SooPAT

多维DMA传输装置与方法

申请号：201110449966.X
申请日：2011-12-29

申请(专利权)人 中国科学院自动化研究所
地址 100190 北京市海淀区中关村东路95号
发明(设计)人 王东琳 刘子君 张星 谢少林

主分类号 G06F13/28(2006.01)I
分类号 G06F13/28(2006.01)I
公开(公告)号 102567258A
公开(公告)日 2012-07-11
专利代理机构 中科专利商标代理有限责任公司 11021
代理人 周国城

www.soopat.com

注：本页蓝色字体部分可点击查询相关专利

(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 102567258 B
(45)授权公告日 2014.08.27

(21)申请号 201110449966.X

(22)申请日 2011.12.29

(73)专利权人 中国科学院自动化研究所
地址 100190 北京市海淀区中关村东路 95 号

(72)发明人 王东琳 刘子君 张星 谢少林

(74)专利代理机构 中科专利商标代理有限责任公司 11021
代理人 周国城

(51)Int.Cl.
G06F 13/28(2006.01)

(56)对比文件
CN 101196860 A,2008.06.11, 全文.
CN 101425044 A,2009.05.06, 全文.
CN 101777031 A,2010.07.14, 全文.
CN 101853229 A,2010.10.06, 全文.
JP 2009271610 A,2009.11.19, 全文.
US 2009172318 A1,2009.07.02, 全文.

审查员 赵洋

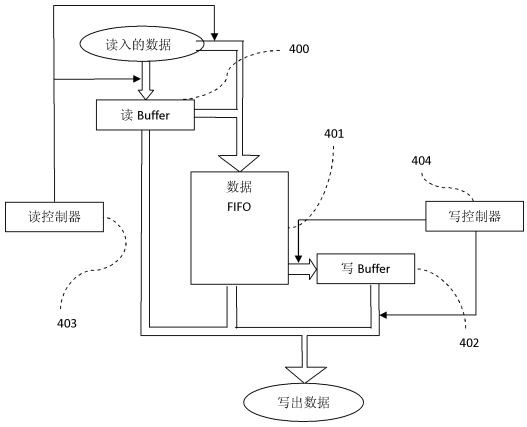
权利要求书2页 说明书5页 附图6页

(54)发明名称

多维 DMA 传输装置与方法

(57)摘要

本发明公开了一种多维 DMA 传输装置与方法，所述装置包括读 Buffer(400)、数据 FIFO(401)、写 Buffer(402)、读控制器(403)和写控制器(404)。所述读 Buffer(400)用于暂存在读操作过程中新读入的数据；所述数据 FIFO(401)用于存储暂时积压下来的数据；所述写 Buffer(402)用于暂存在写操作过程中将要写出的数据；所述读控制器(403)用于控制读操作过程，发送读请求并对该请求的回复进行处理；所述写控制器(404)用于控制写操作过程，发送写请求并对该请求的回复进行处理。本发明适合于矩阵分割、矩阵重组、矩阵扩展等多种 DSP 基本操作，将简单的处理过程与复杂处理过程相分离后并行执行，减少了核处理时间。



CN 102567258 B

1. 一种多维 DMA 传输装置，包括读 Buffer(400)、数据 FIFO(401)、写 Buffer(402)、读控制器 (403) 和写控制器 (404)，其特征在于：

所述读 Buffer(400) 与数据 FIFO(401) 相连，用于暂存在读操作过程中新读入的数据；

所述数据 FIFO(401) 用于暂时存储读写操作中积压的数据；

所述写 Buffer(402) 与数据 FIFO(401) 相连，用于暂存在写操作过程中将要写出的数据；

所述读控制器 (403) 连接于读 Buffer(400) 和数据 FIFO(401)，用于控制读操作过程，发送读请求并对该请求的回复进行处理；

所述写控制器 (404) 连接于写 Buffer(402)、数据 FIFO(401) 和读 Buffer(400)，用于控制写操作过程，发送写请求并对该请求的回复进行处理；

当写 Buffer(402) 和数据 FIFO(401) 中存储的数据之和足够当次写操作所需数据时，所述写控制器 (404) 用于向外部发送写操作请求，并将写 Buffer(402) 和数据 FIFO(401) 中的数据依次写出；

当数据 FIFO(401) 为空并且已经到达 DMA 传输末尾时，所述写控制器 (404) 用于从写 Buffer(402) 和读 Buffer(400) 中选取数据进行写出。

2. 如权利要求 1 所述的多维 DMA 传输装置，其特征在于：

所述读 Buffer(400) 包括读状态寄存器，该读状态寄存器用于存储标志该读 Buffer(400) 中有效数据的情况的状态信息。

3. 如权利要求 1 所述的多维 DMA 传输装置，其特征在于：

所述写 Buffer(402) 包括写状态寄存器，该写状态寄存器用于存储标志该写 Buffer(402) 中有效数据的情况的状态信息。

4. 如权利要求 1 所述的多维 DMA 传输装置，其特征在于：

所述读控制器 (403) 用于将读取的数据暂存于该读 Buffer(400) 和数据 FIFO(401) 中，并修改读 Buffer(400) 中的读状态寄存器的状态信息。

5. 一种多维 DMA 传输方法，用于 DMA 传输装置中，该 DMA 传输装置包括读 Buffer(400)、数据 FIFO(401) 和写 Buffer(402)，其特征在于，所述方法包括读操作步骤和写操作步骤：

在读操作步骤中，发送读请求并对该请求的回复进行处理，并修改该读 Buffer(400) 和数据 FIFO(401) 内的数据及状态信息，使 DMA 存入数据 FIFO(401) 前进行了对齐操作而形成一个连续的数据块，当读 Buffer(400) 中空位足够当次读入操作时，将读入数据进行循环移位后接到读 Buffer(400) 中最后一个有效数据之后，并且，当读到的数据数量和读 Buffer(400) 中的数据数量之和不小于端口位宽 W 时，将 W 个数据排好顺序放入数据 FIFO(401) 中，并剩余数据放入读 Buffer(400) 中；

