



[12] 发明专利说明书

专利号 ZL 200510061703.6

[45] 授权公告日 2008 年 10 月 8 日

[11] 授权公告号 CN 100424654C

[22] 申请日 2005.11.25

[21] 申请号 200510061703.6

[73] 专利权人 杭州中天微系统有限公司

地址 310012 浙江省杭州市西湖区华星路
99 号东部软件园创业大厦 A403

[72] 发明人 严晓浪 秦兴 刘大可 葛海通
罗晓华

[56] 参考文献

US5790442A 1998.8.4

US2003/0233195A1 2003.12.18

CN1329439A 2002.1.2

US5872725A 1999.2.16

审查员 陈颖

[74] 专利代理机构 杭州天正专利事务所有限公司
代理人 王兵 黄美娟

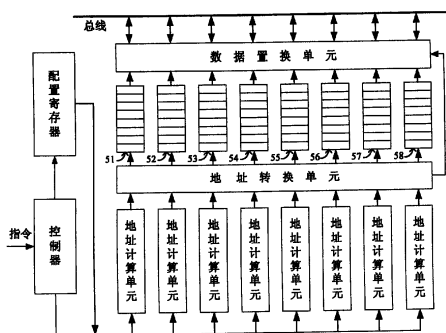
权利要求书 7 页 说明书 22 页 附图 8 页

[54] 发明名称

一种矩阵数据存取方法及其矩阵数据存储装置

[57] 摘要

一种矩阵数据存取方法，包括下列步骤：包含有预定的高维矩阵数据存储空间，它经过多级的降维处理转化为多个二维矩阵数据存储空间，每一个二维矩阵数据存储空间再分解成相互逻辑独立的二维子矩阵数据存储空间；数据在高维矩阵数据存储空间中的存放方式：由高维到低维，在每一个二维子矩阵数据存储空间中的存放方式：由左到右，由上到下，按照模存放；将二维子矩阵数据存储空间通过数据存储模式的配置映射到多个物理存储器中，根据配置条件从多个物理存储器的每一个存储器中的相应地址中读取需要存取的二维子矩阵数据存储空间中的数据块的数据。以及提供了一种矩阵数据存储装置。本发明存储效率高、能满足不同类型数据存取的要求。



1、一种矩阵数据存取方法，该方法包括下列步骤：

包含有预定的高维矩阵数据存储空间，它经过多级的降维处理最终转化为多个二维矩阵数据存储空间，每一个二维矩阵数据存储空间再分解成一系列的相互逻辑独立的二维子矩阵数据存储空间，所述的二维子矩阵为 $m \times n$ 矩阵， m 、 n 为自然数；

数据在所述高维矩阵数据存储空间进行存放时采取由高维到低维的方式，在每一个二维子矩阵数据存储空间中进行存放时采取由左到右，由上到下的顺序，按照模 N 的结果进行存放的方式， N 是指物理存储器的模数；

将所述二维子矩阵数据存储空间通过数据存储模式的配置映射到多个物理存储器中，所述数据存储模式的配置内容包括：基地址、数据元素宽度、行跨度、列跨度和数据存储模式；

在存取所述二维子矩阵数据存储空间的数据时，指定需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移；以及，

需要存取的二维子矩阵数据存储空间中的数据块的排列方向；

需要存取的二维子矩阵数据存储空间中的数据块的长度；

从而从多个物理存储器的每一个存储器中的相应地址中读取需要存取的二维子矩阵数据存储空间中的数据块的数据；

在读取数据时，从物理存储器中读出的数据按照需要经过数据置换以使其满足需要的读取数据顺序；

在存入数据时，数据在送入物理存储器中需要经过数据置换以使其满足物理存储器中需要的存放顺序。

2、如权利要求 1 所述的一种矩阵数据存取方法，其特征在于：所述数据存储模式是行存储模式，在行存储模式下，支持的存取的数据块的排列方向为行排列，每次所能进行存取的数据块长度不超过物理存储器的存储单元总数。

