



(12) 发明专利

(10) 授权公告号 CN 101278269 B

(45) 授权公告日 2011.08.10

(21) 申请号 200680036685.9

G06F 12/10 (2006.01)

(22) 申请日 2006.09.27

(56) 对比文件

(30) 优先权数据

EP 0447145 B1, 2000.07.12, 全文.

11/246,585 2005.10.07 US

EP 0905629 A, 1999.03.31, 全文.

(85) PCT申请进入国家阶段日

US 2004/0064669 A1, 2004.04.01, 说明书第 [0044]-[0060] 段、附图 3A, 3B, 8.

2008.04.02

说明书第 [0044]-[0060] 段、附图 3A, 3B, 8.

(86) PCT申请的申请数据

PCT/EP2006/066806 2006.09.27

US 5347643 A, 1994.09.13, 全文.

(87) PCT申请的公布数据

W02007/042400 EN 2007.04.19

US 2004/0015622 A1, 2004.01.23, 说明书第 [0058] - [0061] 段.

审查员 石志昕

(73) 专利权人 国际商业机器公司

地址 美国纽约

(72) 发明人 M·E·金 培军·P·刘 D·穆伊

齐洁明

(74) 专利代理机构 中国国际贸易促进委员会专

利商标事务所 11038

代理人 吴丽丽

(51) Int. Cl.

G06F 13/28 (2006.01)

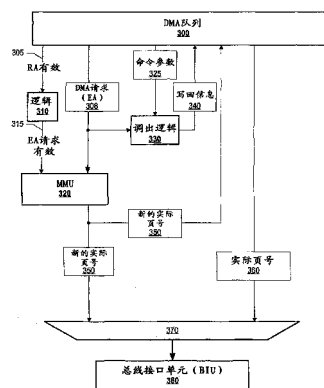
权利要求书 3 页 说明书 9 页 附图 9 页

(54) 发明名称

用于改进的 DMAC 翻译机制的系统和方法

(57) 摘要

提供了一种用于改进的 DMAC 翻译机制的系统和方法。DMA 命令基于该 DMA 命令的传输大小和计算机系统一次传输的数据量而被“调出”。对于第一个 DMA 请求, DMA 队列请求存储器管理单元执行地址翻译。DMA 队列从 MMU 接收实际页号, 在后面的调出请求时, DMA 队列将实际页号提供给总线接口单元而不访问 MMU, 直到传输跨入下一页。调出逻辑在每个 DMA 请求后使 DMA 命令的传输大小递减, 确定是否已到达了新的页, 确定是否已完成 DMA 命令, 并且发送写回信息到 DMA 队列用于后面的 DMA 请求。



1. 一种计算机实施的方法,包括:

在 DMA 队列处选择 DMA 命令,该 DMA 命令包括多个 DMA 命令字段,所述多个 DMA 命令字段包括实际页号字段和传输大小字段;

确定包括在所述实际页号字段中的实际页号是否有效;以及

响应于确定所述实际页号有效,从所述 DMA 队列提供所述实际页号以处理所述 DMA 命令;

识别包括在所述传输大小字段中的传输大小,该传输大小指示对于所述 DMA 命令要传输的数据量;

检测与对于相应的 DMA 命令要传输的数据量相对应的最大传输大小允许值;

从所述传输大小减去所述最大传输大小允许值,该减法操作产生递减后的传输大小;

响应于提供所述实际页号,接收写回信息,该写回信息包括所述递减后的传输大小和所述实际页号;以及

将接收到的实际页号存储在所述实际页号字段中,将所述递减后的传输大小存储在所述传输大小字段中,所述递减后的传输大小指示在处理所述 DMA 请求后对于所述 DMA 命令剩余要传输的数据量。

2. 根据权利要求 1 所述的计算机实施的方法,还包括:

检查包括在所述多个 DMA 命令字段之一中的实际地址有效位,该实际地址有效位表明所述实际页号是否有效;

响应于确定所述实际页号不是有效的,向存储器管理单元发送翻译请求;以及

从所述存储器管理单元提供新的实际页号以处理所述 DMA 命令。

3. 根据权利要求 2 所述的计算机实施的方法,还包括:

配置复用器在所述实际页号有效的情况下选择所述实际页号,在所述实际页号不是有效的情况下选择所述新的实际页号。

4. 根据权利要求 2 所述的计算机实施的方法,还包括:

确定所述存储器管理单元是否找到了与所述翻译请求相对应的新的实际页号;

响应于确定所述存储器管理单元找到了该新的实际页号,将该新的实际页号存储在所述实际页号字段中;以及

响应于确定所述存储器管理单元找到了该新的实际页号,置位所述实际地址有效位。

5. 根据权利要求 1 所述的计算机实施的方法,还包括:

在所述减法操作后,使对应于所述 DMA 命令的有效地址递增;

基于递增后的有效地址确定是否到达了页的末端;以及

响应于确定到达了页的末端,复位包括在所述多个 DMA 命令字段之一中的实际地址有效位。

6. 根据权利要求 1 所述的计算机实施的方法,其中,所述多个 DMA 命令字段中的至少一个是从队列条目有效位字段和实际地址有效位字段中选出的。

7. 根据权利要求 1 所述的计算机实施的方法,还包括:

其中,使用宽带处理器体系结构来执行该方法,该宽带处理器体系结构包括多个异构处理器、公共存储器和公共总线;以及

其中,所述多个异构处理器使用不同的指令集并且共享所述公共存储器和所述公共总

线。

8. 根据权利要求 7 所述的计算机实施的方法,其中,所述宽带处理器体系结构是片上系统。

9. 一种信息处理系统,包括:

用于在 DMA 队列处选择 DMA 命令的装置,该 DMA 命令包括多个 DMA 命令字段,所述多个 DMA 命令字段包括实际页号字段和传输大小字段;

用于确定包括在所述实际页号字段中的实际页号是否有效的装置;

用于响应于确定所述实际页号有效,从所述 DMA 队列提供所述实际页号以处理所述 DMA 命令的装置;

用于识别包括在所述传输大小字段中的传输大小的装置,该传输大小指示对于所述 DMA 命令要传输的数据量;

用于检测与对于相应的 DMA 命令要传输的数据量相对应的最大传输大小允许值的装置;

用于从所述传输大小减去所述最大传输大小允许值的装置,该减法操作产生递减后的传输大小;

用于响应于提供所述实际页号,接收写回信息的装置,该写回信息包括所述递减后的传输大小和所述实际页号;以及

用于将接收到的实际页号存储在所述实际页号字段中,将所述递减后的传输大小存储在所述传输大小字段中的装置,所述递减后的传输大小指示在处理所述 DMA 请求后对于所述 DMA 命令剩余要传输的数据量。

10. 根据权利要求 9 所述的信息处理系统,还包括:

用于检查包括在所述多个 DMA 命令字段之一中的实际地址有效位的装置,该实际地址有效位表明所述实际页号是否有效;

用于响应于确定所述实际页号不是有效的,向存储器管理单元发送翻译请求的装置;以及

用于从所述存储器管理单元提供新的实际页号以处理所述 DMA 命令的装置。

11. 根据权利要求 10 所述的信息处理系统,还包括:

用于配置复用器在所述实际页号有效的情况下选择所述实际页号,在所述实际页号不是有效的情况下选择所述新的实际页号的装置。

12. 根据权利要求 10 所述的信息处理系统,还包括:

用于确定所述存储器管理单元是否找到了与所述翻译请求相对应的新的实际页号的装置;

用于响应于确定所述存储器管理单元找到了该新的实际页号,将该新的实际页号存储在所述实际页号字段中的装置;以及

用于响应于确定所述存储器管理单元找到了该新的实际页号,置位所述实际地址有效位的装置。

13. 根据权利要求 9 所述的信息处理系统,其中,该信息处理系统是宽带处理器体系结构,该宽带处理器体系结构包括多个异构处理器、公共存储器和公共总线;并且其中,所述多个异构处理器使用不同的指令集并且共享所述公共存储器和所述公共总线。

14. 根据权利要求 13 所述的信息处理系统,其中,所述宽带处理器体系结构是片上系统。

用于改进的 DMAC 翻译机制的系统和方法

技术领域

[0001] 本发明涉及用于改进的 DMA 控制器翻译机制的系统和方法。更具体地,本发明涉及用于将实际页号存储在 DMA 命令字段中,以便在 DMA 命令调出期间使用实际页号的系统和方法。

背景技术

[0002] 计算机系统应用一般指定一个有效地址或间接地址来存储和取回数据。为了完成数据存储或取出请求,计算机系统将有效地址翻译成与存储器中的物理位置相对应的实际地址或物理地址。

