

## (12) 发明专利

(10) 授权公告号 CN 101158893 B

(45) 授权公告日 2012. 10. 03

(21) 申请号 200710162252. 4

US 2005273582 A1, 2005. 12. 08,

(22) 申请日 2007. 10. 08

CN 1746842 A, 2006. 03. 15,

(30) 优先权数据

审查员 姜玲玲

0619522. 6 2006. 10. 03 GB

(73) 专利权人 ARM 有限公司

地址 英国剑桥郡

(72) 发明人 M · E · L · 樊尚 F · 贝贡  
C · D · R · 艾罗 N · B · E · 拉塔耶(74) 专利代理机构 北京东方亿思知识产权代理  
有限责任公司 11258

代理人 李晓冬

(51) Int. Cl.

G06F 9/38 (2006. 01)

(56) 对比文件

US 5548776 A, 1996. 08. 20,  
US 2005091475 A1, 2005. 04. 28,  
CN 1581069 A, 2005. 02. 16,  
EP 0918290 A1, 1999. 05. 26,  
CN 101046740 A, 2007. 10. 03,

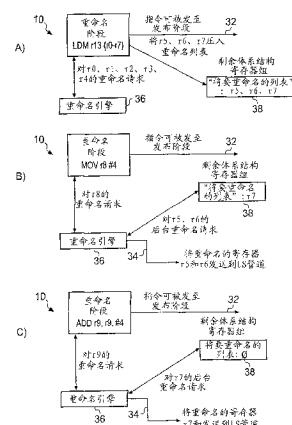
权利要求书 2 页 说明书 6 页 附图 3 页

(54) 发明名称

数据处理系统中的寄存器重命名

(57) 摘要

使用寄存器重命名的处理器 2 通过将重命名任务分成初始组和剩余组执行需要大量将要重命名的体系结构寄存器说明符的程序指令。首先执行初始组并将结果通过主通道 32 传送给用于进一步的处理。依次执行剩余组，结果通过后台通道 34 传送给用于进一步的处理。该技术对于为装入 / 存储多 LDM 指令执行重命名操作尤为有用。



1. 一种处理数据的设备,所述设备包括 :

寄存器重命名装置,用于将体系结构寄存器说明符组的寄存器说明符映射到物理寄存器组的寄存器,所述体系结构寄存器说明符组代表指令集中的指令指定的寄存器,并且所述物理寄存器组是用于处理所述指令集的指令的物理寄存器;其中

所述寄存器重命名装置响应于指定需要多于预定数量的寄存器映射的操作的程序指令,并包括 :

(i) 将所述多于预定数量的寄存器映射分成初始寄存器映射组和剩余寄存器映射组的装置;

(ii) 执行所述初始寄存器映射组产生用于物理寄存器的寄存器说明符,以在主通道上输出用于进一步的处理的装置;以及

(iii) 在执行所述初始寄存器映射组之后,在用于至少一些其他程序指令的寄存器说明符正在由所述寄存器重命名装置产生并在所述主通道上输出的同时,执行所述剩余寄存器映射组来产生用于物理寄存器的寄存器说明符,以在与所述主通道分离的后台通道上输出所述寄存器说明符的装置。

2. 权利要求 1 的设备,其中所述剩余寄存器映射组是作为寄存器映射序列而执行的,每一个寄存器映射产生将在所述后台通道上输出的用于物理寄存器的寄存器说明符并且产生减少的剩余寄存器映射组。

3. 权利要求 1 的设备,其中所述寄存器重命名装置具有用于存储表示所述剩余寄存器映射组的体系结构寄存器说明符的缓冲存储器。

4. 权利要求 3 的设备,其中所述缓冲存储器将所述体系结构寄存器说明符存储为有序列表,该有序列表表示所述剩余寄存器映射组将要被执行的顺序。

5. 权利要求 1 的设备,其中具有包含在所述剩余寄存器映射组中的一体体系结构寄存器说明符的程序指令直到所述体系结构寄存器说明符所代表的寄存器已经从所述剩余寄存器映射组中移除之后才经历寄存器映射。

6. 权利要求 1 的设备,其中如果任何剩余寄存器映射组是从先前的指定需要多于所述预定数量的寄存器映射的操作的程序指令剩余下来的,则所述寄存器重命名装置暂停指定所述需要多于所述预定数量的寄存器映射的操作的所述程序指令。

7. 权利要求 1 的设备,包括装入 / 存储单元,并且其中所述程序指令是由所述装入 / 存储单元执行的装入 / 存储多指令。

8. 权利要求 7 的设备,其中所述后台通道将用于所述物理寄存器的寄存器说明符直接传送到所述装入 / 存储单元。

9. 权利要求 7 的设备,其中所述装入 / 存储单元能够执行装入 / 存储操作,该操作在处理循环中消耗存储在 N 个物理寄存器中的值,并且所述后台通道被配置为在处理循环中向所述装入 / 存储单元传送用于多达 N 个物理寄存器的寄存器说明符。