3、如权利要求 1 所述的一种矩阵数据存取方法，其特征在于：所述数据存储模式是列存储模式，在列存储模式下，支持的存取的数据块的排列方向为列排列，每次所能进行存取的数据块长度不超过物理存储器的存储单元总数。

4、如权利要求 1 所述的一种矩阵数据存取方法，其特征在于：所述数据存储模式是行列交错存储模式，在行列交错存储模式下，支持的存取的数据块的排列方向

为行排列和列排列；

在行列交错存储模式下，数据进行存取时要进行交错，交错包括下列步骤：

(1)、由独立进行交错的数据单元组成的正方形矩阵区域，所述正方形矩阵区域是指在行方向上包含的行跨度的个数等于这个区域所包含的行数，所述包含的行跨度的个数为交错额度，并且交错额度不超过物理存储器的存储单元总数所包含的行跨度的个数；

(2)、将所述正方形矩阵区域中的第一行数据次序保持不变；

(3)、将所述正方形矩阵区域中的第二行数据的最后一个行跨度中的数据元素转移到第一个行跨度相应位置中去，而原来的从第一个行跨度到倒数第二个行跨度中的数据元素依次向后转移一个行跨度；

(4)、将所述正方形矩阵区域中的第三行数据的倒数第二个行跨度中的数据元素转移到第一个行跨度相应位置中去，最后一个行跨度中的数据元素转移到第二个行跨度相应位置中去，而原来的从第一个行跨度到倒数第三个行跨度中的数据元素依次向后转移两个行跨度；

(5)、依次类推，每进行到下一行，则从行尾移到行首的数据元素增加一个，直到进行到所述正方形矩阵区域的最后一行结束。

5、如权利要求 4 所述的一种矩阵数据存取方法，其特征在于：所述的交错方法还包括：

(6)、所述正方形矩阵区域进行交错后得出的新的各行可以进行排列组合，经过排列组合产生的每一种结果均可作为本交错方式的结果。

6、如权利要求 1—5 之一所述的一种矩阵数据存取方法，其特征在于：所述需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移的偏移量为正数或负数。

7、如权利要求 4 或 5 所述的一种矩阵数据存取方法，其特征在于：依据交错额度和行跨度确定行列交错存储模式的修正表方式如下表 1：

行 列	0	1	2	3	...	M-3	M-2	M-1
0	+0	+0	+0	+0	...	+0	+0	+0
1	+X	+X	+X	+X	...	+X	+X	-X(M-1)
2	+2X	+2X	+2X	+2X	...	+2X	-X(M-2)	-X(M-2)
3	+3X	+3X	+3X	+3X	...	-X(M-3)	-X(M-3)	-X(M-3)

...
M-3	+X(M-3)	+X(M-3)	+X(M-3)	-3X	...	-3X	-3X	-3X
M-2	+X(M-2)	+X(M-2)	-2X	-2X	...	-2X	-2X	-2X
M-1	+X(M-1)	-X	-X	-X	...	-X	-X	-X

表 1

表 1 中, M 为交错额度, X 表示行跨度 $xstride$, $X(M-1)$ 表示行跨度 X 与 $(M-1)$ 的乘积, $X(M-2)$ 表示行跨度 X 与 $(M-2)$ 的乘积, $X(M-3)$ 表示行跨度 X 与 $(M-3)$ 的乘积, 依次类推;

根据各个数据在独立进行交错的数据单元组成的正方形矩阵区域中所处的行坐标和列坐标查表 1 得到修正值, 然后在物理地址计算单元得到的未修正的物理地址的基础上再加表中查得的修正偏移量就得到了修正后的各个数据所对应的修正后的物理地址, 如所述正方形矩阵区域进行交错后得出的新的各行可以进行排列组合形成新的交错结果, 则对应的修正表 1 也进行相同的各行的排列组合即可, 其修正值相同。

8、如权利要求 2 所述的一种矩阵数据存取方法, 其特征在于:

设基地址为 $base$, 需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y , 行跨度为 $xstride$, 列跨度为 $ystride$;