[0003] 计算机系统一般包括用于提供有效地址到实际地址翻译的存储器管理单元(MMU)。在一种配置中,由于所有的加载和存储指令需要地址翻译,所以 DMA 队列一直在访问 MMU,这增加了延时并消耗了更多的功率。

[0004] 另一种配置使用有效地址到实际地址翻译器(ERAT)加上第二级 MMU。ERAT 包括用于存储以前的地址翻译的查找表。当 DMA 队列向 ERAT 发送 DMA 请求(包括有效地址)时,ERAT 查看自己的查找表以寻找有效地址匹配。如果存在匹配,则 ERAT 提供相应的实际地址,而避免了 MMU 访问。然而,如果 ERAT 没有找到与 DMA 请求的有效地址相对应的条目,则 ERAT 向 MMU 发送“未命中”,MMU 接着执行在前面第一种配置中讨论的相同步骤。

[0005] 当 MMU 识别出对应的实际页号时,MMU 将该实际页号发送到 ERAT,由 ERAT 将其加载到自己的表中。使用这种配置,由于在 ERAT 发现与有效地址之间的匹配时不需要访问 MMU,所以 ERAT 提供了电路性能改进。然而,面临的挑战是:为了使 ERAT 获得良好的“命中率”,ERAT 的查找表必须包括大量条目,这可能消耗大量的功率和物理面积。此外,面临的另一项挑战是:ERAT 仍然为每一个 DMA 请求执行查找,这也增加了延时。

[0006] 因此,需要一种用于改进的有效地址到实际地址翻译机制的系统和方法。

发明内容

[0007] 已经发现,使用用于将实际页号存储在 DMA 命令字段中,以便在 DMA 命令调出期间使用这些实际页号的系统和方法,就能解决上述挑战。DMA 队列从存储器管理单元接收实际页号并将这些实际页号存储在相应的 DMA 命令字段中用于后续的 DMA 请求。这样,DMA 队列就不那么频繁地访问 MMU 以进行地址翻译,而这通过更低的延时和功耗又提高了系统性能。

[0008] DMA 队列包括具有相应的 DMA 命令字段的 DMA 命令,所述 DMA 命令字段包括:1) 队列条目有效位,2) 有效地址字段,3) 实际地址有效位,4) 实际页号字段,5) 传输类型字段,和 6) 传输大小字段。队列条目有效位标识相应的 DMA 命令是否有效。有效地址字段包含相应的 DMA 命令的有效地址。实际地址有效位标识在实际页号字段中的实际页号是否有效。传输类型字段标识 DMA 命令的类型,例如加载或存储。另外,传输大小字段包括剩余还未传输的数据量。

[0009] DMA 队列识别准备好发出的 DMA 命令（例如，无依赖性）。在命令从 DMA 队列分派出去后，下游逻辑检查 DMA 命令的相应的实际地址有效位，以便确定是否访问 MMU 以进行地址翻译。如果 DMA 命令的实际地址有效位被置位，则包括在实际页号字段中的实际页号通过复用器被发送到总线接口单元，所述复用器被配置为从 DMA 队列接收实际页号。此外，DMA 队列将 DMA 请求（包括有效地址）和命令参数（包括传输大小）发送到调出逻辑。调出逻辑使传输大小递减，确定 DMA 队列是否可以使用现有的实际页号用于后面的调出 DMA 请求，并向 DMA 队列发送“写回信息”。接着，DMA 队列将该写回信息存储到 DMA 命令的命令字段中。这样，当 DMA 队列“调出”相同的 DMA 命令并执行另一个 DMA 请求时，下游逻辑检查实际地址有效位，如果它被置位，则从 DMA 命令的实际页号输出中提取实际页号，并将它发送到总线接口单元。结果，MMU 不被频繁地访问，这改进了总体系统性能。

[0010] 在一种实施方式中，无论实际地址有效位的值是什么，由于硬件约束，DMA 请求都被自动发送到 MMU。在该实施方式中，DMA 逻辑使用实际地址有效位加上其它参数来确定是否需要翻译请求。翻译请求或者有效地址有效请求是从 DMA 逻辑发送到 MMU 的。如果该请求被断言 (assert)，那么 MMU 寻找相应的实际页号并通过复用器将新的实际页号发送到总线接口单元，所述复用器选择该新的实际页号输出。此外，MMU 将该新的实际页号发送到 DMA 队列，在 DMA 队列中，该新的实际页号被写回到相应的实际页号字段中，用于后面的 DMA 请求。

[0011] 当 MMU 执行地址翻译时，调出逻辑还接收 DMA 请求以及命令参数。接着，调出逻辑使传输大小递减，确定 MMU 是否完成了成功的地址翻译，并发送写回信息到 DMA 队列，由 DMA 队列将其存储在 DMA 命令的相应的命令字段中，用于后面的 DMA 请求。

[0012] 以上就是对发明内容的概述，因而在必要的情况下包含了细节的简化、概括和省略；因此，本领域的技术人员将会理解：该发明内容部分只是示意性的，而绝非限制性的。在以下阐述的非限制性的详细描述中将会清楚仅由权利要求限定的本发明的其它方面、创新特征和优点。

附图说明

[0013] 参照附图，可以更好地理解本发明，使本领域的技术人员清楚它的多个目的，特征和优点。

[0014] 图 1 是示出将有效地址翻译成实际页号的存储器管理单元 (MMU) 的图；

[0015] 图 2 是示出正在执行地址翻译的有效地址到实际地址翻译器 (ERAT) 和存储器管理单元 (MMU) 的图；

[0016] 图 3 是示出从 MMU 接收实际页号并将该实际页号存储在相应的 DMA 命令字段中以在后面的 DMA 请求中使用的 DMA 队列的图；

[0017] 图 4 是示出多个 DMA 请求以及如何基于三种不同的配置进行地址翻译的表；

[0018] 图 5 是示出具有 DMA 命令以及相应的命令字段的 DMA 队列的图；

[0019] 图 6 是示出在发送出去或发出 DMA 命令时采用的高级流程图；

[0020] 图 7 是示出在调出处理，例如使 DMA 命令传输大小递减以及提供写回信息到 DMA 队列时采用的步骤的流程图；

[0021] 图 8 是示出在存储器管理单元接收翻译请求以及向总线接口单元和 DMA 队列提供

实际页号时采用的步骤的流程图；以及

[0022] 图 9 图示了一个信息处理系统，它是能够执行这里描述的运算操作的计算机系统的一个简化例子。

具体实施方式

[0023] 下面想要给出对本发明的一个例子的详细描述，但不应被拿来限制发明本身。相反，任意数量的变型可以落入本发明的范围内，这在说明书所附的权利要求书中限定。

[0024] 图 1 是示出将有效地址翻译成实际页号的存储器管理单元 (MMU) 的图。图 1 显示现有技术中用来处理 DMA 命令（例如，存储器加载和存储操作）的地址翻译配置。根据 DMA 命令的传输大小以及计算机系统一次传输的数据量，DMA 命令被“解开”。例如，如果计算机系统以 128 字节的增量传输数据，而 DMA 命令是 512 字节，那么 DMA 命令被解开到 4 个 DMA 请求中 ($128 \times 4 = 512$)。

[0025] 图 1 包括 DMA 队列 100、MMU 120 和总线接口单元 140。DMA 队列 100 识别准备好发出（例如，无依赖性）的 DMA 命令，并发送 DMA 请求 110 到 MMU 120，该请求包括对应于 DMA 命令的有效地址。MMU 120 接收 DMA 请求 110 并且访问包括有效地址和对应的实际页号的查找表。如果包括在 DMA 请求 110 中的有效地址与在查找表中存储的有效地址匹配，则 MMU 120 将相应的实际页号 130 发送到总线接口单元 140。

[0026] 另一方面，如果 MMU 120 没有找到与 DMA 请求 110 相对应的匹配的有效地址，那么 MMU 120 通知 DMA 队列 100 发生了“未命中”。MMU 120 还向存储器请求翻译信息并将来自存储器的翻译信息加载到查找表中。在加载完成后，MMU 120 告知 DMA 队列 100 重新发送先前未命中的相同的 DMA 请求。

[0027] 这样，图 1 中所示的配置对于每个 DMA 请求执行查找或存储器访问，即使该 DMA 请求是对于同一 DMA 命令的多个 DMA 请求的一部分，这样效率不高并且降低了处理器性能。