在写操作步骤中，发送写请求并对该请求的回复进行处理，修改读 Buffer(400)、数据 FIFO(401) 和写 Buffer(402) 内的数据及状态信息，以便从连续的数据块中截取数据并写出。

6. 如权利要求 5 所述的多维 DMA 传输方法，其特征在于：

在写操作步骤中，当需要写出的数据数量不大于写 Buffer(402) 中有效数据的数量时，直接从写 Buffer(402) 读取数据；或者

当需要写出的数据数量大于写 Buffer(402) 中的有效数据量时，并且数据 FIFO(401) 不空时，数据 FIFO(401) 弹出 W 个数据，将写 Buffer(402) 中的数据和该 W 个数据拼接后截取需要写出的数据长度的数据并写出，并将剩余数据放入写 Buffer(402) 中。

7. 如权利要求 5 所述的多维 DMA 传输方法，其特征在于：

在写操作步骤中，当到达 DMA 传输结尾时，将写 Buffer(402) 中的数据和读 Buffer(400) 中的数据进行组合，并使写 Buffer(402) 中数据在前，选取需要读取的数据并移位写出，将剩下数据存入写 Buffer(402) 中。

多维 DMA 传输装置与方法

技术领域

[0001] 本发明主要涉及多维DMA(Direct Memory Access)的传输装置与方法,与片内数据存储结构和片外数据存储结构紧密相关。

背景技术

[0002] 随着SIMD(Single Instruction Multiple Data)的出现以及向量处理器的不断发展,DSP对数据供给的要求越来越高。DSP算法的不断发展使得复杂的数据结构已经成为DSP算法的结构,也对DMA传输模式提出了更高的要求。

[0003] 普通的DMA传输方法只是将连续的数据搬运到一块连续的地址段内,缺乏必要的地址变化,为之后的Load/Store操作(读数据操作/写数据操作)带来了不必要的麻烦。

发明内容

[0004] (一)要解决的技术问题

[0005] 本发明所要解决的技术问题是克服普通DMA传输数据分布简单、对多维数据支持不足、冗余读写操作较多的缺点,提出一种新的多维DMA传输方法及装置。

[0006] (二)技术方案

[0007] 本发明提出的多维DMA传输装置包括读Buffer、数据FIFO、写Buffer、读控制器和写控制器,所述读Buffer与数据FIFO相连,用于暂存在读操作过程中新读入的数据;所述数据FIFO用于暂时存储读写操作中积压的数据;所述写Buffer与数据FIFO相连,用于暂存在写操作过程中将要写出的数据;所述读控制器连接于读Buffer和数据FIFO,用于控制读操作过程,发送读请求并对该请求的回复进行处理;所述写控制器连接于写Buffer、数据FIFO和读Buffer,用于控制写操作过程,发送写请求并对该请求的回复进行处理。

[0008] 所述读Buffer包括读状态寄存器,该读状态寄存器用于存储标志该读Buffer中有效数据的情况的状态信息。

[0009] 所述写Buffer包括写状态寄存器,该写状态寄存器用于存储标志该写Buffer中有效数据的情况的状态信息。

[0010] 所述读控制器用于将读取的数据暂存于该读Buffer和数据FIFO中,并修改读Buffer中的读状态寄存器的状态信息。

[0011] 当写Buffer和数据FIFO中存储的数据之和足够当次写操作所需数据时,所述写控制器用于向外部发送写操作请求,并将写Buffer和数据FIFO中的数据依次写出。

[0012] 当数据FIFO为空并且已经到达DMA传输末尾时,所述写控制器用于从写Buffer和读Buffer中选取数据进行写出。

[0013] 本发明还提出一种多维DMA传输方法,用于DMA传输装置中,该DMA传输装置包括读Buffer、数据FIFO和写Buffer,所述方法包括读操作步骤和写操作步骤:在读操作步骤中,发送读请求并对该请求的回复进行处理,并修改该读Buffer和数据FIFO内的数据及状态信息,使DMA存入数据FIFO前进行了对齐操作而形成一个连续的数据块;在写操作步骤

中，发送写请求并对该请求的回复进行处理，修改读 Buffer、数据 FIFO 和写 Buffer 内的数据及状态信息，以便从连续的数据块中截取数据并写出。

[0014] 在读操作步骤中，当读 Buffer 中空位足够当次读入操作时，将读入数据进行循环移位后接到读 Buffer 中最后一个有效数据之后；或者，

[0015] 当读到的数据数量和读 Buffer 中的数据数量之和不小于端口位宽 W 时，将 W 个数据排好顺序放入数据 FIFO 中，并剩余数据放入读 Buffer 中。

[0016] 在写操作步骤中，当需要写出的数据数量不大于写 Buffer 中有效数据的数量时，直接从写 Buffer 读取数据；并且

[0017] 当需要写出的数据数量大于写 Buffer 中的有效数据量时，并且数据 FIFO 不空时，数据 FIFO 弹出 W 个数据，将写 Buffer 中的数据和该 W 个数据拼接后截取需要写出的数据长度的数据并写出，并将剩余数据放入写 Buffer 中。

[0018] 在写操作步骤中，当到达 DMA 传输结尾时，将写 Buffer 中的数据和读 Buffer 中的数据进行组合，并使写 Buffer 中数据在前，选取需要读取的数据并移位写出，将剩下数据存入写 Buffer 中。

[0019] （三）有益效果

[0020] 根据本发明的多维 DMA 传输装置与方法，由于多维数据当读入时在缓存区内自动对齐使数据形成一个连续的数据块，实现了多维数据的无缝连接。而写出操作从连续的数据块中截取数据并进行合适的位移，保证写出操作按多维 DMA 要求进行操作。