需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址 z_i 计算式为 (1):

$$z_i = base + (x + i) \times xstride + y \times ystride \quad (1)$$

其中, i 为各个数据相对于该数据块的首地址的距离, 单位为行跨度;

将全局物理地址 z_i 同时进行除以 N 和模 N 的运算后得到每一个存储器单元所需要的物理地址信息, N 是指所述物理存储器的模数。

9、如权利要求 3 所述的一种矩阵数据存取方法, 其特征在于:

设基地址为 $base$, 需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y , 行跨度为 $xstride$, 列跨度为 $ystride$;

需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址 z_i 计算式为 (2):

$$z_i = base + x \times xstride + (y + i) \times ystride \quad (2)$$

其中, i 为各个数据相对于该数据块的首地址的距离, 单位为列跨度;

将全局的物理地址 z_i 同时进行除以 N 和模 N 的运算后得到每一个存储器单元所需要的物理地址信息, N 是指所述物理存储器的模数。

10、如权利要求 7 所述的一种矩阵数据存取方法, 其特征在于:

设基地址为 $base$, 需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y , 行跨度为 $xstride$, 列跨度为 $ystride$;

行列交错存储模式的交错额度为 N , 需要存取的二维子矩阵数据存储空间中的数据块的各个数据在物理存储器中对应的物理地址的计算方法包括以下步骤:

1)、计算未经修正的数据块中各个数据对应的全局物理地址 z_i ;

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列时, 计算式为 (1):

$$z_i = base + (x + i) \times xstride + y \times ystride \quad (1)$$

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列时, 计算式为 (2):

$$z_i = base + x \times xstride + (y + i) \times ystride \quad (2)$$

其中 i 为各个数据相对于该数据块的首地址的距离, 单位为行跨度;

2)、同时计算数据块的各个数据在独立进行数据交错的正方形区域中所在位置的行坐标 $R(z_i)$ 和列坐标 $C(z_i)$;

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列时, 计算式为 (3)、(4):

$$R(z_i) = \begin{cases} (x + i) \% N \times xstride & (x + i \geq 0) \\ (N + (x + i) \% N) \times xstride & (x + i < 0) \end{cases} \quad (3)$$

$$C(z_i) = \begin{cases} y \% N & (y \geq 0) \\ N + y \% N & (y < 0) \end{cases} \quad (4)$$

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列时, 计算式为 (5)、(6):

$$R(z_i) = \begin{cases} (x \% N) \times xstride & (x \geq 0) \\ (N + x \% N) \times xstride & (x < 0) \end{cases} \quad (5)$$

$$C(z_i) = \begin{cases} (y + i) \% N & (y + i \geq 0) \\ N + (y + i) \% N & (y + i < 0) \end{cases} \quad (6)$$

3)、按照不同的交错方式得到修正表, 依据前面求出的 $R(z_i)$ 和 $C(z_i)$ 得出数据块的各个数据在交错过程中移动的偏移量, 利用这个偏移量对 z_i 进行修正, 从而

得出需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址；

4)、将全局的物理地址 z_i 同时进行除以 N 和模 N 的运算后得到每一个存储器单元所需要的物理地址信息， N 是指所述物理存储器的模数。

11、如权利要求 6 所述的一种矩阵数据存取方法，其特征在于：所述物理存储器由多个位宽为字节的存储器单元并联组成，所述的基地址、数据元素宽度、行跨度、列跨度的数值以字节大小为单位。

12、一种用于实现如权利要求 1 所述的矩阵数据存取方法的矩阵数据存储装置，其特征在于：

所述的矩阵数据存储装置的逻辑存储空间包含有预定的高维矩阵数据存储空间，所述高维矩阵数据存储空间经过多级的降维处理最终转化为多个二维矩阵数据存储空间，每一个二维矩阵数据存储空间再分解成一系列的相互逻辑独立的二维子矩阵数据存储空间，所述的二维子矩阵为 $m \times n$ 矩阵， m 、 n 为自然数；