10. 一种处理数据的方法,所述方法包括 :

映射步骤:将体系结构寄存器说明符组的寄存器说明符映射到物理寄存器组的寄存器,所述体系结构寄存器说明符组代表指令集中的指令指定的寄存器并且所述物理寄存器组是用于处理所述指令集的指令的物理寄存器;其中

响应于指定需要多于预定数量的寄存器映射的操作的程序指令,所述映射步骤为:

- (i) 将所述多于预定数量的寄存器映射分成初始寄存器映射组和剩余寄存器映射组；
  - (ii) 执行所述初始寄存器映射组产生用于物理寄存器的寄存器说明符，以在主通道上输出用于进一步的处理；以及
  - (iii) 在执行所述初始寄存器映射组之后，在用于至少一些其他程序指令的寄存器说明符正在由所述映射步骤产生并在所述主通道上输出的同时，执行所述剩余寄存器映射组来产生用于物理寄存器的寄存器说明符，以在与所述主通道分离的后台通道上输出所述寄存器说明符。
11. 权利要求 10 的方法，其中所述剩余寄存器映射组是作为寄存器映射序列执行的，每个寄存器映射产生将在所述后台通道上输出的用于物理寄存器的寄存器说明符并且产生减少的剩余寄存器映射组。
12. 权利要求 10 的方法，包括在缓冲存储器中存储表示所述剩余寄存器映射组的体系结构寄存器说明符。
13. 权利要求 12 的方法，其中所述缓冲存储器将所述体系结构寄存器说明符存储为有序列表，该有序列表表示所述剩余寄存器映射组将要被执行的顺序。
14. 权利要求 10 的方法，其中具有包括在所述剩余寄存器映射组中的一体体系结构寄存器说明符的程序指令直到所述体系结构寄存器说明符所代表的寄存器已经从所述剩余寄存器映射组中移除之后才受到寄存器映射。
15. 权利要求 10 的方法，其中如果任何剩余寄存器映射组是从指定需要多于所述预定数量的寄存器映射的操作的先前的程序指令剩余下来的，则暂停指定所述需要多于所述预定数量的寄存器映射的操作的所述程序指令。
16. 权利要求 10 的方法，其中所述程序指令是由装入 / 存储单元执行的装入 / 存储多指令。
17. 权利要求 16 的方法，其中所述后台通道将用于所述物理寄存器的寄存器说明符直接传送到所述装入 / 存储单元。
18. 权利要求 16 的方法，其中所述装入 / 存储单元能够执行装入 / 存储操作，该操作在处理循环中消耗存储在 N 个物理寄存器中的值，并且所述后台通道被配置为在处理循环中向所述装入 / 存储单元传送用于多达 N 个物理寄存器的寄存器说明符。

## 数据处理系统中的寄存器重命名

### 技术领域

[0001] 本发明涉及数据处理系统。更具体地，本发明涉及使用寄存器重命名的数据处理系统，借此一组体系结构寄存器的寄存器说明符映射到一组物理寄存器的寄存器说明符。

### 背景技术

[0002] 寄存器重命名是用于帮助支持无序处理的公知技术。典型地，这种系统都针对高性能应用，其中减少门数和减少功率消耗并不是最主要的因素。

[0003] 当处理引用大量的寄存器的程序指令，并且每个寄存器都需要重命名时，寄存器重命名技术就出现了问题。支持一次重命名大量寄存器所必需的机制需要不利的大的门数以及相关的不利的大量功率消耗。此外，一旦重命名机制重命名了寄存器，那么所有这些重命名的寄存器说明符必须沿着指令流水线被并行地传送到将使用它们的单元。沿着指令流水线并行地传输这么大量的重命名的寄存器说明符也需要不利的门数和面积代价，功率消耗也有相关的不利增加。虽然在那些门数、面积和功率消耗并不是特殊约束的系统中上述问题也许不重要，但是在需要保持低门数、小面积和低功耗的不昂贵的、节约能量的小处理器环境中这些却是严重的问题。

[0004] 上述的替代方案是暂停需要大量寄存器重命名操作的指令，使得这些能够在几个处理周期中完成。然而，这会不利地暂停后续程序指令，并且一旦产生了重命名的寄存器说明符则还需要沿着流水线经过不适宜的宽路径并行传输这些寄存器说明符。

