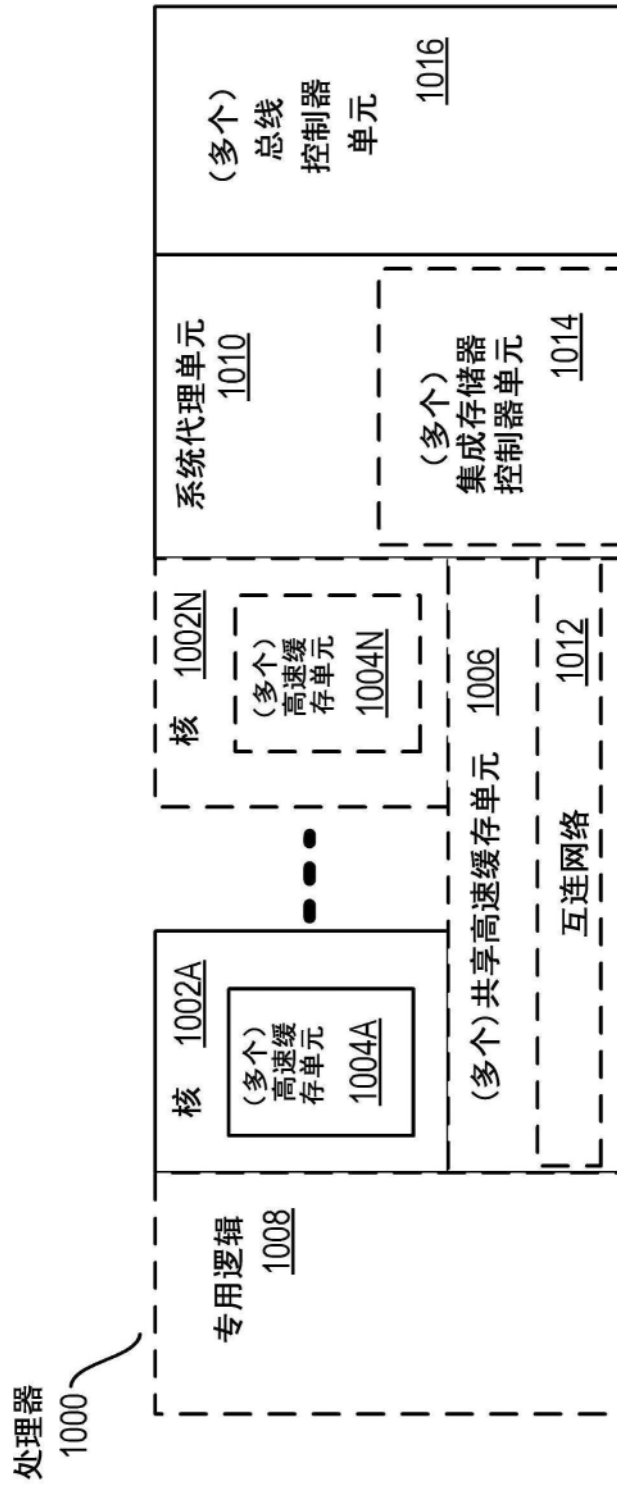


图10