所述的矩阵数据存储装置包括：

配置寄存器，包括基址寄存器，数据元素字节宽度寄存器，行跨度寄存器，列跨度寄存器，循环存取次数寄存器，循环存取使能寄存器，数据存储模式寄存器；

地址计算单元组，由数量与物理存储器单元相同的地址计算单元组成，每一个地址计算单元具有设定的模块号，所述地址计算单元用于根据配置寄存器中的内容和模块号，按照控制器选择的数据通路，从控制器读取需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据所对应的物理地址；

地址转换模块，用于把地址计算单元组生成的各路物理地址信息来转化成为各个物理存储器单元的地址信息，输送到各个物理存储器单元；同时根据模块号为 0 的地址计算单元产生的物理地址信息计算需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元，并将这一信息输送到数据置换模块；

数据置换模块，用于在读取数据时，按照地址转换模块产生的需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的信息对物理存储器单元组产生的一组数据进行置换，以使其满足需要的读取数据顺序；以

及在写数据时，按照地址转换单元产生的需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的信息对送入物理存储器单元组的一组数据进行置换以使其满足物理存储器中需要的存放顺序；

物理存储器单元组，由一组并行的物理存储器单元组成；

控制器，用于从程序存储器中读取指令，并对指令进行解析转化为对配置寄存器的配置信息，或是将从指令中解析出的需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移、需要存取的二维子矩阵数据存储空间中的数据块的排列方向输送到地址计算单元组，同时根据需要存取的二维子矩阵数据存储空间中的数据块的长度决定使能地址计算单元的数量，根据需要存取的二维子矩阵数据存储空间中的数据块的排列方向选择地址计算单元的数据通路，将这些信息也输送到地址计算单元组；

总线，连接多个物理存储器单元组的数据通道。

13、如权利要求 12 所述的矩阵数据存储装置，其特征在于：所述地址计算单元包括：

物理地址计算单元，用于根据控制器选择的数据通路和模块号，以及由控制器输入的需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据所对应的全局物理地址，所述全局物理地址在行存储模式下将直接输出，在行列交错存储模式下将输入到地址修正单元进行修正后输出；

行坐标计算单元，用于在行列交错存储模式下，根据自身的模块号和需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据在独立进行数据交错的正方形矩阵区域中所在的行坐标；

列坐标计算单元，用于在行列交错存储模式下，根据自身的模块号和需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据在独立进行数据交错的正方形矩阵区域中所在的列坐标；

地址修正单元，用于在行列交错存储模式下，根据物理地址计算单元生成的全局物理地址、行坐标计算单元生成的数据在独立进行数据交错的正方形矩阵区域中所在的行坐标、列坐标计算单元生成的数据在独立进行数据交错的正方形矩

阵区域中所在的列坐标进行修正，最终生成修正后的物理地址；

输出数据选择器，用于在行列交错模式下选择的是输出地址修正单元的结果，而在行存储模式和列存储模式下选择的是直接输出物理地址计算单元的计算结果。

14、如权利要求 12 或 13 所述的矩阵数据存储装置，其特征在于：所述数据置换模块包括：

数据控制单元，用于接受来自地址转换模块的首字节所在的物理存储器单元模块号，将其转化为对数据选择单元的选择信号，输送给读取数据选择单元或写入数据选择单元；

读取数据选择单元，它由与物理存储器单元数量相同的一组选择器组成，每一个选择器用于从多个物理存储器单元中根据数据控制单元的选择信号选择各个读出数据以便并行输出需要读取的数据块的所有数据；

写入数据选择单元，它由与物理存储器单元数量相同的一组选择器组成，每一个选择器用于根据数据控制单元的选择信号选择要写入各个物理存储器单元中的数据以便向物理存储器单元组并行输出需要写入的数据块的所有数据。