附图说明

[0021] 图 1 显示了 3 维 DMA 传输的数据分布结构。

[0022] 图 2 显示了多维 DMA 传输的读操作行为级描述。

[0023] 图 3 显示了多维 DMA 传输的写操作行为级细节。

[0024] 图 4 显示了整个多维 DMA 传输控制器 (CSU) 的整体结构。

[0025] 图 5 显示了当读 Buffer 中空位足够当次读入操作时的存储区变化情况的一个示例。

[0026] 图 6 显示了当读 Buffer 中空位不够当次读入操作时的存储区变化情况的一个示例。

[0027] 图 7 显示了当写 Buffer 中数据足够当次写出操作时的存储区变化情况的一个示例。

[0028] 图 8 显示了当写 Buffer 中数据不够当次写出操作时的存储区变化情况的一个示例。

[0029] 图 9 显示了 DMA 传输结尾时存储区可能的变化情况的一个示例。

具体实施方式

[0030] 为使本发明的目的、技术方案和优点更加清楚明白，以下结合具体实施例，并参照附图，对本发明进一步详细说明。

[0031] 为了克服普通 DMA 传输数据分布简单、对多维数据支持不足、冗余读写操作较多的缺点，本发明提出一种多维 DMA 传输装置与方法，所述多维 DMA 传输装置与方法可以通过

将多块数据拼接成一块数据、一块拆分成多块、或者多块数据重新划分等操作，为接下来的Load/Store 操作提供一定的便利。如对于一个大矩阵，本发明可以通过多维 DMA 传输的方式将它拆成多个小矩阵供给 DSP 进行处理。

[0032] 本发明的多维 DMA 传输可以通过配置多个参数进行简单的数据转换，一般一个多维 DMA 传输包含多个参数，除了第一维只存在一个 Num 参数外，其他各维都包含两个参数：Num 和 Step。Num 表示包含多少个数据组，Step 表示该维下每个数据组之间的距离。

[0033] 图 1 表示三维 DMA 传输的数据分布结构，如图所示，其中第一维有 D1Num 个数据，第二维有 D2Num 个一维数据组，且一维数据组之间的距离为 D2Step，第三维有 D3Num 个二维数据组，且二维数据组之间的距离为 D3Step。

[0034] 本发明的多维 DMA 传输具有比较灵活的特性，既支持读写同维数下 DMA 传输的处理，也支持读写不同维数情况下的 DMA 传输的处理，适用于各种结构的 DMA 数据直接搬运和简单处理，大大增强了 DMA 单元的功能。

[0035] 如图 2 和图 3 所示，图 2 表示了读过程中地址和数据参数的处理的行为级描述示例，图 3 表示了写过程中地址和数据参数的处理的行为级细节示例，其中均通过嵌套的 For 语句循环方式表示了 DMA 传输的具体过程。

[0036] 读控制器按照图 2 所示方式读取数据后放入控制器中的数据缓存中，当写控制器发现数据缓存中的数据时则进行该次写操作，地址根据图 3 所示结构进行细化处理，保证多维 DMA 传输的正确性。

[0037] 本发明在多维 DMA 传输过程中，对于高一维的 DMA 传输，它是以低一维的 DMA 传输过程为一个基本子过程，完成 Num 个起始地址进行等差偏移的子过程，如此反复嵌套直到完成最高维的 DMA 操作，即完成整个多维 DMA 传输。

[0038] 下面将详细描述本发明的具体实施方式，以展示本发明的各个具体构成部分和实现细节。

[0039] 图 4 所示为根据本发明的多维 DMA 传输装置的一个实施例，在该实施例中，该多维 DMA 传输装置实施为一个 DMA 控制器，该 DMA 控制器包括：读 Buffer 400、数据 FIFO 401、写 Buffer 402、读控制器 403 和写控制器 404。其中，读 Buffer 400 和写 Buffer 402 均连接于数据 FIFO，读控制器 403 与读 Buffer 400、数据 FIFO 均相连接，写控制器 404 与写 Buffer 402、数据 FIFO 401、读 Buffer 400 均相连接，。

[0040] 读 Buffer 400 用于暂存读操作过程中新读入的数据，包括读状态寄存器（未示出），该读状态寄存器用于存储标志该读 Buffer 400 中有效数据的情况的状态信息。

[0041] 数据 FIFO 401 用于存储因读写速度不同而暂时积压下来的数据。

[0042] 写 Buffer 402 用于暂存在写操作过程中将要写出的数据，包括写状态寄存器（未示出），该写状态寄存器用于存储标志该写 Buffer402 中有效数据情况的状态信息。

[0043] 读控制器 403 用于控制读操作过程，发送读请求并对该请求的回复进行处理。在进行读操作时，当所请求读取的数据返回时，将读取的数据暂存于该读 Buffer 400 和数据 FIFO 中，并修改读 Buffer 400 中的读状态寄存器的状态信息。

[0044] 写控制器 404 用于控制写操作过程，发送写请求并对该请求的回复进行处理。在进行写操作时，根据 DMA 的实际操作状态进行判断。主要分为两种情况：一种情况是，写数据来自于写 Buffer 402 和数据 FIFO 401，当这两个部件中存储的数据之和足够当次写操作

所需数据时，写控制器 404 向外部发送写操作请求，并将写 Buffer 402 和数据 FIFO 401 中的数据依次写出；另一情况是，当数据 FIFO 401 为空并且已经到达 DMA 传输末尾时，写控制器 404 从写 Buffer 402 和读 Buffer 400 中选取数据进行写出，以保证剩余数据的传输完成。

[0045] 以下描述根据本发明的多维 DMA 传输方法的具体实施方式。

[0046] 根据本发明，通过在各缓存器（即读 Buffer 400、数据 FIFO 401 和写 Buffer 402，之间的数据传输实现多维数据的重组和分割。

[0047] 在一个具体实施例中，多维 DMA 的传输方法包括如下步骤：A、读操作控制步骤；B、写操作控制步骤。

[0048] 在读操作控制步骤 A 中，读控制器 403 控制读操作过程，发送读请求并对该请求的回复进行处理，并修改读 Buffer 400 和数据 FIFO 401 内的数据及状态信息，使 DMA 存入数据 FIFO 401 前进行了对齐操作而形成一个连续的数据块。该步骤可以分为图 5 和图 6 两种情况，下面将详细说明。