[0028] 图 2 是示出正在进行地址翻译的有效地址到实际地址翻译器 (ERAT) 和存储器管理单元 (MMU) 的图。图 2 示出了一种替代的地址翻译配置。ERAT 200 跟踪以前的地址翻译结果，因此不是对于每个 DMA 请求都访问 MMU 120。然而，正如下面所讨论的，ERAT200 仍然对于每个 DMA 请求都执行查找，这在功率和面积方面仍然效率不高，但是与图 1 中的配置相比改进了延时。

[0029] DMA 队列 100 向 ERAT 200 发送与准备好发出的 DMA 命令相对应的 DMA 请求 110，该 DMA 请求包括对应于 DMA 命令的有效地址。接着，ERAT 200 访问包含有效地址和相应的实际页号的查找表。ERAT 200 进行检查以发现查找表是否包括与包括在 DMA 请求 110 中的有效地址相关的条目。如果包括在 DMA 请求 110 中的有效地址与查找表中的存储的有效地址匹配，则 ERAT 200 将实际页号 130 发送到总线接口单元 140。DMA 队列 100、DMA 请求 110、实际页号 130 和总线接口单元 140 与图 1 中所示的相同。

[0030] 另一方面，如果 ERAT 200 没有发现与 DMA 请求 110 的有效地址相对应的匹配，则 ERAT 200 将未命中 220 发送到 MMU 120。MMU120 然后对包括在 DMA 请求 110 中的有效地址进行地址翻译。

[0031] 接着，MMU 120 访问它的包括有效地址和相应的实际页号的查找表，以发现对于 DMA 请求 110 的有效地址是否存在匹配（例如在图 1 中讨论的步骤）。如果查找表包含有

效地址,则 MMU 120 识别其对应的实际页号并将 MMU RPN 230 发送到 ERAT 200。ERAT 200 将有效地址和 MMU RPN 230 包括在其查找表中,以供后面引用。此外,DMA 请求必须被重新发送。当请求被再次重发时,ERAT 200 对有效地址进行匹配,并将 RPN 输送到总线接口单元。MMU 120 与图 1 中所示的相同。

[0032] 如果 MMU 120 在其查找表中没有找到与 DMA 请求 110 的有效地址相对应的条目,那么 MMU 120 访问存储器,以便获得正确的地址翻译(例如,在图 1 中讨论的步骤)。一旦 MMU 120 识别出与 DMA 请求 110 的有效地址相对应的实际页号,那么 MMU 120 发送 MMURPN 230 到 ERAT 200,由 ERAT 200 将其存储在自己的查找表中。DMA 请求必须被再次重发,此时 ERAT 200 对有效地址进行匹配并将 RPN 130 输送到 BIU 140。可以看出,DMA 队列 100 访问 ERAT 200 以对每个 DMA 请求进行地址翻译,即使该 DMA 请求是对于同一 DMA 命令的多个 DMA 请求的一部分。

[0033] 图 3 是示出从 MMU 接收实际页号并将该实际页号存储在相应的 DMA 命令字段中以用于后面的 DMA 请求的 DMA 队列的图。因此,图 3 所示的配置比图 1 和 2 所示的配置效率更高。

[0034] DMA 队列 300 包括具有相应的 DMA 命令字段的 DMA 命令,所述 DMA 命令字段包括:1) 队列条目有效位,2) 有效地址字段,3) 实际地址有效位,4) 实际页号字段,5) 传输类型,和 6) 传输大小字段。队列条目有效位标识相应的 DMA 命令是否有效。有效地址字段包含相应的 DMA 命令的有效地址。实际地址有效位标识包括在实际页号字段中的实际页号是否有效。传输类型标识 DMA 命令的类型,例如加载或存储。另外,传输大小字段包括剩余还未传输的数据量(参考附图 5 和相应的文字描述以了解有关 DMA 命令字段的更多细节)。

[0035] DMA 队列 300 识别准备好发出的 DMA 命令(例如,无依赖性),并且检查 DMA 命令的相应的实际地址有效位,以便确定是否访问 MMU 320 进行地址翻译。如果 DMA 命令的实际地址位被置位,那么 DMA 队列 300 从实际页号字段提取实际页号并且通过复用器 370 将实际页号 360 发送到总线接口单元 380,复用器 370 被配置为从 DMA 队列 300 接收实际页号。此外,DMA 队列 300 将 DMA 请求 308 和命令参数 325 发送到调出逻辑 330,所述命令参数 325 包括 DMA 命令的传输大小。调出逻辑 330 使传输大小递减,确定 DMA 队列 300 是否可以使用现有的实际页号用于后面的调出 DMA 请求,并发送写回信息 340 到 DMA 队列 300(参考图 7 和相应的文字描述以了解有关调出处理的更多细节)。接着,DMA 队列 300 将写回信息 340 加载到 DMA 命令的命令字段中。因此,当 DMA 队列 300 “调出”相同的 DMA 命令来执行另一个 DMA 请求时,DMA 队列 300 检查实际地址有效位,如果它被置位,则从 DMA 命令的实际页号字段提取实际页号,并将实际页号 360 发送到总线接口单元 380。结果,MMU 320 不被那么频繁地访问,这改进了整体系统性能。

[0036] 逻辑 310 控制被发送到 MMU 320 的有效地址(EA)请求有效 315,该有效地址请求有效 315 是基于几种因素而定的。首先,在实模式中,MMU 320 不执行任何翻译,因此,EA 请求有效 315 被反断言。其次,诸如“sync”和“eieio”等命令类型不需要翻译,因而 EA 请求有效 315 保持反断言。第三,在缺少最初两个条件的情况下,逻辑 310 使用 RA 有效 305 来确定 EA 请求有效 315 是否应被断言。

[0037] 当逻辑 310 识别出 DMA 命令的实际地址有效位没有被置位(上面第三个前述条件),因此需要地址翻译时,逻辑 310 激活 EA 请求有效 315。同时,DMA 队列 300 从 DMA 队列

300 向 MMU 320 发送 DMA 请求 308。接着, MMU 320 寻找相应的实际页号并通过复用器 370 将新的实际页号 350 发送到总线接口单元 380,所述复用器 370 选择新的实际页号 350 作为输入。此外, MMU 320 将新的实际页号 350 发送到 DMA 队列 300,由 DMA 队列 300 将其加载到相应的实际页号字段中用于后面的 DMA 请求。

[0038] 如果 MMU 320 在自己的查找表中没有找到与 DMA 请求 308 的有效地址相对应的条目,则 MMU 320 访问存储器以便获得正确的地址翻译。对于 DMA 命令设置 MMU 发出依赖性,使得该命令不被重发,直到 MMU 已从存储器加载了翻译。一旦 MMU 发出依赖性清零,则 DMA 命令重新发出,并且它的有效地址产生“命中”。参考图 7 和 8,了解有关更多细节的相应文字描述。

[0039] 当逻辑 310 在缺少用于确定 EA 请求有效 315 的最初两个因素(上面已讨论)的情况下识别出 RA 有效 305 被断言时,不需要任何翻译,因此逻辑 310 不断言 EA 请求有效 315。在这种情况下, DMA 300 经由复用器 370 将实际页号 360 输送到 BIU 380。

[0040] 调出逻辑 330 接收 DMA 请求 308 和命令参数 325。接着,调出逻辑 330 使传输大小递减,确定 MMU 320 是否执行了成功的地址翻译,并且发送写回信息 340 到 DMA 队列 300,由 DMA 队列 300 将该信息存储到 DMA 命令的相应的命令字段中(参考图 7 和相应的文字描述以了解有关调出处理的更多细节)。

[0041] 图 4 是示出基于三种不同配置的多个 DMA 请求以及它们的地址翻译位置的表。表 400 包括列 420 到 440,它们分别对应于在表 1-3 中示出的配置。列 420 到 440 包括用于在 DMA 命令被调出时提供地址翻译的 DMA 请求活动。在图 4 中示出的例子对应于请求传输 4KB 数据的 DMA 命令,计算机系统以 128 字节增量来传输数据。因此,为了执行 DMA 命令,需要三十二次请求(由列 410 指示)($128\text{B} \times 32 = 4\text{KB}$)。

[0042] 列 420 显示对于图 1 所示的实施方式, DMA 队列对于三十二次请求中的每次请求都访问存储器管理单元。列 430 显示对于图 2 所示的实施方式, DMA 队列对于第一次请求访问 ERAT,接下来 ERAT 访问 MMU。对于第 2 次到第 32 次请求, DMA 队列访问 ERAT,由 ERAT 从 ERAT 的查找表提供实际页号,从而不必访问 MMU。