一种矩阵数据存取方法及其矩阵数据存储装置

(一) 技术领域

本发明涉及一种矩阵数据存取方法及其矩阵数据存储装置。

(二) 背景技术

矢量处理器广泛地被应用在多媒体处理和科学计算等领域，它操作的数据对象多为维数不同、数据类型各异的矩阵数据，所以，实现高效的矩阵数据存取是十分重要。

如中国发明专利 95103825.7，专利名称为应用线性变换进行图像处理地并行存储装置，公开了一种使用线性变换进行图像处理地并行存储装置，包括多个地址计算单元、多个存储单元以及奥米伽网络和移位电路单元，所述地址计算单元适于根据存取方式、参考坐标和相应的存储模块的存储模块数来产生每个相应于存储模块中的一个模块的地址；所述的存储单元适于接收每个相应的地址计算单元和读/写信号所产生的地址，并且执行实际数据的存储、输入、输出；所述的奥米伽网络适于接收来自与其相应的存储单元的数据，对该数据产生合适的通路使其产生与一处理器所需逻辑顺序一致的存储模块的实际处理顺序，并通过该产生的通路输出数据，所述的移位电路单元适于根据该存取方式所指示的输入在每个奥米伽网络和数据终端之间产生合适的通路。这种装置是通过在多个存储模块内适当的分配并行存取的数据而获得了高的并行性能。这一类方案实际上就是通过存储模块之间地址空间的冗余作为代价实现并行，对于某一个可存取的诸如水平线、垂直线或方框之类的一定几何形状的图形点子集，实际上在每一个存储模块中都要在有一个与之相应的存储地址空间，存储效率较低，并且存在着无法在一任意的位置上以水平线、垂直线、二维方框方式来存取。虽然这种装置通过将存储模块数增加到可同时存取图像点数目的两倍来克服这一缺陷，然而其代价是再一次增加了存储模块的冗余度，降低了存储效率。

如中国发明专利 01139343.2，专利名称是使用存储器的方法、两维数据存取存储器以及操作处理设备，公开了一种使用存储器的方法，该方法包括下列步骤：包含有预定的两维存储空间，它由预定的按照两维方式形成的并沿纵向和横向方向排列的虚拟最小两维存储器空间构成，将所述虚拟最小两维存储器空间的各个

地址预先分配给多个物理存储器中按照与所述虚拟最小两维存储器的大小相关的方式确定的每一个存储器内的预定地址;在读取所述两维存储器空间中的数据时,指定所述虚拟最小两维存储器中的读取地址以及,参照指定的读取地址指定预定的读取方向,从而,根据指定的地址和指定的方式从多个物理存储器的每个存储器中的相应的地址中读取所述虚拟最小两维存储器中的数据,在读取数据时,还指定数据的读取存取模式,如数据跨越了参照数据现有范围的边缘,则通过将不包含在参照数据的现有范围内的数据读取地址修改成位于参照数据现有范围的边缘处的预定数据的地址而读取数据。该方法将整个两维存储空间划分成一系列的 $N \times N$ 的两维最小存储空间,且 N 相对较小,一方面不满足实际应用中所需存储空间大小不定的要求,必然造成存储的不方便和存储空间的冗余;另外地址计算方法限制在两维最小存储空间中,采用一维地址计算方法,编程人员需要将两维坐标偏移换算为一维正偏移,实际应用中,用户所见的一般为两维坐标偏移,且可为正负偏移,所以增加了计算成本,也不方便灵活使用。

(三) 发明内容

为了克服已有技术的数据存储方法存储效率不高、不能满足多种数据类型存取的要求的不足,本发明提供一种存储效率高、能满足不同类型数据存取的要求的矩阵数据存取方法及其矩阵数据存储装置。

本发明解决其技术问题所采用的技术方案是:

一种矩阵数据存取方法,该方法包括下列步骤:包含有预定的高维矩阵数据存储空间,它经过多级的降维处理最终可以转化为多个二维矩阵数据存储空间,每一个二维矩阵数据存储空间可以再分解成一系列的相互逻辑独立的二维子矩阵数据存储空间,所述的二维子矩阵为 $m \times n$ 矩阵, m 、 n 为自然数;