### 发明内容

[0005] 从一个方面来看，本发明提供了用于处理数据的设备，所述设备包括：

[0006] 物理寄存器组；

[0007] 寄存器重命名电路，其响应于体系结构寄存器说明符组的寄存器说明符将所述寄存器说明符映射到所述物理寄存器组的寄存器，所述体系结构寄存器说明符组代表指令集中的指令指定的寄存器，所述物理寄存器组是用于处理所述指令集的指令的物理寄存器；其中

[0008] 所述寄存器重命名电路响应于指定需要多于预定数目的寄存器映射的操作的程序指令以：

[0009] (i) 将所述预定数目的寄存器映射分成初始寄存器映射组和剩余寄存器映射组；

[0010] (ii) 执行所述初始寄存器映射组产生用于物理寄存器的寄存器说明符，以在主通道上输出用于进一步的处理；以及

[0011] (iii) 在执行所述初始寄存器映射组之后，在用于至少一些其他程序指令的寄存器说明符正在由所述寄存器重命名电路产生并在所述主通道上输出的同时，执行所述剩余寄存器映射组来产生用于物理寄存器的寄存器说明符，以在与所述主通道分离的后台(background) 通道上输出。

[0012] 本技术认识到在实际中，需要大量物理寄存器说明符的指令的物理寄存器说明符

实际上不会一次全部消耗掉,所以也就不需要在单次操作中产生并沿指令流水线一起传输。本技术通过从寄存器重命名电路中提供主通道和后台通道用于输出用于进一步处理的寄存器说明符来开发这种认识。因此,初始寄存器说明符组和用于后续程序指令的寄存器说明符可以沿着主通道向下从寄存器重命名逻辑输出用于进一步的处理。这提供了开始执行需要大量寄存器说明符的程序指令的进一步处理所需要的初始寄存器说明符是可用的而沿正常主通道向下没有延迟,而且后续程序指令也可以沿正常主通道向下提供它们的寄存器说明符。后台通道提供了这样的机制,借此机制,可以传输用于需要很多寄存器说明符的指令的剩余寄存器说明符组以作进一步的处理,与此同时其他寄存器说明符被沿着主通道传送。

[0013] 作为示例,考虑 ARM 指令集的装入 / 存储多指令。例如,这样的指令可以指定从存储器读出的数据值装入八个寄存器。这个装入 / 存储多指令由装入 / 存储单元执行,在采用寄存器重命名的系统环境中,需要给装入 / 存储单元提供用于重命名的物理寄存器组的寄存器说明符,使得其可以将数据值装入到合适的物理寄存器中。然而,这种装入 / 存储单元典型地具有有限的并行装载数据值的能力并且这也许小于将要发生的寄存器装载的总数量。因此,如果装入 / 存储单元可被沿着主通道传送启动整个装入 / 存储操作所需要的寄存器说明符并且通过后台通道馈给随后的寄存器说明符,那么可以在执行装入 / 存储多指令中合理使用装入 / 存储单元而不需要暂停在重命名阶段以允许所有的重命名操作一次完成;也不需要从重命名阶段并行传送很多的寄存器说明符;或也不需要阻止后续程序指令能够在传送剩余寄存器说明符的同时通过主通道传送它们的重命名物理寄存器说明符。

[0014] 虽然后台通道有可能在单次操作中传送所有的剩余寄存器说明符组用于进一步的处理,但是更有效、更灵活的是提供了后台通道以序列的形式输出剩余寄存器说明符组。

[0015] 为了参与其中,寄存器重命名逻辑具有用于存储体系结构寄存器说明符的缓冲存储器,体系结构寄存器说明符表示将要执行的剩余寄存器映射组。这个缓冲存储器可以作为有序表存储体系结构寄存器说明符,有序表表示剩余寄存器映射组将被执行和输出的次序,并且这个次序可以与期望电路所使用的次序匹配,其中该电路以保持与体系结构模型一致的方式使用那些寄存器说明符执行进一步处理,在该体系结构模型中产生并一次传输寄存器说明符用于进一步的处理。

[0016] 可能产生于本技术的限制是如果体系结构寄存器说明符被包含在将要执行的剩余寄存器映射组中,那么包括那个体系结构寄存器说明符的后续程序指令直到先前的程序指令及其相关的剩余体系结构寄存器说明符组已被处理到这样的程度以致有关的体系结构寄存器说明符已经受到合适的重映射之前不会发生重映射。这保持了重命名操作的上游以及包括重命名操作的有序执行,因为是重命名操作以及典型地相关的恢复机制和联合机制允许容忍无序执行。

[0017] 可发生的另一限制是指令将不会使用剩余映射组除非来自任何先前指令的剩余要被执行的映射的剩余映射组是空的。

[0018] 将理解的是具有超出预定数量的并且要求使用主通道和后台通道的大量寄存器说明符的程序指令可以有各种不同的形式。本技术可以采用的一种特别的示例形式是由相关的装入 / 存储单元执行的装入 / 存储指令。在本文上下文中,后台通道可以将用于物理寄存器的寄存器说明符直接传送到装入 / 存储单元。如果装入 / 存储单元能够在任意处理