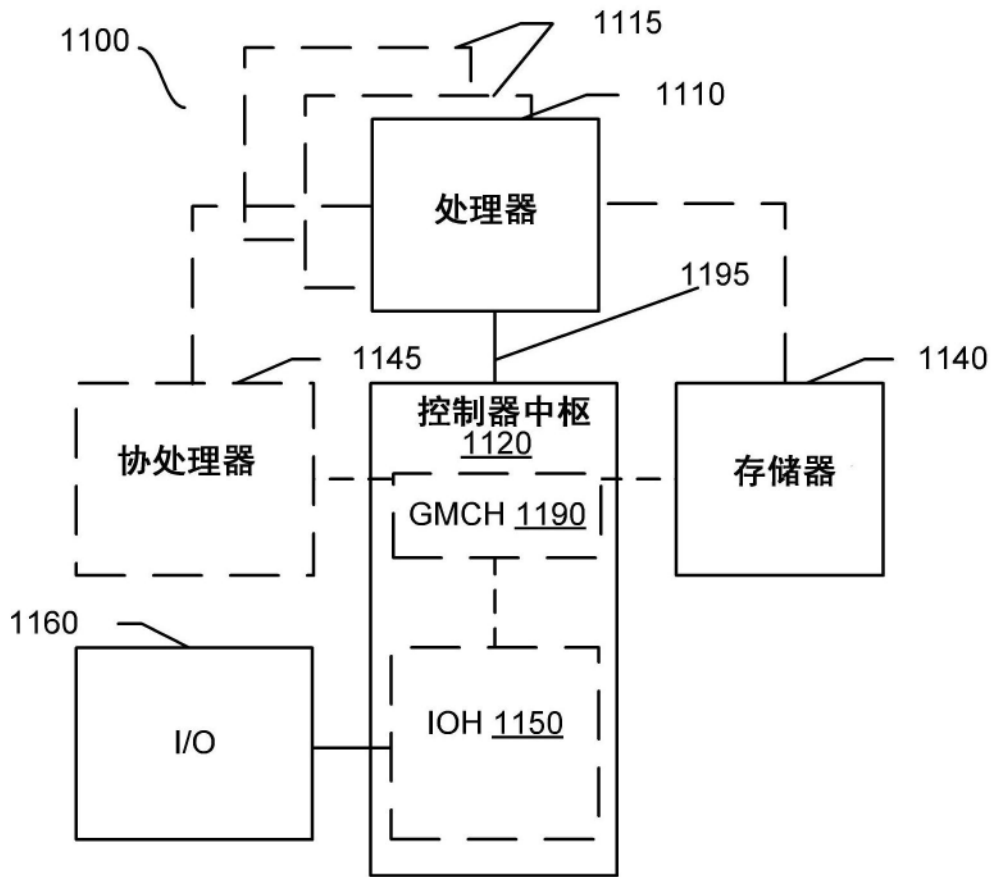


图11