数据在所述高维矩阵数据存储空间进行存放时采取由高维到低维的方式,在每一个二维子矩阵数据存储空间中进行存放时采取由左到右,由上到下的顺序,按照模进行存放的方式;

将所述二维子矩阵数据存储空间通过数据存储模式的配置映射到多个物理存储器中,所述数据存储模式的配置内容包括:基地址、数据元素宽度、行跨度、列跨度和数据存储模式;

在存取所述二维子矩阵数据存储空间的数据时,指定需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行

偏移和列偏移；以及，

需要存取二维子矩阵数据存储空间中的数据块的排列方向；

需要存取二维子矩阵数据存储空间中的数据块的长度；

从而根据以上条件从多个物理存储器的每一个存储器中的相应地址中读取需要存取的二维子矩阵数据存储空间中的数据块的数据；

在读取数据时，从物理存储器中读出的数据要按照需要经过数据置换以使其满足需要的读取数据顺序；

在存入数据时，数据在送入物理存储器中需要经过数据置换以使其满足物理存储器中需要的存放顺序。

进一步，如所述数据存储模式是行存储模式，在行存储模式下，支持的存取的数据块的排列方向为行排列，每次所能进行存取的最大数据块长度为不超过物理存储器的存储单元总数。

或者，所述数据存储模式是列存储模式，在列存储模式下，支持的存取的数据块的排列方向为列排列，每次所能进行存取的最大数据块长度为不超过物理存储器的存储单元总数。

再或者，所述数据存储模式是行列交错存储模式，在行列交错存储模式下，支持的存取的数据块的排列方向为行排列和列排列；

在行列交错存储模式下，数据进行存取时要进行交错，交错包括下列步骤：

(1)、由独立进行交错的数据单元组成的正方形矩阵区域，所述正方形区域是指在行方向上包含的行跨度的个数等于这个区域所包含的行数，所述包含的行跨度的个数为交错额度，它不超过物理存储器的存储单元总数所包含的行跨度的个数；

(2)、将所述正方形区域中的第一行数据次序保持不变；

(3)、将所述正方形区域中的第二行数据的最后一个行跨度中的数据元素转移到第一个行跨度相应位置中去，而原来的从第一个行跨度到倒数第二个行跨度中的数据元素依次向后转移一个行跨度；

(4)、将所述正方形区域中的第三行数据的倒数第二个行跨度中的数据元素转移到第一个行跨度相应位置中去，最后一个行跨度中的数据元素转移到第二个行跨度相应位置中去，而原来的从第一个行跨度到倒数第三个行跨度中的数据元素依次向后转移两个行跨度；

(5)、依次类推，每进行到下一行，则从行尾移到行首的数据元素增加一个，直

到进行到所述正方形区域的最后一行结束。

所述的交错方法还包括：(6)、所述正方形区域进行交错后得出的新的各行可以进行排列组合，经过排列组合产生的每一种结果均可作为本交错方式的结果。

依据交错额度和行跨度确定行列交错存储模式的修正表方式如下表 1：

行 列	0	1	2	3	...	M-3	M-2	M-1
0	+0	+0	+0	+0	...	+0	+0	+0
1	+X	+X	+X	+X	...	+X	+X	-X(M-1)
2	+2X	+2X	+2X	+2X	...	+2X	-X(M-2)	-X(M-2)
3	+3X	+3X	+3X	+3X	...	-X(M-3)	-X(M-3)	-X(M-3)
...
M-3	+X(M-3)	+X(M-3)	+X(M-3)	-3X	...	-3X	-3X	-3X
M-2	+X(M-2)	+X(M-2)	-2X	-2X	...	-2X	-2X	-2X
M-1	+X(M-1)	-X	-X	-X	...	-X	-X	-X

表 1

上表中，M 为交错额度，X 表示行跨度 $xstride$ ， $X(M-1)$ 表示行跨度 X 与 (M-1) 的乘积， $X(M-2)$ 表示行跨度 X 与 (M-2) 的乘积， $X(M-3)$ 表示行跨度 X 与 (M-3) 的乘积，依次类推；