器循环中消耗 N 个寄存器说明符,那么这个可以被用来选择后台通道的宽度使得其可以支持在处理循环中传输多达 N 个寄存器说明符以便按照希望的程度保持装入 / 存储单元完全被占用。

[0019] 从另一方面来看,本发明提供了一种处理数据的方法,所述方法包括以下步骤:

[0020] 将体系结构寄存器说明符组的寄存器说明符映射到物理寄存器组的寄存器,所述体系结构寄存器说明符组代表指令集中的指令指定的寄存器,所述物理寄存器组是用于处理所述指令集的指令的物理寄存器;其中

[0021] 响应于指定需要多于预定数量的寄存器映射的操作的程序指令,所述映射步骤为:

[0022] (i) 将所述预定数量的寄存器映射分成初始寄存器映射组和剩余寄存器映射组;

[0023] (ii) 执行所述初始寄存器映射组产生用于物理寄存器的寄存器说明符以在主通道上输出用于进一步的处理;以及

[0024] (iii) 在执行所述初始寄存器映射组之后,在用于至少一些其他程序指令的寄存器说明符正在由所述映射步骤产生并在所述主通道上输出的同时,执行所述剩余寄存器映射组来产生用于物理寄存器的寄存器说明符,以在与所述主通道分离的后台通道上输出。

[0025] 从又一方面看来,本发明提供了用于处理数据的设备,所述设备包括:

[0026] 物理寄存器装置组;

[0027] 寄存器重命名装置,其响应于体系结构寄存器说明符组的寄存器说明符用于将所述寄存器说明符映射到所述物理寄存器组的寄存器,所述体系结构寄存器说明符组代表指令集中的指令指定的寄存器,所述物理寄存器组是用于处理所述指令集的指令的物理寄存器;其中

[0028] 所述寄存器重命名装置响应于程序指令,该程序指令说明需要多于预定数量的寄存器映射的操作,以便:

[0029] (i) 将所述预定数量的寄存器映射分成初始寄存器映射组和剩余寄存器映射组;

[0030] (ii) 执行所述初始寄存器映射组产生用于物理寄存器的寄存器说明符以在主通道上输出用于进一步的处理;以及

[0031] (iii) 在执行所述初始寄存器映射组之后,在用于至少一些其他程序指令的寄存器说明符正在由所述寄存器重命名装置产生并在所述主通道上输出的同时,执行所述剩余寄存器映射组来产生用于物理寄存器的寄存器说明符,以在与所述主通道分离的后台通道上输出。

[0032] 结合附图阅读以下说明性实施例的详细描述将清楚理解本发明的上述以及其他目的、特征和优势。

## 附图说明

[0033] 图 1 示意性地示出了数据处理装置,其包括具有寄存器重命名阶段的指令流水线并且包括用于传输寄存器说明符的主通道和后台通道

[0034] 图 2 示意性地示出了在处理装入 / 存储多指令的不同时间的寄存器重命名电路,其形成了图 1 的系统的一部分。

[0035] 图 3 是示意性地示出了使用传输寄存器说明符的主通道和后台通道的寄存器重

命名电路的操作的流程图。

### 具体实施方式

[0036] 图1示出了数据处理设备2,其包括指令高速缓冲存储器4,指令高速缓冲存储器4将程序指令供给到指令流水线中,指令流水线包括预取阶段6、解码阶段8、重命名阶段10、发布阶段12、执行阶段14(利用相关的物理寄存器15)和写回阶段16。将理解的是实际中这种数据处理设备2中可以包括附加的和/或可选的流水线阶段。此外,将理解的是完整的数据处理设备2典型地包括许多附加元件,这对于本技术领域的技术人员是显而易见的并且为清楚起见已从图1中省略。

[0037] 执行阶段14包括可以并行处理程序指令的多个单元,其包括第一主执行单元18、乘法累加单元20、第二主执行单元22和装入/存储地址阶段24(其将请求发送到装入/存储单元,其中装入/存储单元是系统数据侧的状态机)。第一主执行单元18和乘法累加单元20共享写回单元26。第二主执行单元22和装入/存储地址阶段24每一个分别具有其自己的写回单元28、30。如超标量系统中常见的,可以从发布阶段12将多指令并行发布到执行阶段14的各个合适的部分用于并行执行。重命名阶段10允许发布阶段12凭借执行的寄存器重命名进行无序指令发布。方便无序执行的寄存器重命名其本身是已知的技术不需要在此处进一步阐述。本技术对有序指令发布系统也是有用的,例如,避免沿流水线将所有的寄存器说明符流水线化。