[0049] 在写操作控制步骤 B 中，写控制器 404 控制写操作过程，发送写请求并对该请求的回复进行处理，修改读 Buffer 400、数据 FIFO 401 和写 Buffer 402 内的数据及状态信息，以便从连续的数据块中截取数据并写出。该步骤可以分为图 7、图 8 和图 9 三种情况，也将在下面详细说明。

[0050] 图 5 显示了当读 Buffer 400 中空位足够当次读入操作时的存储区变化情况的一个示例。该读 Buffer 400 存储数据的模式为循环方式。如图 5 所示，假设端口位宽 W 为 8 个最小数据宽度（最小数据宽度是连续的不可分割的数据，即最小粒度），读入的数据只有 4 个有效数据，为 D2 ～ D5，而读 Buffer 400 中只有 3 个有效数据，此时将 D2 ～ D5 进行循环移位到合适位置，然后将读入数据接到读 Buffer 400 中最后一个有效数据 (B5) 之后，于是，读 Buffer 400 中变为拥有 7 个有效数据，即从 B3 所在位置开始，到 D5 结束。

[0051] 当读到的数据数量和读 Buffer 400 中的数据数量之和不小于端口位宽 W 时，则将 W 个数据排好顺序放入数据 FIFO 401 中，剩余数据放入读 Buffer 400 中。

[0052] 图 6 显示了当读 Buffer 400 中空位不够当次读入操作时的存储区变化情况的一个示例。如图 6 所示，在此示例中，W 仍设为 8 个最小数据宽度，读入的数据有 5 个有效数据，为 D2 ～ D6，而读 Buffer 400 中也有 5 个有效数据，为 B1 ～ B5，此时，读 Buffer 400 中的 5 个数据和读入数据中的前 3 个 (8-5 ＝ 3) 数据组合起来（由于读 Buffer 400 中数据为上一次来的数据，因此将读 Buffer 400 中的数据放在前面，当次读入数据放在后面）后放入数据 FIFO 401 中，将剩下的读入数据 (D5 ～ D6) 放入读 Buffer 400 中。通过这样的读控制，将读入数据存储以便写控制器 404 使用。

[0053] 对于写控制器 404 来说，根据当前 DMA 所处状态和所有缓冲器（包括读 Buffer、数据 FIFO 和写 Buffer) 数据状态进行判断，如果有效数据足够当次写出，则发送写出请求，否则不发送写出请求。此处的判断遵循一次传输尽可能多的原则。以端口位宽 W 为 8 个最小数据宽度例，当次写出地址为 Addr，使用％表示求模操作，则当次最大能传输 8-Addr％ 8 个数据；将该结果与最小维度 DMA 剩余操作数量进行比较，取较小值得出当次应传输多少数据 (D)。由此，当次写操作须等待存可用数据不小于 D 时才进行写数据传输。

[0054] 如果写控制器 404 进行写操作，则从所有缓冲器（包括读 Buffer、数据 FIFO 和写

Buffer) 获取当次写操作所需数据并修改其状态信息以完成该次写操作。写操作基本上分为三种情况。

[0055] 当需要写出的数据数量不大于写 Buffer 中有效数据的数量时，直接从写 Buffer 读取数据，不需要修改其它存储单元。

[0056] 图 7 显示了当写 Buffer 402 中数据足够当次写出操作时的存储区变化情况的一个示例。如图 7 所示，设端口位宽 W 仍为 8 个最小数据宽度，写 Buffer 402 中有 5 个有效数据，为 B1 ～ B5，需要写出 3 个数据，此时，直接从写 Buffer 402 中取 3 个数据 B1 ～ B3(B1 为第一个有效数据) 并移位到合适位置后写出，并且修改写 Buffer 402 的状态信息。

[0057] 当需要写出数据数量大于写 Buffer 402 中的有效数据量，并且数据 FIFO 401 不空时，数据 FIFO 401 弹出 W 个数据，将写 Buffer 402 中的数据和这 W 个数据拼接后截取需要写出的数据长度的数据并写出，并将剩余数据放入写 Buffer 402 中。

[0058] 图 8 显示了当写 Buffer 402 中的数据不够当次写出操作时的存储区变化情况的一个示例。如图 8 所示，设 W 仍为 8，写 Buffer 402 中有 5 个有效数据，为 B1 ～ B5，需要写出 7 个数据，则从数据 FIFO 401 中弹出 8 个数据后将其和写 Buffer 402 数据组合 (写 Buffer 402 中数据在前)，将前 7 个数据后进行合理的移位写出，将剩下的 6 个 (5+8-7 ＝ 6) 数据存入写 Buffer402 中。

[0059] 如果当次 DMA 传输的所有读操作已完成并且数据 FIFO 401 为空，则将写 Buffer 402 中的数据和读 Buffer 400 中数据拼接后截取需要写出的数据长度的数据并写出，并将剩余数据放入写 Buffer 402 中。

[0060] 图 9 显示了 DMA 传输结尾时所有缓冲器 (包括读 Buffer、数据 FIFO 和写 Buffer) 可能的变化情况的一个示例。如图 9 所示，设 W 仍为 8，写 Buffer 402 中有 3 个有效数据，为 B1 ～ B3，读 Buffer 400 中有 4 个有效数据 (由于是循环存储，数据顺序为 D6、D7、D0、D1)，需要写出 5 个数据，此时，将写 Buffer 402 中的数据和读 Buffer 400 中的数据组合 (写 Buffer402 中数据在前)，选取前 5 个数据后进行合理的移位写出，将剩下的 2 个 (3+4-5 ＝ 2) 数据存入写 Buffer 402 中。

[0061] 如果在时序要求比较高的设计中，可以修改写控制器 404，一般情况下，写出数据扩展为在数据 FIFO 401，写 Buffer 402 和读 Buffer 400 中进行选择，这样可以提高数据传输时序，保证多维 DMA 传输性能的进一步提升。但此方法在控制逻辑上较本设计中的实现要复杂许多，综合考虑一般情况下写出数据从数据 FIFO 401 和写 Buffer 402 进行选择。