所述正方形区域进行交错后得出的新的各行可以进行排列组合形成新的交错结果，对应的修正表 1 也进行相同的各行的排列组合。

再进一步，所述需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移，其中偏移量为正数或负数。

更进一步，在行存储模式下，物理地址计算过程为：设基地址为 $base$ ，需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y ，行跨度为 $xstride$ ，列跨度为 $ystride$ ；

需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局的物理地址 z_i 计算式为 (1)：

$$z_i = base + (x + i) \times xstride + y \times ystride \quad (1)$$

其中, i 为各个数据相对于该数据块的首地址的距离, 单位为行跨度;

将全局的物理地址 z_i 进行除以 N 和取 N 的模的运算后得到每一个存储器单元所需要的物理地址信息, N 是指所述物理存储器的模数。

在列存储模式下, 物理地址计算过程为: 设基地址为 $base$, 需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y , 行跨度为 $xstride$, 列跨度为 $ystride$;

需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局的物理地址 z_i 计算式为 (2):

$$z_i = base + x \times xstride + (y + i) \times ystride \quad (2)$$

其中, i 为各个数据相对于该数据块的首地址的距离, 单位为列跨度;

将全局的物理地址 z_i 进行除以 N 和取 N 的模的运算后得到每一个存储器单元所需要的物理地址信息, N 是指所述物理存储器的模数。

在行列交错存储模式下, 物理地址计算过程为: 设基地址为 $base$, 需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y , 行跨度为 $xstride$, 列跨度为 $ystride$;

行列交错存储模式的交错额度为 N , 需要存取的二维子矩阵数据存储空间中的数据块的各个数据在物理存储器中对应的物理地址的计算方法包括以下步骤:

(1)、计算未经修正的数据块中各个数据对应的全局物理地址 z_i ;

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列时, 计算式为 (1):

$$z_i = base + (x + i) \times xstride + y \times ystride \quad (1)$$

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列时, 计算式为 (2):

$$z_i = base + x \times xstride + (y + i) \times ystride \quad (2)$$

其中 i 为各个数据相对于该数据块的首地址的距离, 单位为行跨度;

(2)、同时计算数据块的各个数据在独立进行数据交错的正方形区域中所在位置的行坐标 $R(z_i)$ 和列坐标 $C(z_i)$:

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列时, 计算式为 (3)、(4):

$$R(z_i) = \begin{cases} (x + i) \% N \times xstride & (x + i \geq 0) \\ (N + (x + i) \% N) \times xstride & (x + i < 0) \end{cases} \quad (3)$$

$$C(z_i) = \begin{cases} y \% N & (y \geq 0) \\ N + y \% N & (y < 0) \end{cases} \quad (4)$$

上式中，%为整除取余计算符号；

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列时，计算式为(5)、(6)：

$$R(z_i) = \begin{cases} (x \% N) \times xstride & (x \geq 0) \\ (N + x \% N) \times xstride & (x < 0) \end{cases} \quad (5)$$

$$C(z_i) = \begin{cases} (y+i) \% N & (y+i \geq 0) \\ N + (y+i) \% N & (y+i < 0) \end{cases} \quad (6)$$

(3)、按照不同的交错方式得到修正表，依据前面求出的 $R(z_i)$ 和 $C(z_i)$ 得出数据块的各个数据在交错过程中移动的偏移量，利用这个偏移量对 z_i 进行修正，从而得出需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址；

根据各个数据在独立进行交错的数据单元组成的正方形矩阵区域中所处的行坐标和列坐标查表得到修正值，即在物理地址计算单元得到的 z_i 的基础上再加表中查得的修正偏移量就得到了修正后的各个数据所对应的修正后的物理地址 z'_i 。如所述正方形区域进行交错后得出的新的各行可以进行排列组合形成新的交错结果，则对应的修正表1也进行相同的各行的排列组合即可，其修正值相同。