[0038] 本技术的重命名阶段10具有主通道输出32和后台通道输出34用于传输寄存器说明符以及其他控制信号到指令流水线的下游。主通道32用于需要相对较少的寄存器说明符并且因此具有与这种正常指令匹配的宽度的大多数指令,所述宽度例如是能够传送与三个要读的寄存器和两个要写的寄存器相关的寄存器说明符的宽度,虽然将理解的是这些度量仅作为示例并且可以根据有关应用使用不同的宽度。后台通道34具有的宽度能够独立于任何正沿主通道32传输的信号支持每个处理循环提供两个寄存器说明符到装入/存储地址阶段24。

[0039] 当碰到装入/存储指令需要超过主通道容量(即大于预定数量)的数量的寄存器说明符时,将要执行的寄存器映射分成初始组和剩余组。首先执行初始组,并将结果沿着正常的主通道32从重命名阶段10向下传送到发布阶段12。作为后台操作在随后的处理循环中执行剩余重映射组并将结果沿着后台通道34一次两个地直接传送到装入/存储位置阶段24。因此,一旦初始映射组已被重命名阶段10执行并沿主通道32传输,则主通道32和重命名阶段10的至少一些重命名容量可用于处理后续程序指令,而不需要暂停(stall)这些后续程序指令。在后续程序指令指定包含在要执行的剩余重映射组中的体系结构寄存器的情况下会出现例外,因为重映射本身被限制要顺序执行。就停止(halt)了这样的后续程序指令。

[0040] 图2示出了执行与装入多指令和后续程序指令相关的映射操作的重命名阶段10。在状态A,装入多指令LDM在重命名阶段10接收并指定将要用从寄存器r13内保存的值所指定的存储器地址开始的存储器中依次读取的数据值装载寄存器r0-r7。在这个示例中重命名阶段10能够沿着主通道32一次传输五个寄存器说明符到发布阶段12,但是这对完全执行LDM指令所需要的传输所有八个重映射的寄存器说明符来说是不够的。包括在重命名

阶段 10 的重命名引擎 (renaming engine) 36 能够在一个处理循环中形成五个寄存器重命名操作。因此,在点 A 所示的状态,执行寄存器 r0-r4 的重命名并且将这些重命名的寄存器说明符与和 LDM 指令相关的控制信号一起沿主通道 32 输出到发布阶段 12 和装入 / 存储地址阶段 24。提供了缓冲存储器 38,其中以“将要重命名的列表”存储用于尚未重命名的剩余体系结构寄存器组的说明符。

[0041] 状态 B 示出了后续循环,其中后续指令 MOV 指示用指定的数据值装载寄存器 r8。用于有关寄存器,即 r8,的重命名请求被传送到重命名引擎 36 并且将返回的重命名的寄存器说明符与和 MOV 指令相关的控制信号一起沿着主通道 32 向下发布到发布阶段 12。与此同时,在代表剩余体系结构寄存器组的“将要重命名的列表”中的下两个体系结构寄存器被传送到重命名引擎 36 用于重命名。在这个示例中,存储在缓冲存储器 8 中的有序列表中的下两个体系结构寄存器是寄存器 r5 和 r6。将这些寄存器 r5 和 r6 从“将要重命名的列表”中移除,由重命名引擎 36 对其进行映射并在后台通道 34 上沿其装入 / 存储流水线直接输出到装入存储地址阶段 24。状态 B 在缓冲存储器 38 中留下一个体系结构寄存器 r7 剩余用于重映射。

[0042] 状态 C 示出了下一个处理循环,其中执行 ADD 指令,该指令将固定值加到寄存器 r9 中保存的数据值并且将该结果存回寄存器 r9 中。有关寄存器 r9 的重命名请求被发送到重命名引擎 36。“将要重命名的列表”中的最后的体系结构寄存器 r7 也被从缓冲存储器 38 传输到重命名引擎 36 并进行重命名,然后在后台通道 34 上输出到装入 / 存储流水线。这使得缓冲存储器 38 变空。与此同时,将用于与寄存器 r9 有关的 ADD 指令的重命名的寄存器说明符及与 ADD 指令相关的控制信号一起沿主通道 32 传送出。

[0043] 图 3 是示出重命名阶段 10 的操作的流程图。将要理解的是图 3 的流程图有必要地将操作按照顺序方式执行的那样呈现出来,而这在硬件操作中不一定能反映出来。然而,整个操作达到同样的结果,并且对本领域的技术人员显而易见的是如何可以提供各种能提供图 3 所示操作的硬件实现。