[0043] 列 440 显示对于这里描述的发明(图 3), DMA 队列对于第一次请求访问 MMU,之后对于第 2 次到第 32 次请求,因为 DMA 队列在第一次 DMA 请求后将实际页号存储在 DMA 命令的相应的 DMA 命令字段中,所以由 DMA 队列自己将实际页号提供给总线接口单元。

[0044] 图 5 是示出具有 DMA 命令以及相应的命令字段的 DMA 队列的图。DMA 队列 300 与图 3 中所示的相同,它包括与具体的 DMA 命令相对应的 DMA 命令字段(列 510 到 560)。

[0045] 列 510 包括队列条目有效位,它标明相应的 DMA 命令是否有效。在相应的数据被完整地传输之前, DMA 命令一直保持有效。例如,对于以 128 字节增量传输数据的系统以及请求传输 128 字节以下的数据的 DMA 命令,队列条目有效位对于一次传输保持有效,然后被重置。在另一个例子中,对于以 128 字节增量传输数据的系统以及请求传输 512 字节数据的 DMA 命令,队列条目有效位保持有效,直到四次传输被成功地送到总线接口单元(128×4),然后队列条目有效位被重置。

[0046] 列 520 包括 DMA 命令有效地址,它们被翻译成实际页号并被存储在列 540 中。列 530 包括实际地址有效位,它标明包括在列 540 中的实际页号是否有效。使用上面有关请求传输 512 字节数据的 DMA 命令的例子,在第一次传输时, DMA 命令的实际页号字段没有有效

的实际页号,因而实际地址有效位未被置位。DMA 队列 300 在第一次传输时从 MMU 接收实际页号,还从包括被置位的实际地址有效位的调出逻辑接收写回信息。因此,在第二次到第四次传输时,DMA 队列 300 将存在于列 540 中的实际页号发送到总线接口单元,并且对于第二次到第四次传输不访问 MMU。

[0047] 列 550 包括 DMA 命令的相应的传输类型,例如它是加载还是存储操作。不需要由 MMU 进行翻译的其他命令,例如“sync”和“eieio”也可以从 DMA 队列发出。列 560 是传输大小字段,它包括 DMA 命令传输大小值。传输大小值对应于在相应的 DMA 命令下剩余还未传输的数据量。使用以上请求传输 512 字节数据的 DMA 命令的例子,在第一次传输前,传输大小值是 512 字节。在第二次传输前,传输大小值是 384 字节 (512B-128B),这是从调出逻辑接收的递减后的传输大小(参考图 7 和相应的文字描述以了解有关递减的传输大小的更多细节)。在第三次传输前,传输大小值是 256 字节 (384B-128B)。而在第四次传输前,传输大小值是 128 字节。

[0048] DMA 队列 300 包括对应于 DMA 命令的行 570 到 595。行 570 和 595 显示相应的 DMA 命令是有效的(列 510)并且在列 540 中包括有效的实际地址(列 530)。行 580 显示相应的 DMA 命令是有效的(列 510)但在列 540 中不包括有效的实际地址(列 530)。行 590 显示相应的 DMA 命令不是有效的(列 510)并且在列 540 中不包括有效的实际地址(列 530)。

[0049] 图 6 是示出在 DMA 队列调出 DMA 命令时采用的步骤的高级流程图。DMA 队列使用包括在 DMA 命令字段中的信息来确定是自身向总线接口单元发送实际页号还是请求存储器管理单元(MMU)执行地址翻译(参考图 5 和相应的文字描述以了解有关 DMA 命令参数的更多细节)。

[0050] 处理开始于 600,此后在步骤 605,DMA 队列识别准备好发出的 DMA 命令(例如,无依赖性)。确定 DMA 命令的相应的实际地址有效位是否被置位,这表明 DMA 命令的相应的实际页号是否有效(判决 610)。

[0051] 假设系统不在实模式下运行并且命令类型是需要地址翻译的类型,如果 DMA 命令的相应的实际地址有效位被置位,则判决 610 前往“是”分支 612,之后处理在步骤 650 发送命令参数到调出处理。命令参数包括实际地址有效位值和传输大小,其中传输大小对应于 DMA 命令希望加载或存储的数据量,例如 4KB 的数据。调出处理接收命令参数并且相应地修改命令参数,例如使传输大小递减传输大小允许量(参考图 7 和相应的文字描述以了解有关调出处理步骤的更多细节)。

[0052] 在步骤 660,DMA 队列从 DMA 命令字段之一提取 DMA 命令的实际页号并且将该实际页号发送到总线接口单元 380。总线接口单元 380 与图 3 所示的相同,实际页号在到达总线接口单元 380 之前可以通过复用器,例如图 3 所示的复用器 370。

[0053] 在步骤 670,DMA 队列从调出处理接收写回信息。写回信息包括如上所述的实际地址有效位和递减的传输大小。在步骤 680,DMA 队列将写回信息加载到 DMA 命令的相应的 DMA 命令字段用于后面的 DMA 请求。

[0054] 另一方面,如果 DMA 队列识别出 DMA 命令的实际地址有效位没有被置位,那么判决 610 前往“否”分支 618,之后 DMA 队列在步骤 620 将命令参数发送到调出处理。在步骤 625,DMA 队列向执行地址翻译的存储器管理单元(MMU)发送翻译请求。MMU,例如图 3 中所示的 MMU 320,执行地址翻译并将实际页号发送到总线接口单元 380。MMU 还将实际页号提

供给 DMA 队列, DMA 队列处理在步骤 630 接收该实际页号。

[0055] 在一种实施方式中, 如果 MMU 在它的表中没有找到对应于有效地址的表条目, 那么 MMU 发送“未命中”到调出处理, 并且从存储器取得翻译。在加载了翻译后, MMU 处理发送“清除依赖性”到 DMA 队列, 以通知 DMA 队列重发相应的命令 (参考图 8 和相应的文字描述以了解有关 MMU 处理的更多细节)。

[0056] 在步骤 640, DMA 队列从调出处理接收写回信息, 并且在步骤 645, DMA 队列将写回信息和实际页号加载到 DMA 命令字段中。

[0057] 判断 DMA 命令是否完成 (判决 690)。如果 DMA 命令需要进一步的调出 (例如, 队列条目有效位被置位), 则判决 690 前往“是”分支 692, 它循环回去继续处理 DMA 命令。该循环一直继续, 直到 DMA 命令被调出, 此时判决 690 前往“否”分支 698, 之后 DMA 命令处理在 699 结束。

[0058] 图 7 是示出在调出处理, 例如使 DMA 命令传输大小递减以及提供写回信息到 DMA 队列时采用的步骤的流程图。调出处理开始于 700, 之后调出处理在步骤 705 从 DMA 队列处理接收命令参数 (参考图 6 和相应的文字描述以了解有关 DMA 队列处理的更多细节)。命令参数包括实际地址有效位和传输大小。

[0059] 通过检查实际地址有效位的状态做出有关实际页号是否有效的判断 (判决 710)。当实际地址有效位被置位时, DMA 队列提供实际页号到总线接口单元。当实际页号没有被置位时, DMA 队列请求存储器管理单元 (MMU) 执行地址翻译并且提供实际页号到总线接口单元和 DMA 队列。

[0060] 如果实际页号有效, 则判决 710 前往“是”分支 718, 之后处理绕开存储器管理单元请求步骤。另一方面, 如果实际页号不是有效的, 则判决 710 前往“否”分支 712, 之后处理在步骤 715 从 MMU 接收命中或未命中信号, 表明 MMU 是否找到了与 DMA 请求的有效地址相对应的实际页号 (参考图 8 和相应的文字描述以了解有关 MMU 处理的更多细节)。

[0061] 判断 MMU 是否执行了地址翻译 (判决 720)。如果 MMU 未曾执行地址翻译, 则判决 720 前往“否”分支 722, 之后处理对 MMU 发出依赖性位进行置位, 该位信息告知 DMA 队列特定的指令依赖于 MMU 执行地址翻译 (步骤 725)。在稍后 MMU 已从存储器收到翻译时, MMU 发送信号到 DMA 逻辑以清除 MMU 发出依赖性。直到从 MMU 接收到 MMU 未命中清零信号, DMA 命令才被重发 (参考图 8 和相应的文字描述以了解更多的细节)。

[0062] 另一方面, 如果 MMU 执行了地址翻译, 则判决 720 前往“是”分支 728, 之后处理在步骤 730 对实际地址有效位进行置位, 该实际地址有效位被包括在调出处理发送到 DMA 队列处理的写回信息中 (参考下文)。