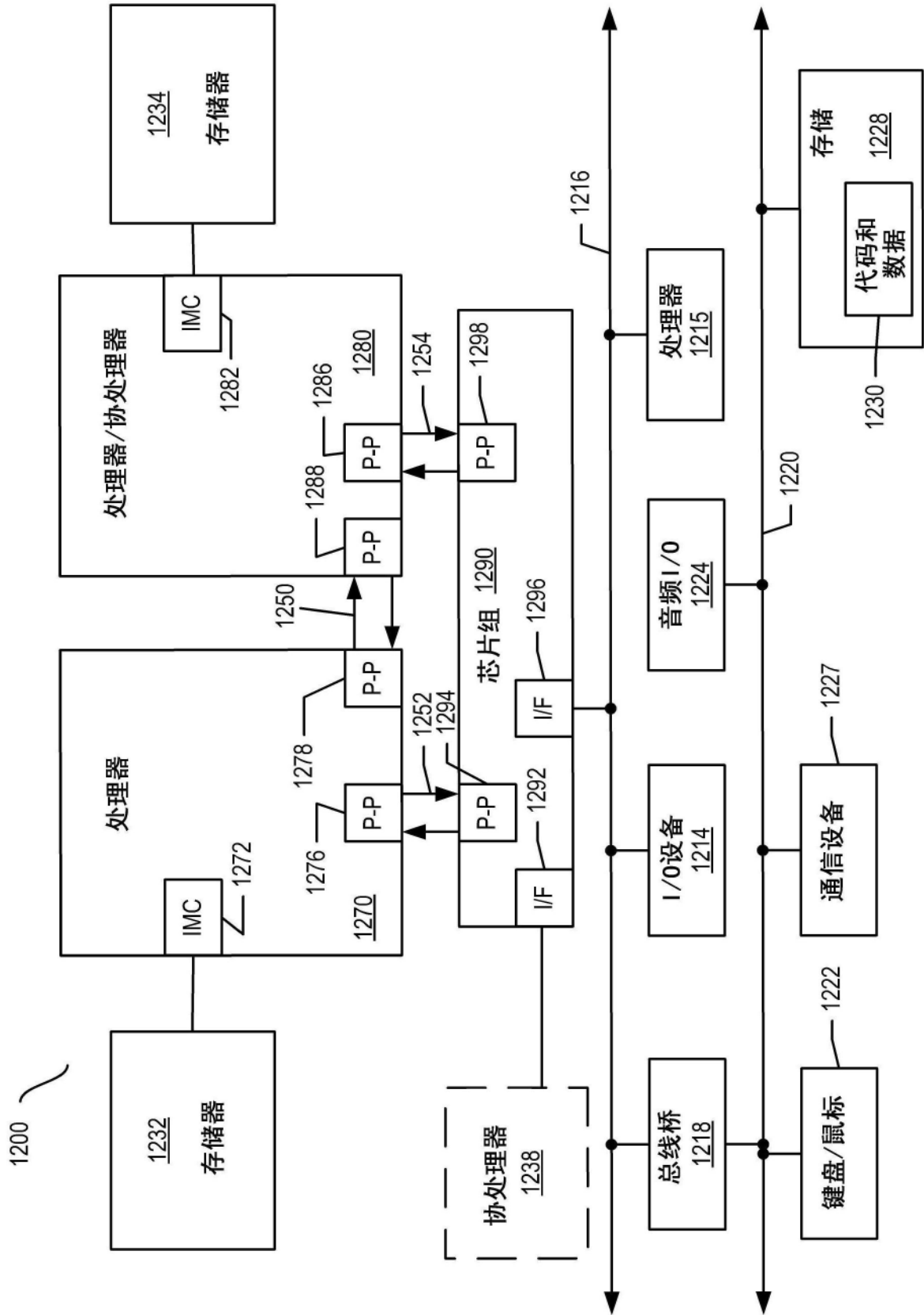


图12

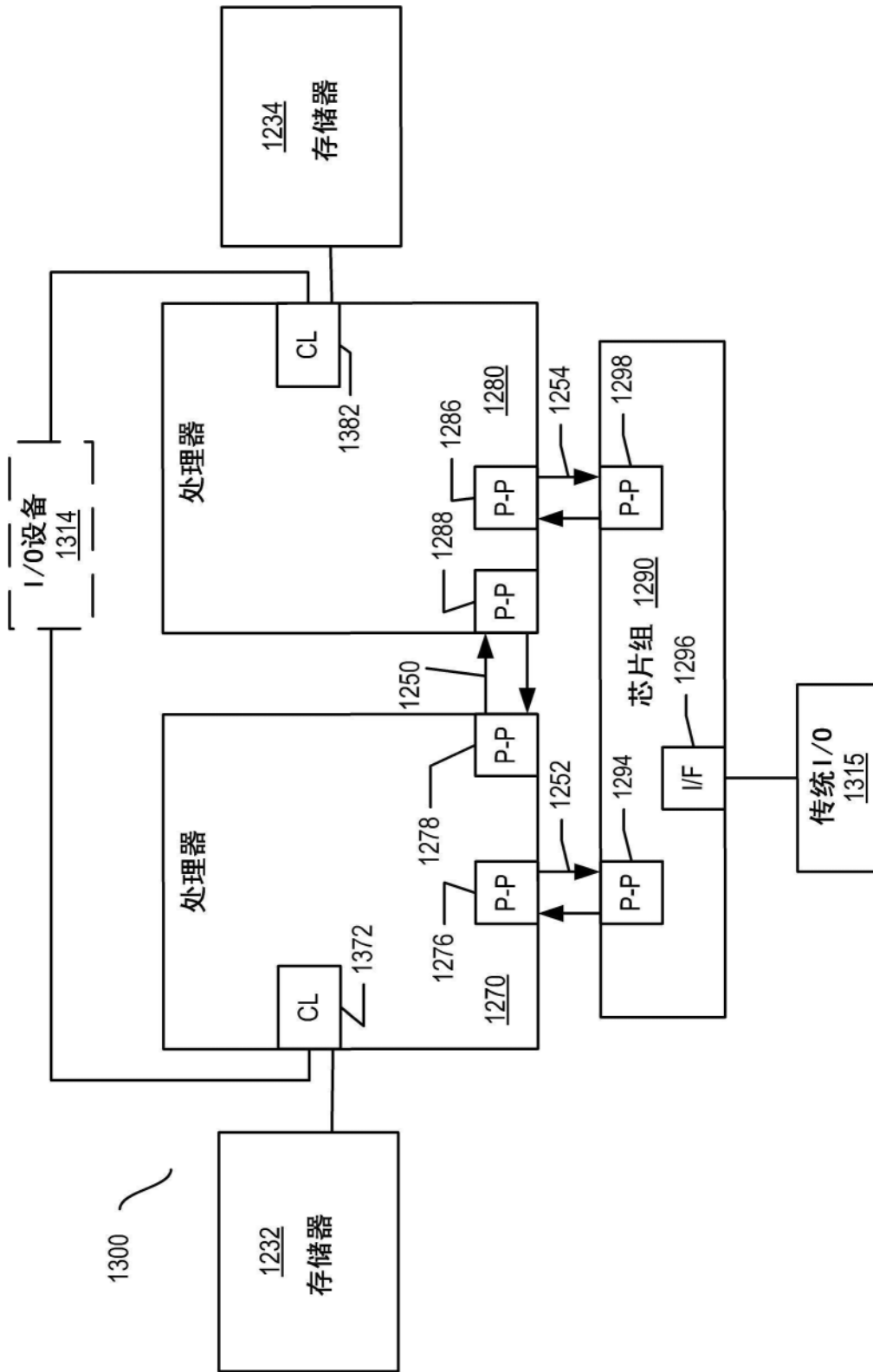