[0044] 在步骤 42,重命名阶段 10 等待接收需要重命名的指令。步骤 44 检测接收的指令是否包括重映射体系结构寄存器的要求,其中该体系结构寄存器已经借助存储在缓冲存储器 38 的“将要重命名的列表”中的已经受到待处理的重映射操作。如果检测到这样的待处理的重映射,那么重命名阶段 10 等待执行新的重映射,直到已经完成了这个先前的重映射之后。

[0045] 在步骤 46,重命名阶段 10 确定新程序指令是否需要执行大于预定数量的寄存器重映射。作为示例,指定仅需装载几个寄存器的 LDM 指令可以被重映射并传送到流水线的剩余部分,而不需要使用后台通道 34。然而,包括长寄存器列表的 LDM 指令将需要使用后台通道 34

[0046] 如果没有超过预定的数量,那么处理进行到步骤 48,其中由重命名引擎 36 执行寄存器重命名,然后到步骤 50,将物理寄存器说明符和有关的指令一起沿主通道 32 输出到发布阶段 12。

[0047] 如果超过了预定的数量,那么处理转到步骤 47,其中确定“将要重命名的列表”是否为空。如果“将要重命名的列表”非空,则暂停步骤 42 接收的用于重命名的指令直到“将要重命名的列表”为空,以避免将“将要重命名的列表”中来自不同指令的体系结构寄存器

说明符混合。这个暂停条件是很少见的，在许多情况中导致其他原因的暂停。在步骤 52，需要重命名的前五个寄存器受到重命名操作。需要重命名的剩余的体系结构寄存器说明符被存入缓冲存储器 38 的“将要重命名的列表”。在步骤 54，用于已经重映射的该五个寄存器的物理寄存器说明符与相关指令一起在主通道 32 上传送出。（图 3 的步骤 54 和 50 相对于需要重映射的指令的接收通常具有同样的定时。）在阶段 54 之后，在下一个处理循环，使用重命名引擎 36 对存储在缓冲存储器 38 中的“将要重命名的列表”上的直到下两个寄存器进行重命名。这在图 2 的阶段 B 和 C 中示出了。在步骤 58，用于剩余体系结构寄存器说明符组的重命名的成员的物理寄存器说明符从通入装入 / 存储流水线的后台通道 34 输出到装入存储地址阶段 24。步骤 60 确定存储在缓冲存储器 38 中的“将要重命名的列表”中是否剩余任何寄存器。如果有，则处理返回到步骤 56。

[0048] 虽然此处参考附图详细描述了本发明的说明性实施例，将要理解的是本发明并不限于这些具体的实施例，并且本领域的技术人员可以作出各种改变和修改而不脱离随附权利要求所定义的本发明的范围和精神。