[0062] 根据本发明，由于多维 DMA 数据当读入时在存入数据 FIFO 401 前进行了自动对齐操作，使数据形成一个连续的数据块，实现了多维数据的无缝连接。而写出操作从连续的数据块中截取数据并进行合适的位移，保证写出操作按多维 DMA 要求进行操作。

[0063] 本发明的多维 DMA 传输装置与方法适合于矩阵分割、矩阵重组、矩阵扩展等多种 DSP 基本操作，计划了 DSP 核处理过程，将简单的处理过程与复杂处理过程相分离后并行执行，减少了核处理时间。

[0064] 以上所述的具体实施例，对本发明的目的、技术方案和有益效果进行了进一步详细说明，所应理解的是，以上所述仅为本发明的具体实施例而已，并不用于限制本发明，凡在本发明的精神和原则之内，所做的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。

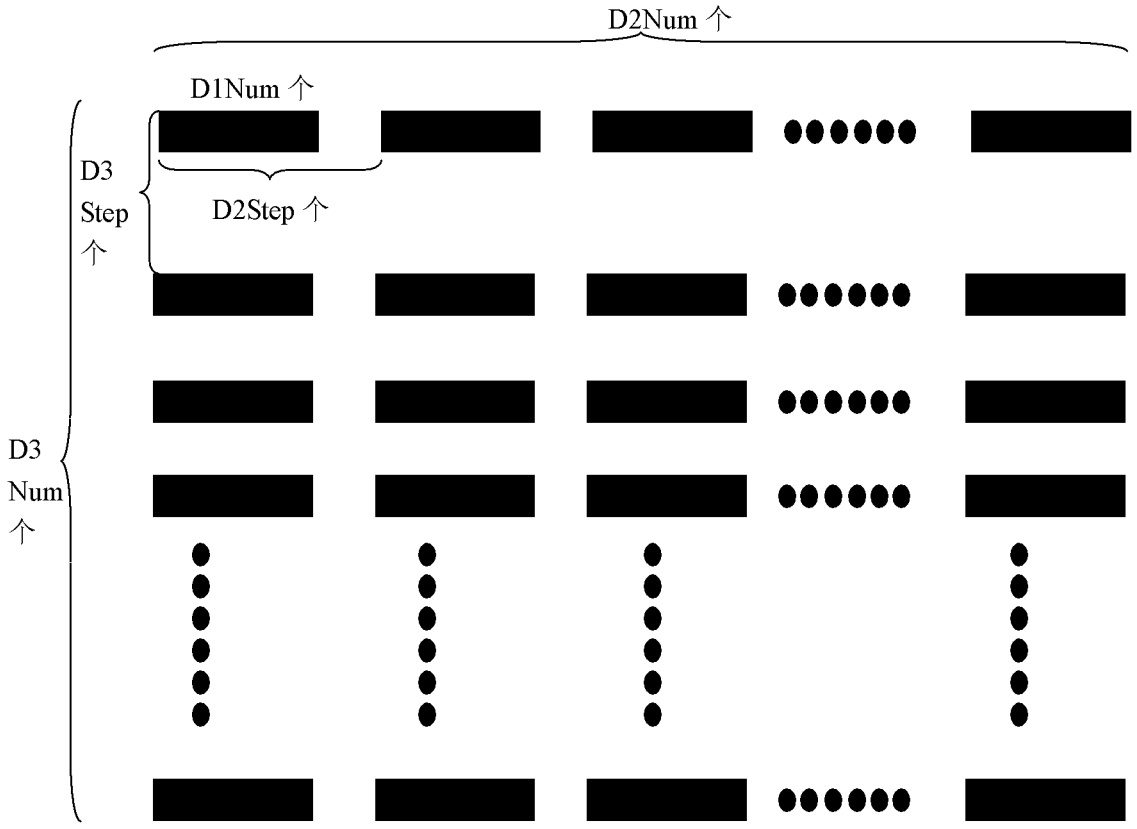


图 1

```
for(DnRCounter=0;DnRCounter<DnRNum;DnRCounter++) {
  ........

  for(D2RCounter=0;D2RCounter<D1RNum;D2RCounter++) {

    .....//读取D1RNum个数据，地址从D1RAddr开始

    D2RAddr=D2RAddr+D2RStep;
    D1RAddr=D2RAddr;
  }
  .......

  DnRAddr=DnRAddr+DnRStep;
  Dn-1RAddr=DnRAddr;
}
```

图 2

```
for(DnWCounter=0;DnWCounter<DnRNum;DnWCounter++) {
  ........

  for(D2WCounter=0;D2WCounter<D1RNum;D2WCounter++) {

    .....//写D1WNum个数据，地址从D1WAddr开始

    D2WAddr=D2WAddr+D2WStep;
    D1WAddr=D2WAddr;
  }
  .......

  DnWAddr=DnWAddr+DnWStep;
  Dn-1WAddr=DnWAddr;
}
```