[0063] 在步骤 735, 处理通过从它在命令参数中接收到的传输大小减去允许的最大传输大小而使 DMA 命令的传输大小递减 (参考上面的 705)。例如, 如果接收到的传输大小是 384 字节, 允许的最大传输大小是 128 字节, 则递减后的传输大小是 256 字节。通过分析递减后的传输大小判断是否还需要其它 DMA 请求 (判决 740)。即, 如果递减后的传输大小是零, 则不再需要其它 DMA 请求。然而, 如果传输大小不是零, 则还需要其它 DMA 请求以便完成 DMA 命令。

[0064] 如果不需要其它 DMA 请求, 则判决 740 前往“否”分支 742, 之后处理复位队列有效位, 它表明相应的 DMA 命令已被完成, 因此不再有效 (步骤 745)。另一方面, 如果还需要其

它 DMA 请求,则判决 740 前往“是”分支 748,之后处理在步骤 750 使有效地址递增到下一地址。

[0065] 判断 DMA 命令是否到达了页的末端(判决 760)。例如,存储器配置可以具有 4KB 增量的页。如果处理已到达页的末端,则判决 760 前往“是”分支 768,之后处理在步骤 770 复位实际地址有效位,该位信息通知 DMA 队列请求 MMU 对于 DMA 命令的下一个翻译执行地址翻译。另一方面,如果处理还没有到达页的末端,则判决 760 前往“否”分支 762,绕过实际地址有效位复位步骤。

[0066] 在步骤 780,调出处理发送写回信息到 DMA 队列处理。写回信息包括最近的变化,例如队列有效位、实际地址有效位和递减的传输大小。通过检查队列条目有效位来判断 DMA 命令是否需要进一步的 DMA 请求(判决 790)。如果队列条目有效位被置位,则判决 790 前往“是”分支 792,之后处理循环回去,处理对于 DMA 命令的更多的 DMA 请求。该循环一直继续,直到 DMA 命令完成,此时判决 790 前往“否”分支 798,之后调出处理在 799 结束。

[0067] 图 8 是示出在存储器管理单元接收翻译请求以及向总线接口单元和 DMA 队列提供实际页号时采用的步骤的流程图。使用这里描述的实施方式,当 DMA 命令在它的 DMA 命令字段之一中不包括有效实际页号,传输类型是需要翻译的类型,并且系统正运行在虚模式下(即,实模式不活动)时,DMA 队列访问存储器管理单元(MMU)。

[0068] MMU 处理开始于 800,之后 MMU 在步骤 810 从 DMA 队列接收 DMA 请求。在步骤 820,处理从 DMA 请求提取有效地址。MMU 在步骤 830 访问 MMU 表 840,以找到与该有效地址相对应的实际页号。MMU 表 840 包括一个有效地址和相应的实际页号的列表。

[0069] 判断 MMU 表 840 是否包括与有效地址相对应的表条目(判决 850)。如果 MMU 找到了表条目,则判决 850 前往“是”分支 852,之后处理在步骤 855 发送命中信号到调出处理(参考图 7 和相应的文字描述以了解有关调出处理的更多细节)。在步骤 860,MMU 发送找到的实际页号到总线接口单元 380 和 DMA 队列处理,并且 MMU 处理在 890 结束。DMA 队列处理将实际页号与 DMA 命令存储在一起,以便将实际页号用于后面的 DMA 请求。总线接口单元 380 与图 3 所示的相同。

[0070] 另一方面,如果 MMU 在 MMU 表 840 中没有找到与有效地址相对应的表条目,则判决 850 前往“否”分支 858,之后处理发送“未命中”到调出处理(步骤 870)并从存储库 875 取得翻译,该翻译被加载到 MMU 表 840 中。在加载了翻译后,MMU 处理在步骤 885 发送“清除依赖性”到 DMA 队列,MMU 处理在 890 结束。存储库 875 可以被存储在非易失存储区上,例如计算机硬盘。

[0071] 图 9 图示了一个信息处理系统,它是能够执行这里描述的运算操作的计算机系统的一个简化例子。宽带处理器体系结构(BPA)900 包括在单个芯片上的多个异构处理器(片上系统)、公共存储器和公共总线。异构处理器是具有不同指令集的处理器,它们共享公共存储器和公共总线。例如,异构处理器之一可以是使用 RISC 指令集的 PowerPC,另一个异构处理器可以是使用向量指令集的协同处理单元,二个处理器共享同一存储空间。

[0072] BPA 900 包括协同处理单元(SPC)910、930 和 950。每个 SPC 包括被耦合到 DMA 控制器(DMAC)的处理器核,而 DMAC 被耦合到总线接口单元(BIU)。SPC 910 包括处理器核 915、DMAC 920 和 BIU 925。SPC 930 包括处理器核 935、DMAC 940 和 BIU 945。而 SPC950 包括处理器核 955、DMAC 960 和 BIU 965。

[0073] BPA 900 还包括处理单元 975, 处理单元 975 运行操作系统 (OS) 978。例如, 处理单元 975 可以是被嵌入在 BPA 900 中的 Power PC 核, OS 978 可以是 Linux 操作系统。处理单元 975 管理用于 BPA 900 的公共存储器映射表。该存储器映射表对应于包括在 BPA 900 中的存储器位置, 例如 L2 存储器 980 以及包括在 SPC 910、930 和 950 中的非私有存储器。

[0074] 每个 SPC 被用于处理数据信息, 并且每个 SPC 可以具有不同的指令集。例如, BPA 900 可以用于无线通信系统中, 每个 SPC 可以负责单独的处理任务, 例如调制、芯片速率处理、编码和网络接口。在另一个例子中, 每个 SPC 可以具有相同的指令集并且可以被并行地用于执行受益于并行处理的操作。每个 SPC 都包括协同处理单元 (SPU)。SPU 优选地是单指令、多数据 (SIMD) 处理器, 例如数字信号处理器、微控制器或者这些核的组合。在一个优选实施方式中, 每个 SPU 包括局部存储器、寄存器、四个浮点单元和四个整数单元。然而, 根据所需的处理能力, 可以采用更多或更少的浮点单元和整数单元。

[0075] SPC 910、930 和 950 被连接到处理器元件总线 970, 该总线在 SPC、处理单元 975、L2 存储器 980、存储器控制器 985 和 I/O 控制器 990 之间传递信息。存储器控制器 985 与用于存取数据的存储器件 988 通信, 而 I/O 控制器 990 与属于外围设备的 I/O 设备 995 通信。

[0076] 虽然图 9 中所示的计算机系统能够执行这里描述的处理, 但是该计算机系统只是计算机系统的一个例子。本领域的技术人员将认识到, 很多其它计算机系统也能够执行这里描述的处理。

[0077] 本发明的优选实施方案之一是客户端应用, 即在例如可以驻留在计算机的随机访问存储器中的代码模块中的一组指令 (程序代码)。在被计算机需求前, 该组指令可以被存储在另一个计算机存储器中, 例如存储在硬盘驱动器中, 或者存储在可移动存储器例如光盘 (最终用于 CD ROM 中) 或软盘 (最终用于软盘驱动器中) 中, 或者可以通过因特网或其它计算机网络下载。因此, 本发明可以被实施为用于计算机中的计算机程序产品。此外, 虽然所描述的多种方法被方便地实施在通过软件选择性地激活或重新配置的通用计算机中, 但是本领域的技术人员也会认识到这样的方法可以用硬件、固件或者被构建来执行所需的方法步骤的更专用的装置来实现。

[0078] 虽然已经显示和描述了本发明的具体实施方式, 但本领域的技术人员很清楚, 根据这里的教导可以做出多种改变和变形, 而不会偏离本发明及其更宽广的方面。因此, 所附权利要求在它们的范围内包括落入本发明的真实精神和范围内的所有这样的改变和变形。此外, 可以理解本发明仅由所附的权利要求来限定。本领域的技术人员将会明白, 如果想要保护特定数量的引入的权利要求特征, 这样的意图将被直接记载在该权利要求中, 在缺少这样的记载的情况下, 则不存在这样的限制。举一个非限制性的例子, 为帮助理解, 所附权利要求中包含介绍性的短语“至少一个”和“一个或多个”的使用, 以引入权利要求的技术特征。然而, 这样的短语的使用不应被理解为意味着前面加上不定冠词的技术特征要将包含这样的权利要求特征的任何具体权利要求限制为仅包含一个这样的技术特征的发明, 即使当同一权利要求包括介绍性的短语“一个或多个”或“至少一个”和不定冠词时; 对于使用定冠词的权利要求同样如此。