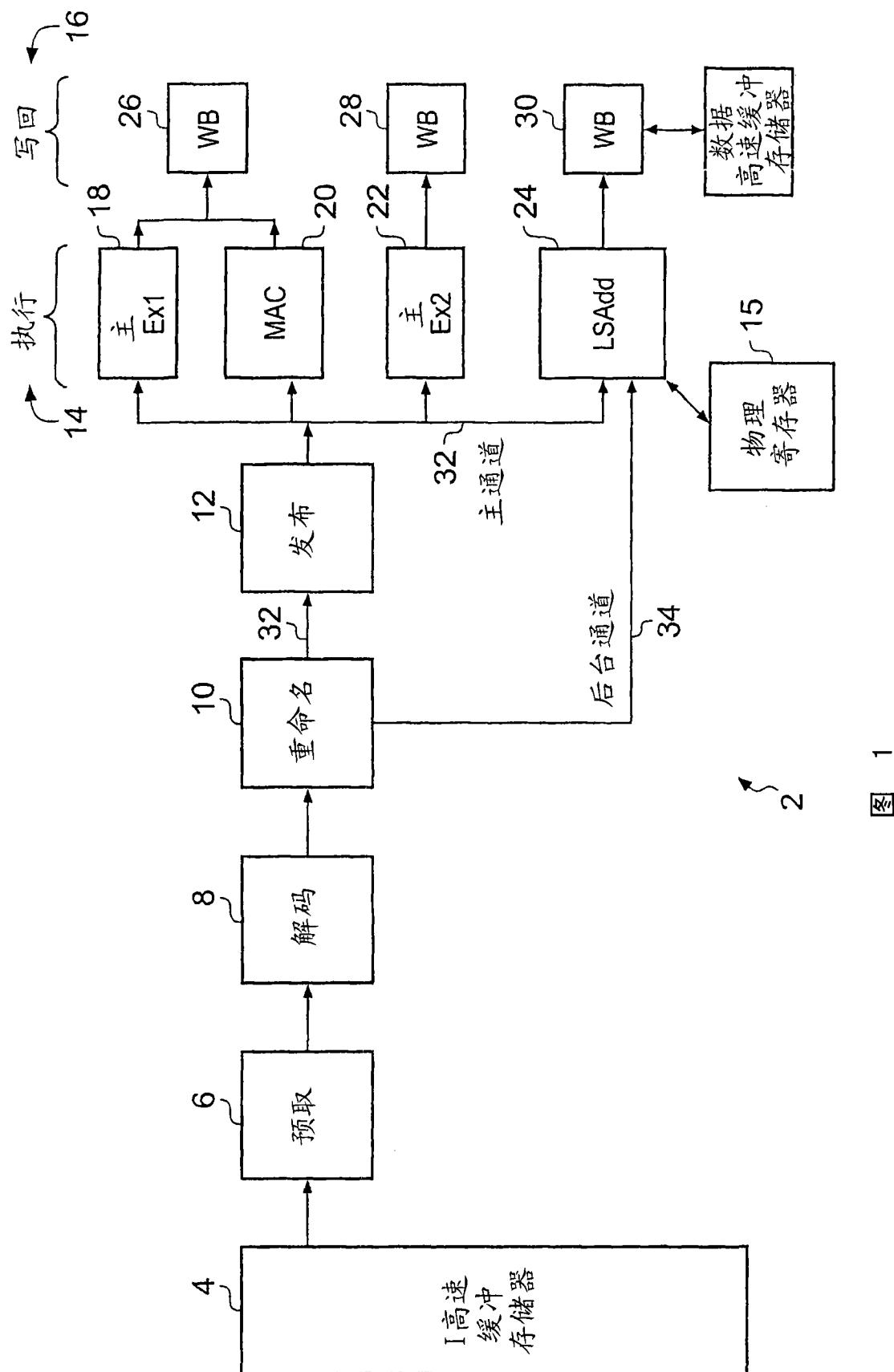


图 1

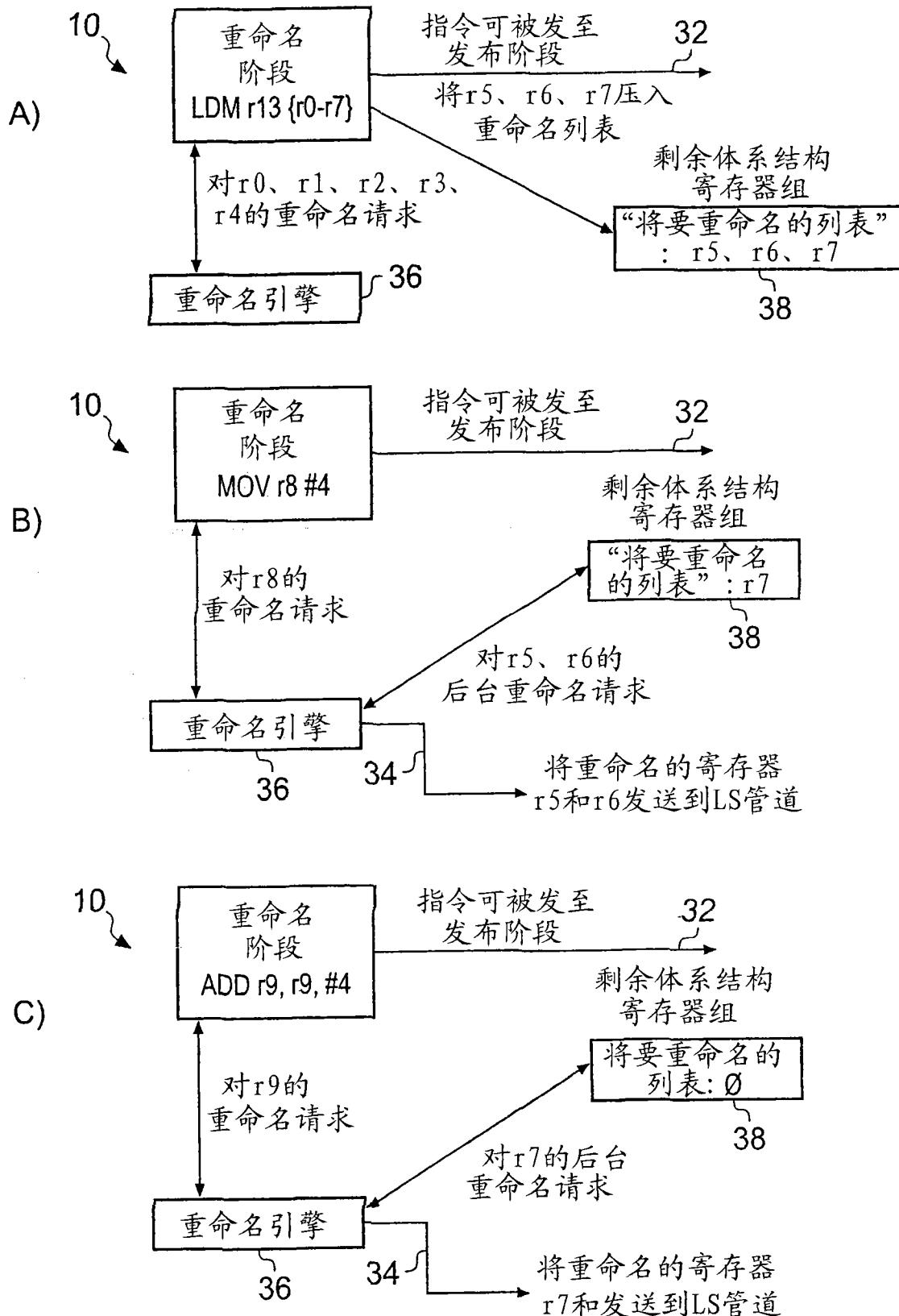