图 3

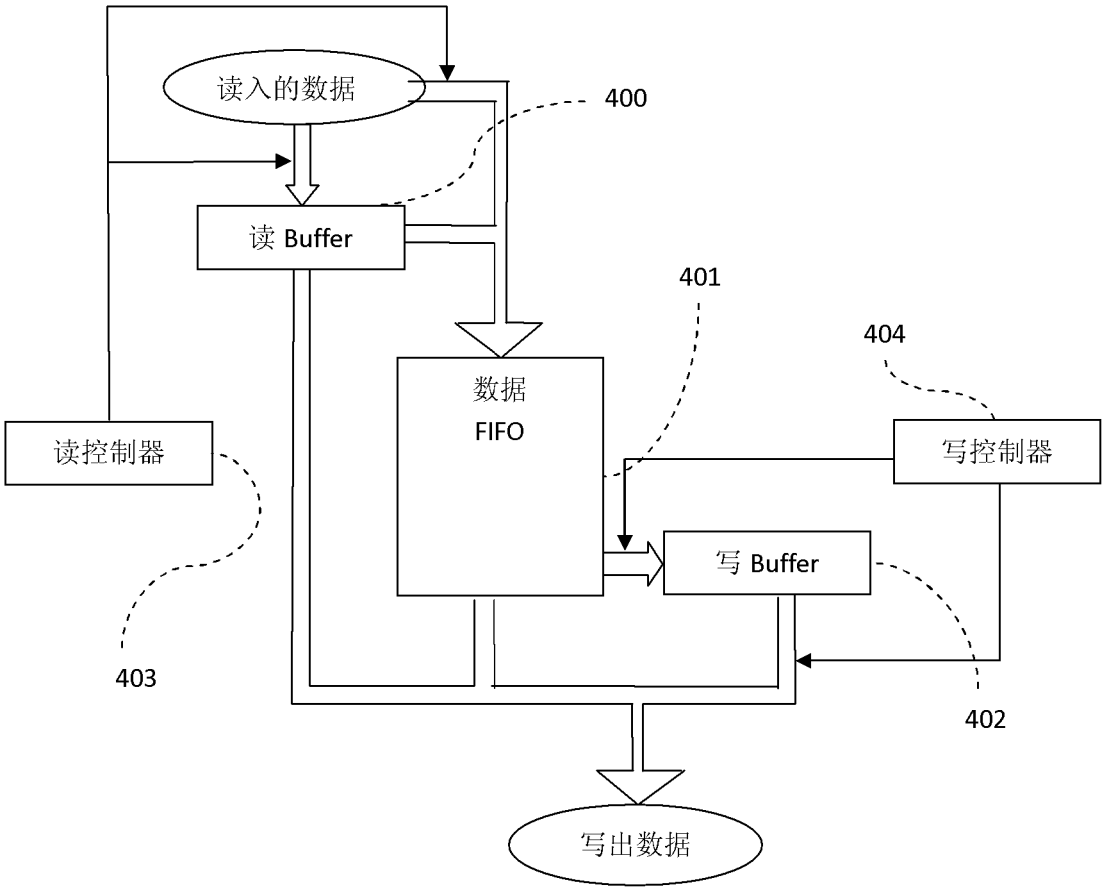


图 4

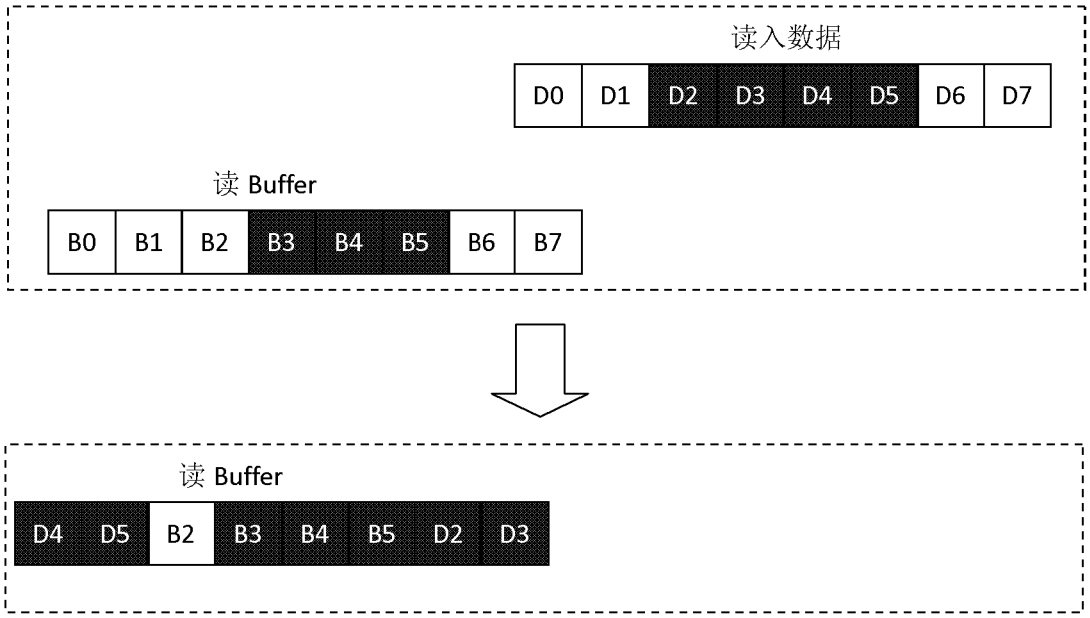

读入数据
D0 D1 D2 D3 D4 D5 D6 D7
读 Buffer
B0 B1 B2 B3 B4 B5 B6 B7
读 Buffer
D4 D5 B2 B3 B4 B5 D2 D3