图13

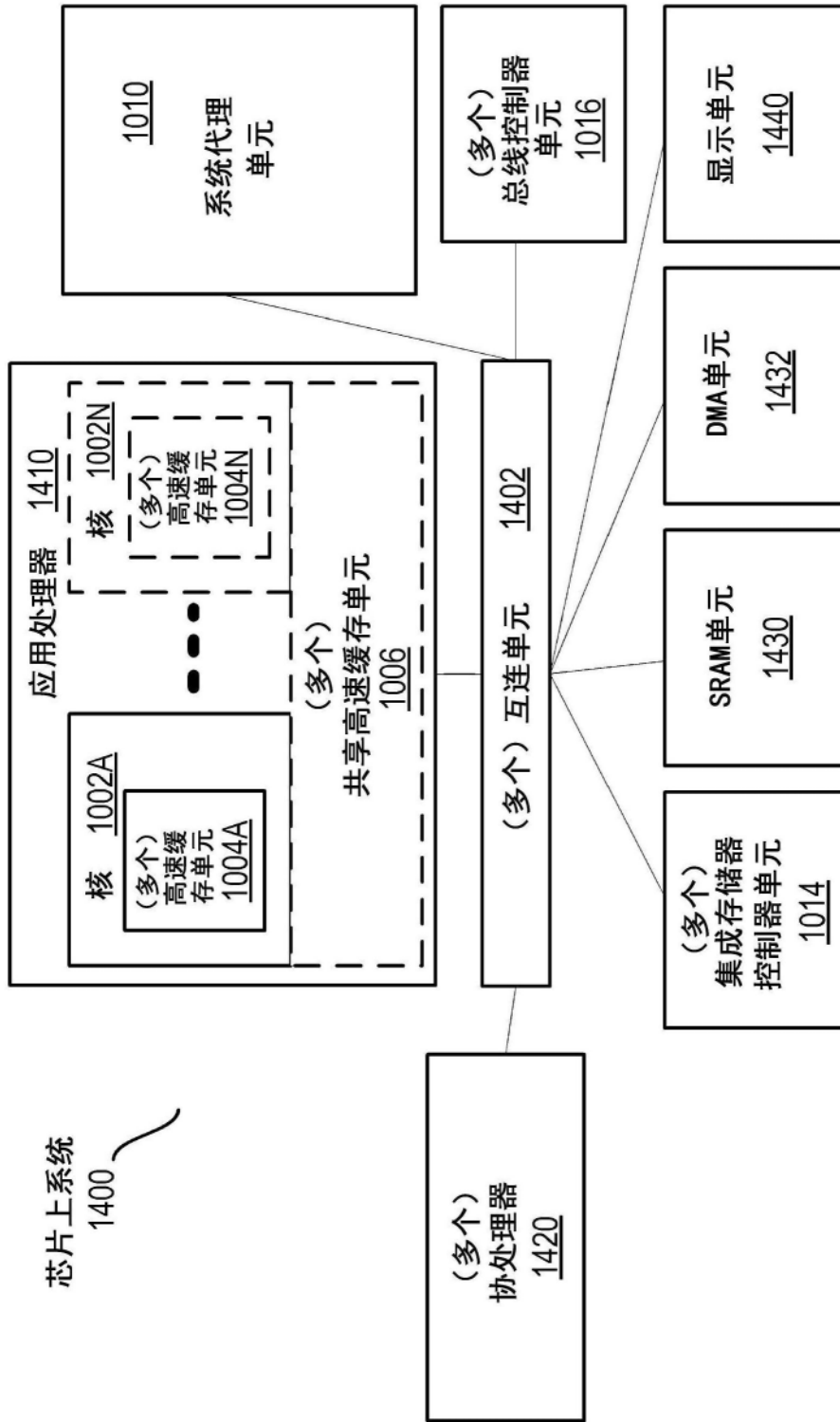