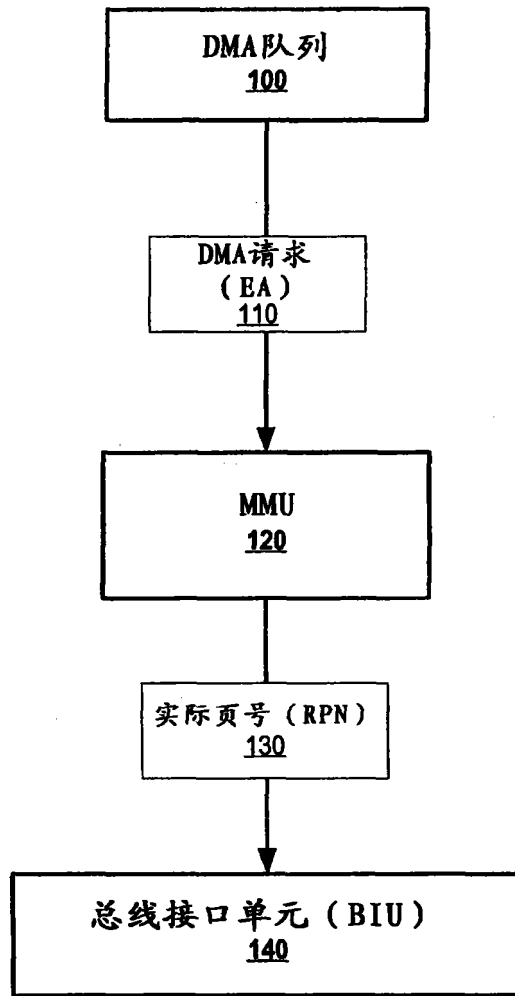


图 1 (现有技术)

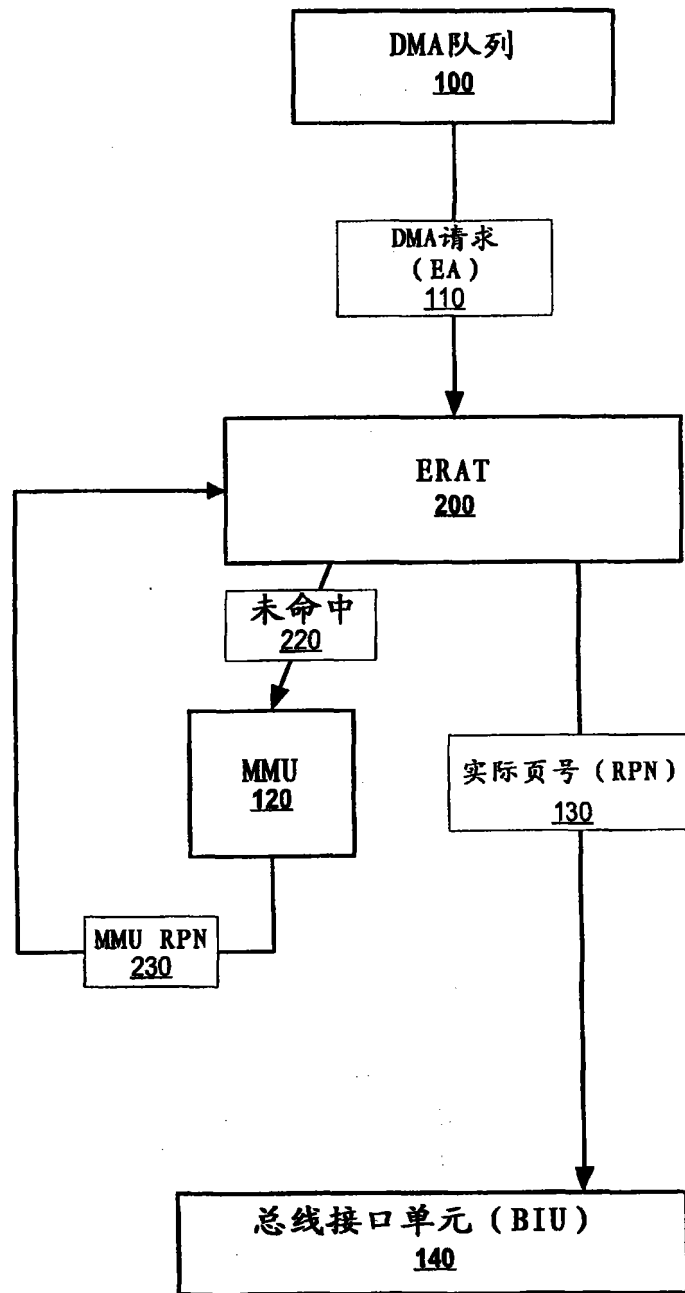


图 2(现有技术)