图 5

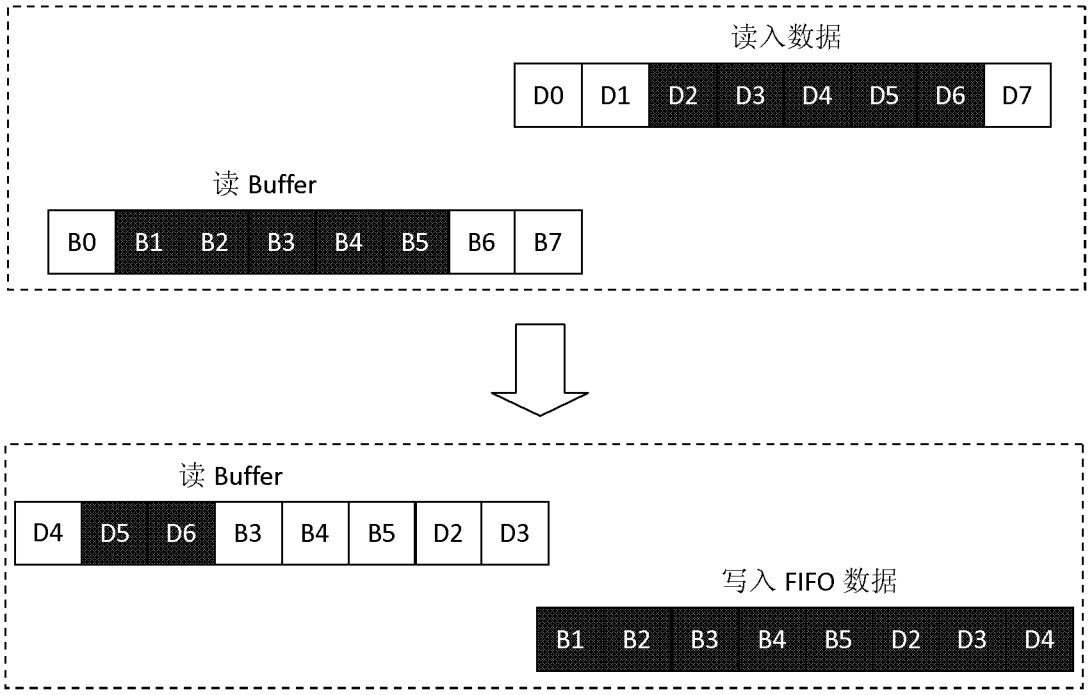

读入数据
D0 D1 D2 D3 D4 D5 D6 D7
读 Buffer
B0 B1 B2 B3 B4 B5 B6 B7
读 Buffer
D4 D5 D6 B3 B4 B5 D2 D3
写入 FIFO 数据
B1 B2 B3 B4 B5 D2 D3 D4

图 6

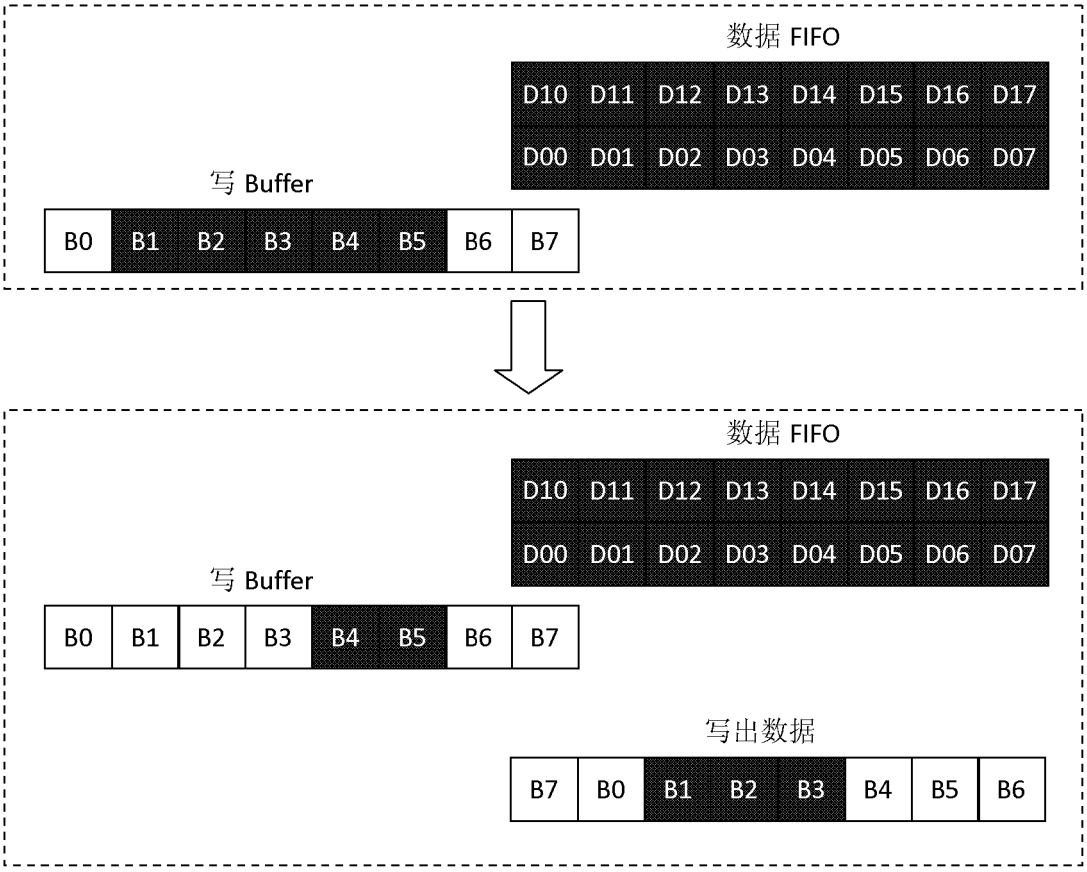

数据 FIFO
D10 D11 D12 D13 D14 D15 D16 D17
D00 D01 D02 D03 D04 D05 D06 D07
写 Buffer
B0 B1 B2 B3 B4 B5 B6 B7
数据 FIFO
D10 D11 D12 D13 D14 D15 D16 D17
D00 D01 D02 D03 D04 D05 D06 D07
写 Buffer
B0 B1 B2 B3 B4 B5 B6 B7
写出数据
B7 B0 B1 B2 B3 B4 B5 B6