图 2

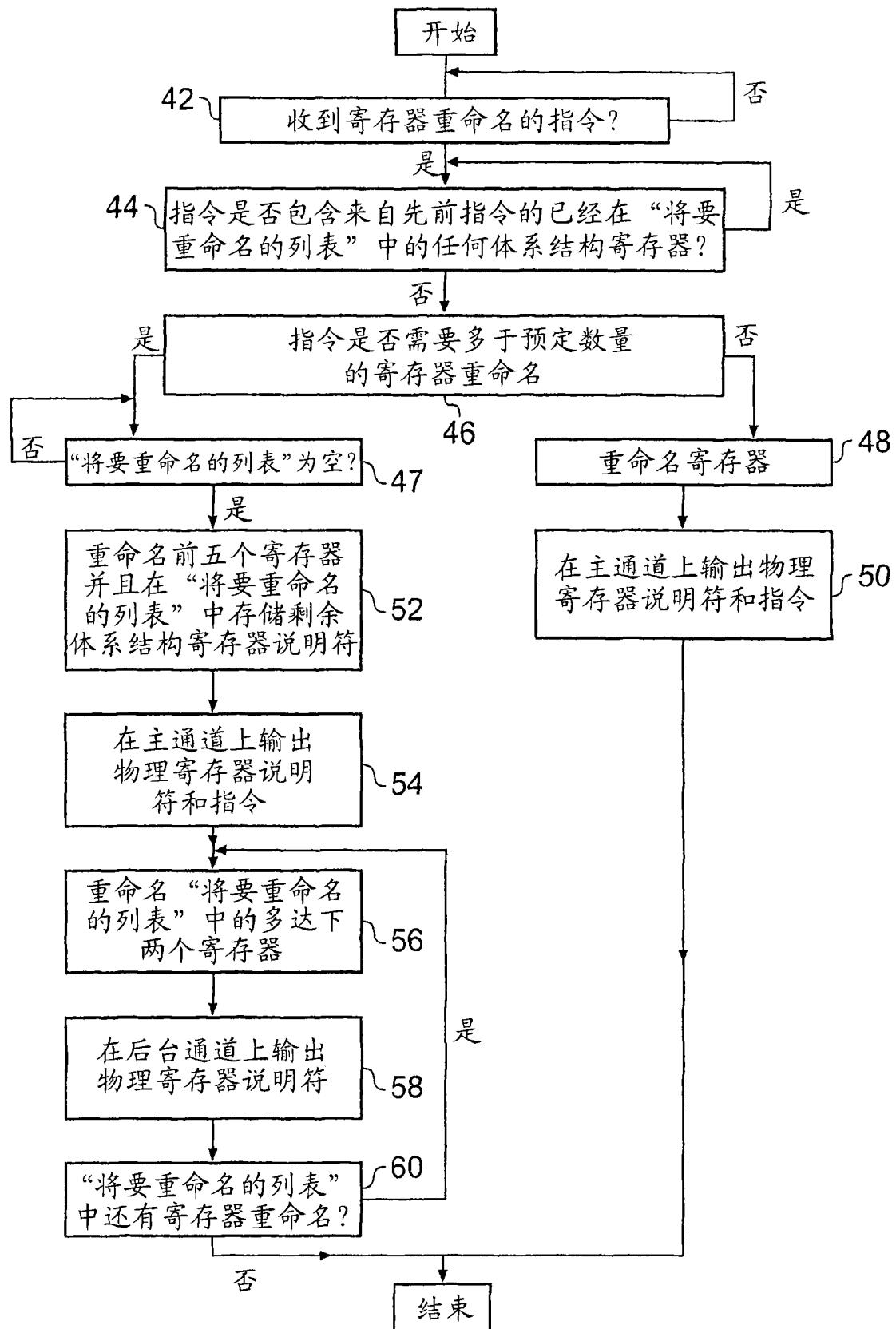


图 3