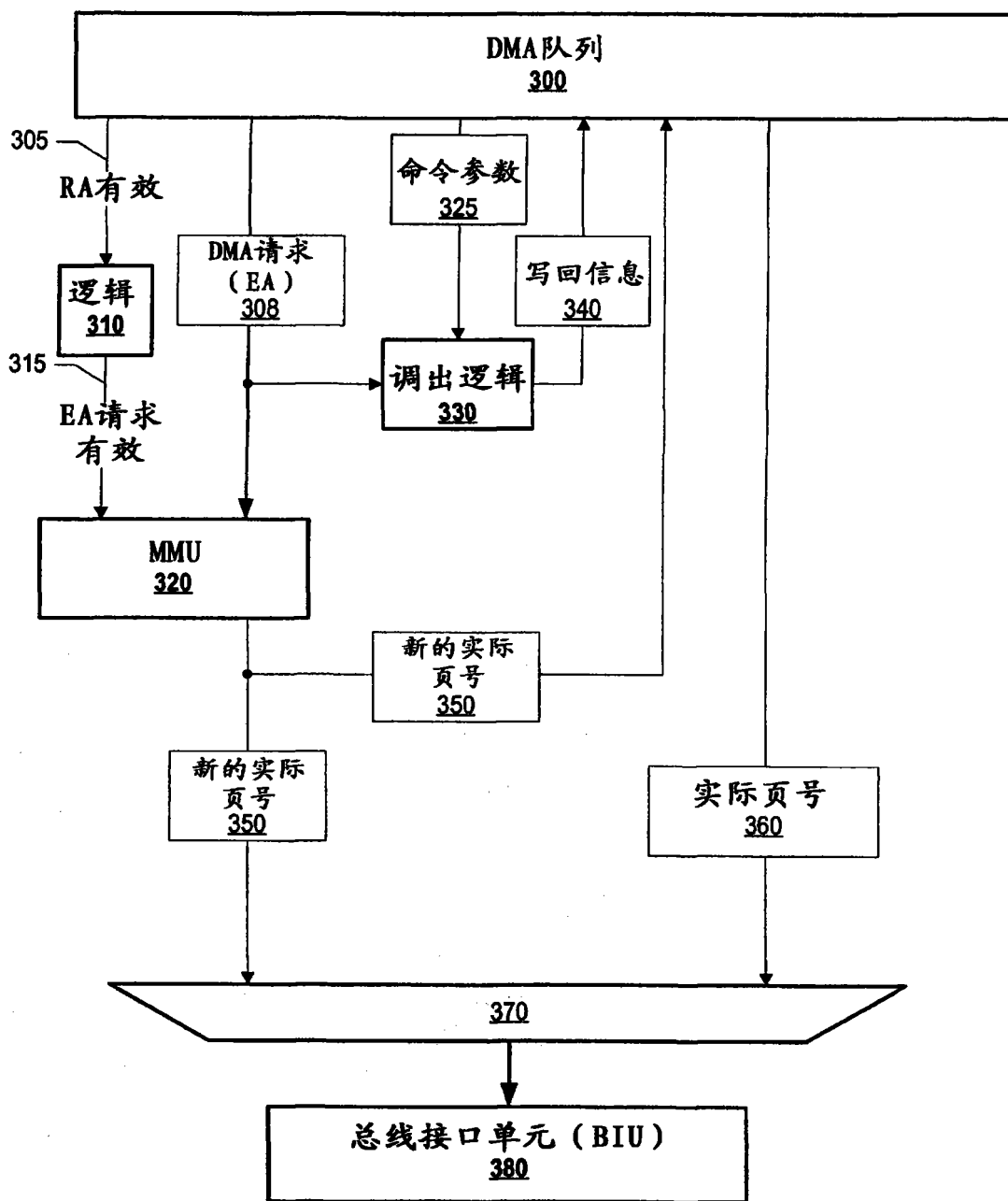


图 3

400

请求号	只有MMU	ERAT和MMU	改进的翻译机制
1	访问MMU	访问ERAT和MMU	访问MMU
2	访问MMU	访问ERAT	DMAQ
3	访问MMU	访问ERAT	DMAQ
4	访问MMU	访问ERAT	DMAQ
5	访问MMU	访问ERAT	DMAQ
6	访问MMU	访问ERAT	DMAQ
7	访问MMU	访问ERAT	DMAQ
8	访问MMU	访问ERAT	DMAQ
9	访问MMU	访问ERAT	DMAQ
10	访问MMU	访问ERAT	DMAQ
11	访问MMU	访问ERAT	DMAQ
12	访问MMU	访问ERAT	DMAQ
13	访问MMU	访问ERAT	DMAQ
14	访问MMU	访问ERAT	DMAQ
15	访问MMU	访问ERAT	DMAQ
16	访问MMU	访问ERAT	DMAQ
17	访问MMU	访问ERAT	DMAQ
18	访问MMU	访问ERAT	DMAQ
19	访问MMU	访问ERAT	DMAQ
20	访问MMU	访问ERAT	DMAQ
21	访问MMU	访问ERAT	DMAQ
22	访问MMU	访问ERAT	DMAQ
23	访问MMU	访问ERAT	DMAQ
24	访问MMU	访问ERAT	DMAQ
25	访问MMU	访问ERAT	DMAQ
26	访问MMU	访问ERAT	DMAQ
27	访问MMU	访问ERAT	DMAQ
28	访问MMU	访问ERAT	DMAQ
29	访问MMU	访问ERAT	DMAQ
30	访问MMU	访问ERAT	DMAQ
31	访问MMU	访问ERAT	DMAQ
32	访问MMU	访问ERAT	DMAQ

图 4

DMA队列 300					
510	520	530	540	550	560
队列 条目有效	有效地址	实际 地址有效	实际页号	传输类型	传输大小
570	1	EA 1	1	RPN 1	4KB
580	1	EA 2	0	-	256B
590	0	EA 3	0	-	32B
	·	·	·	·	·
	·	·	·	·	·
595	1	EA N	1	RPN N	128B