图 7

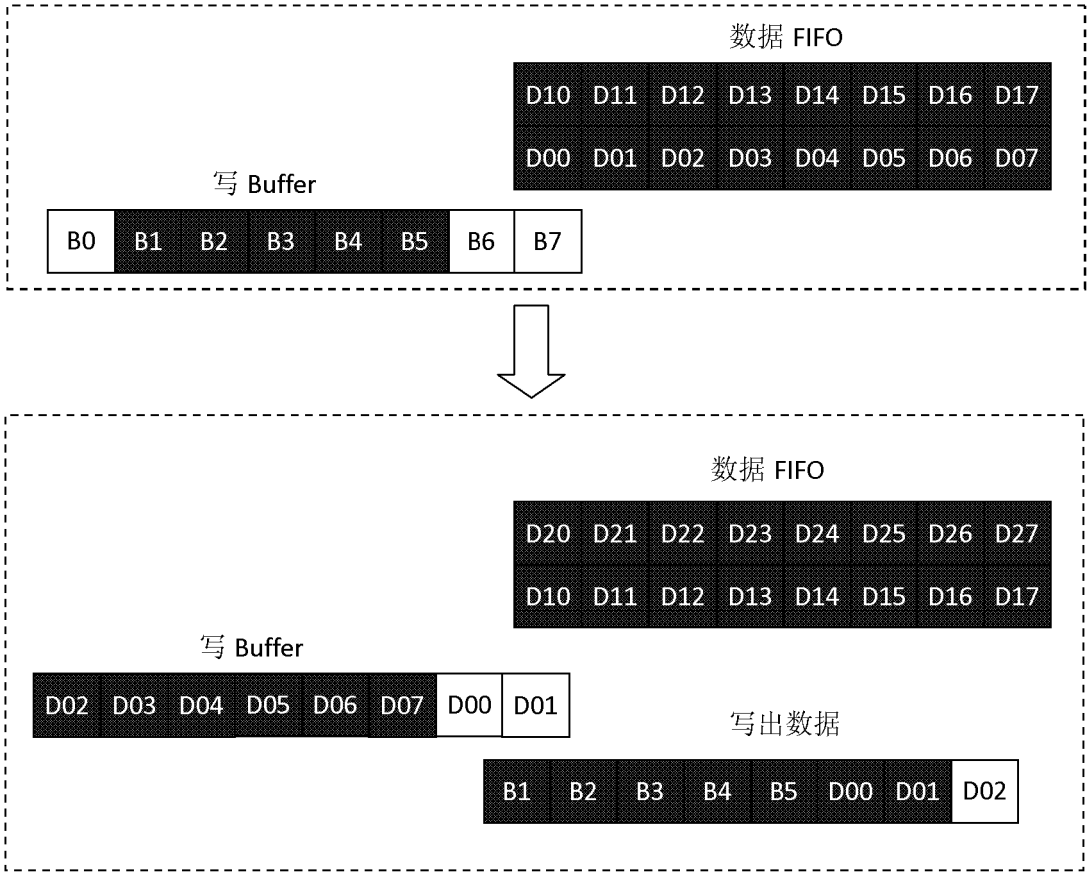

数据 FIFO
D10 D11 D12 D13 D14 D15 D16 D17
D00 D01 D02 D03 D04 D05 D06 D07
写 Buffer
B0 B1 B2 B3 B4 B5 B6 B7
数据 FIFO
D20 D21 D22 D23 D24 D25 D26 D27
D10 D11 D12 D13 D14 D15 D16 D17
写 Buffer
D02 D03 D04 D05 D06 D07 D00 D01
写出数据
B1 B2 B3 B4 B5 D00 D01 D02

图 8

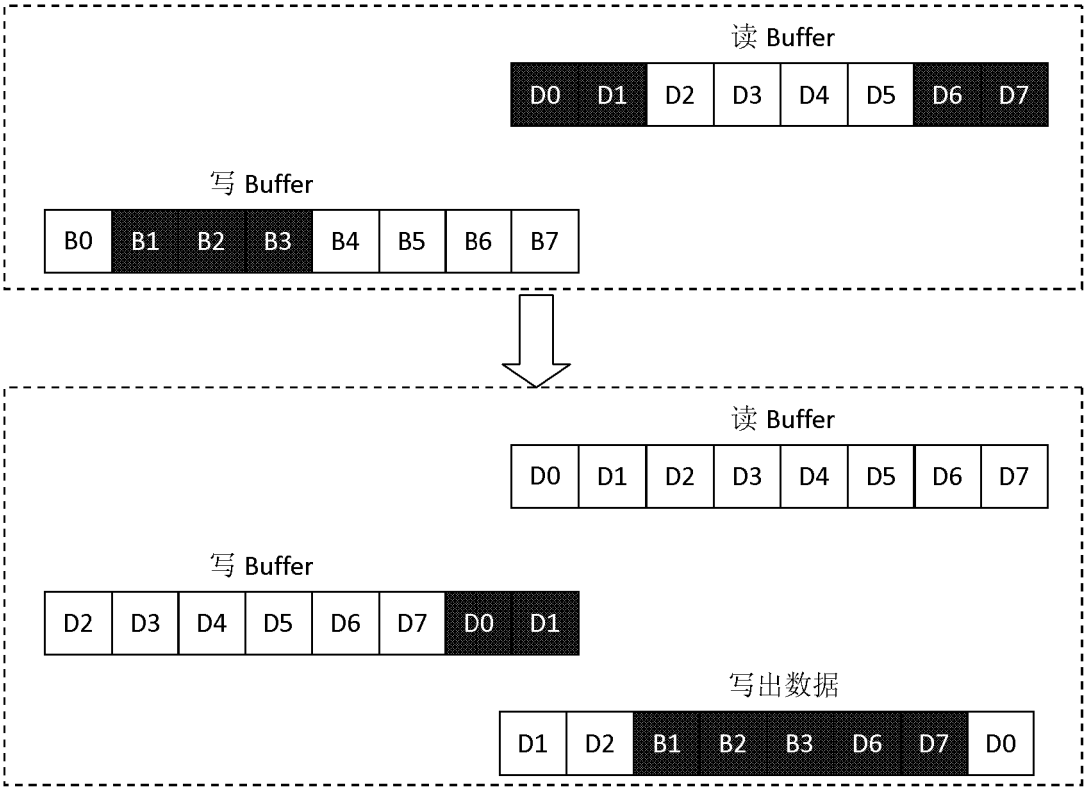

读 Buffer
D0 D1 D2 D3 D4 D5 D6 D7
写 Buffer
B0 B1 B2 B3 B4 B5 B6 B7
读 Buffer
D0 D1 D2 D3 D4 D5 D6 D7
写 Buffer
D2 D3 D4 D5 D6 D7 D0 D1
写出数据
D1 D2 B1 B2 B3 D6 D7 D0

图 9