图14

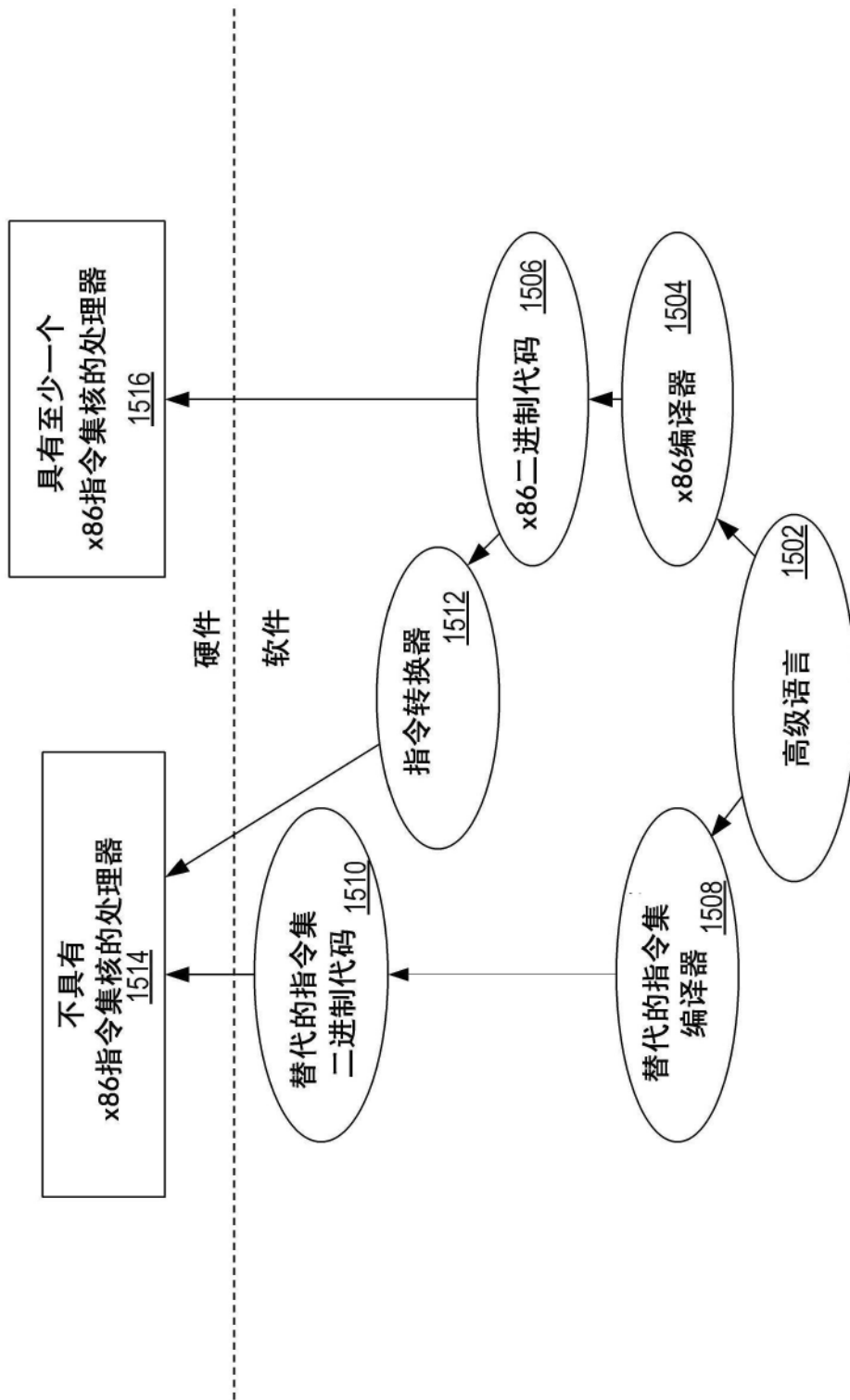


图15