图 5

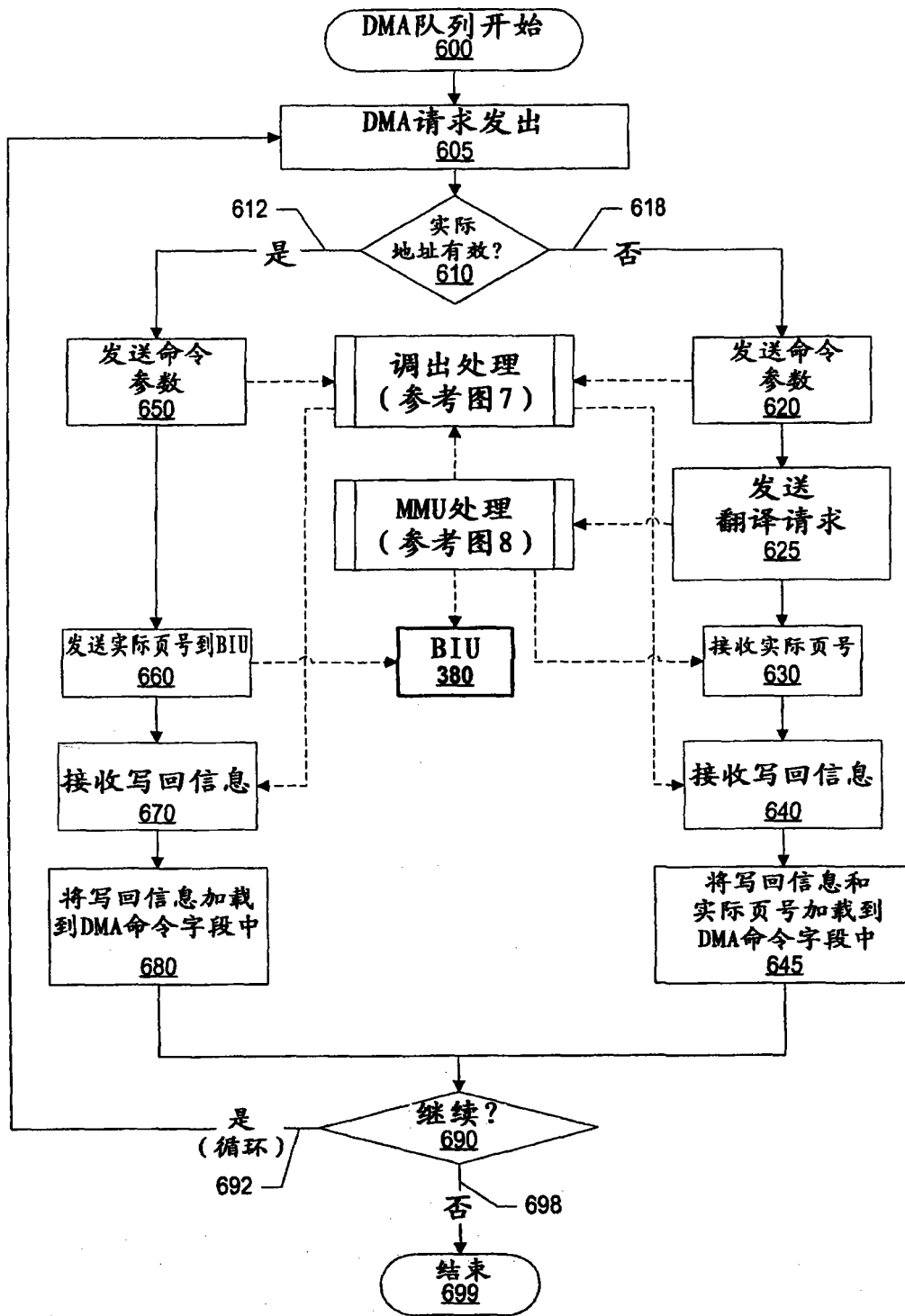


图 6

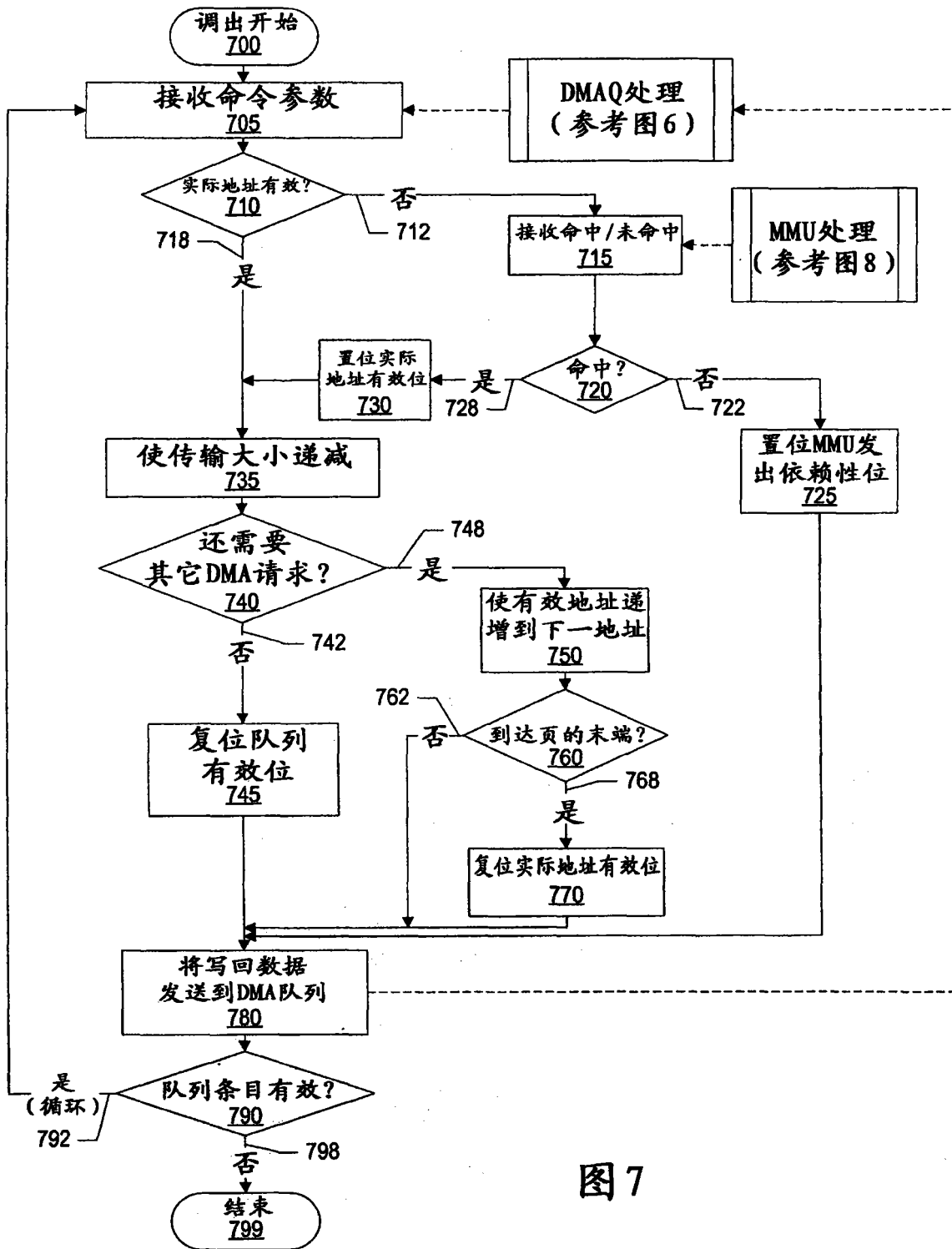


图 7

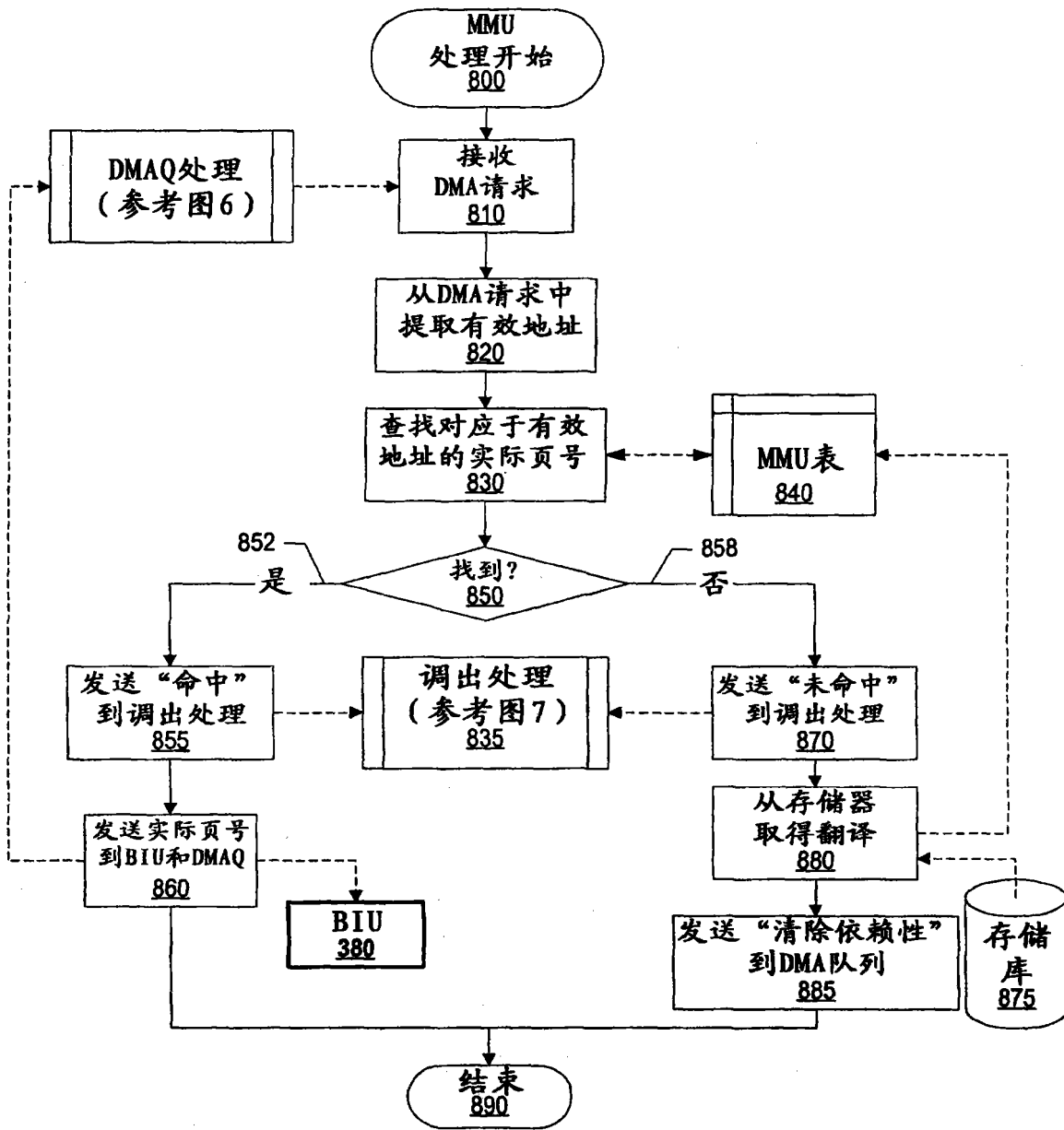


图 8

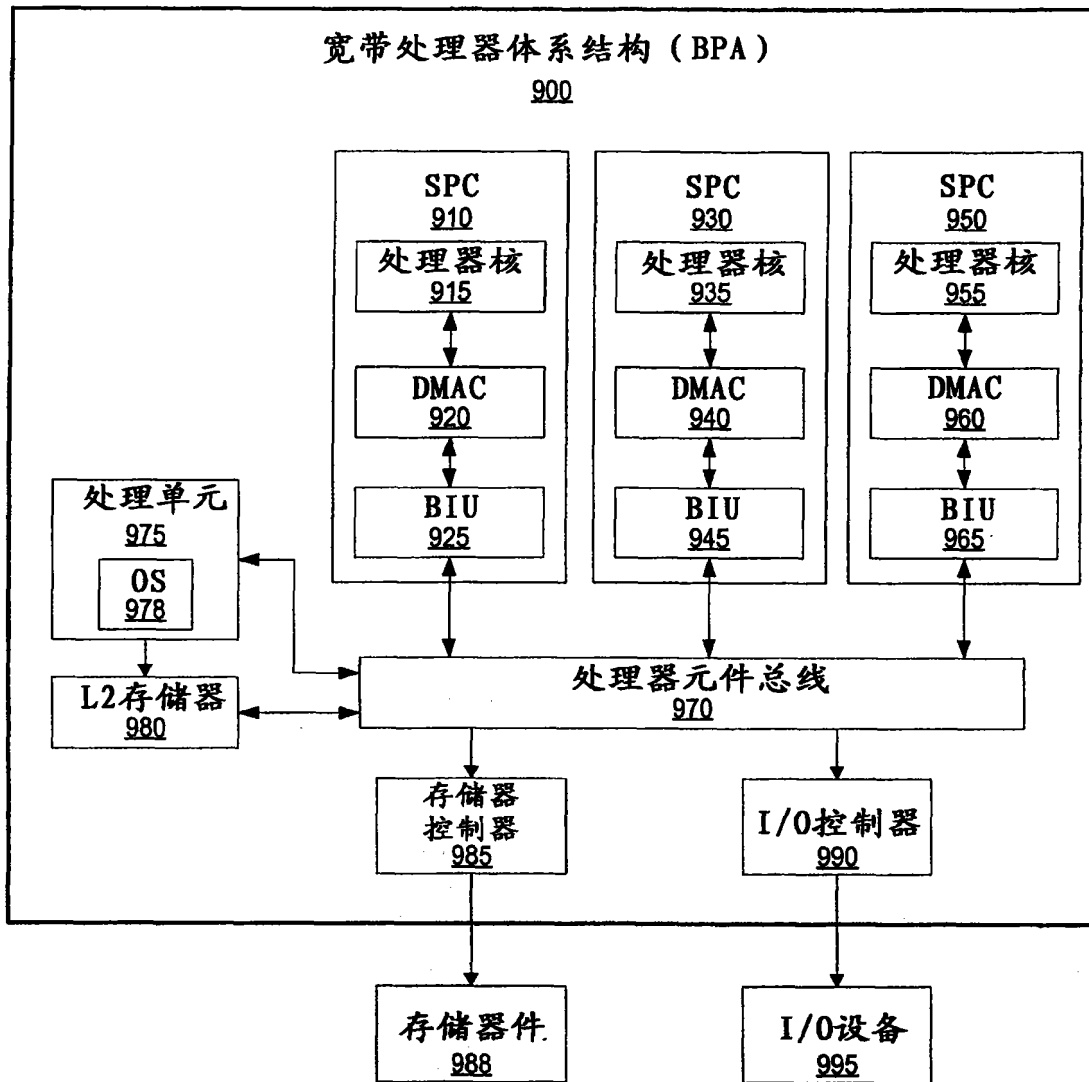


图 9