(4)、将全局的物理地址 z_i 进行除以N和取N的模的运算后得到每一个存储器单元所需要的物理地址信息，N是指所述物理存储器的模数。

所述物理存储器由多个位宽为字节的存储器单元并联组成，所述的基地址、数据元素宽度、行跨度、列跨度的数值以字节大小为单位。位宽也可以是双字节或多字节，只要配置的内容的单位与位宽一致即可。

一种用于实现所述的矩阵数据存取方法的矩阵数据存储装置，所述的矩阵数据存储装置的逻辑存储空间包含有预定的高维矩阵数据存储空间，所述高维矩阵数据存储空间经过多级的降维处理最终可以转化为多个二维矩阵数据存储空间，每一个二维矩阵数据存储空间可以再分解成一系列的相互逻辑独立的二维子矩阵数据存储空间，所述的二维子矩阵为 $m \times n$ 矩阵， m 、 n 为自然数；

所述的矩阵数据存储装置包括：

配置寄存器，包括基址寄存器，数据元素字节宽度寄存器，行跨度寄存器，列跨度寄存器，循环存取次数寄存器，循环存取使能寄存器，数据存储模式寄存

器；

地址计算单元组，由数量与物理存储器单元相同的地址计算单元组成，每一个地址计算单元具有设定的模块号，所述地址计算单元用于根据配置寄存器中的内容和模块号，按照控制器选择的数据通路，从控制器读取需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据所对应的物理地址；

地址转换模块，用于把地址计算单元组生成的各路物理地址信息来转化成为各个物理存储器单元的地址信息，输送到各个物理存储器单元；同时根据模块号为0的地址计算单元产生的物理地址信息计算需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元，并将这一信息输送到数据置换模块；

数据置换模块，用于在读取数据时，按照地址转换模块产生的需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的信息对物理存储器单元组产生的一组数据进行置换，以使其满足需要的读取数据顺序；以及在写数据时，按照地址转换模块产生的需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的信息对送入物理存储器单元组的一组数据进行置换以使其满足物理存储器中需要的存放顺序；

物理存储器单元组，由一组并行的物理存储器单元组成；

控制器，用于从程序存储器中读取指令，并对指令进行解析转化为对配置寄存器的配置信息，或是将从指令中解析出的需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移、需要存取的二维子矩阵数据存储空间中的数据块的排列方向输送到地址计算单元组，同时根据需要存取的二维子矩阵数据存储空间中的数据块的长度决定使能地址计算单元的数量，根据需要存取的二维子矩阵数据存储空间中的数据块的排列方向选择地址计算单元的数据通路，将这些信息也输送到地址计算单元组；

总线，连接多个物理存储器单元组的数据通道。

进一步，所述地址计算单元包括：

物理地址计算单元，用于根据控制器选择的数据通路和模块号，以及由控制器输入的需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子

矩阵数据存储空间的基地址的行偏移和列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据所对应的全局物理地址，所述全局物理地址在行存储模式下将直接输出，在行列交错存储模式下将输入到地址修正单元进行修正后输出；

行坐标计算单元，用于在行列交错存储模式下，根据自身的模块号和需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据在独立进行数据交错的正方形区域中所在的行坐标；

列坐标计算单元，用于在行列交错存储模式下，根据自身的模块号和需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据在独立进行数据交错的正方形区域中所在的列坐标；

地址修正单元，用于在行列交错存储模式下，根据物理地址计算单元生成的全局物理地址、行坐标计算单元生成的数据在独立进行数据交错的正方形区域中所在的行坐标、列坐标计算单元生成的数据在独立进行数据交错的正方形区域中所在的列坐标进行修正，最终生成修正后的物理地址；

输出数据选择器，用于选择是输出地址修正单元的结果还是直接输出物理地址计算单元的计算结果。

再进一步，所述数据置换模块包括：

数据控制单元，用于接受来自地址转换模块的首字节所在的物理存储器单元模块号，将其转化为对数据选择单元的选择信号，输送给读取数据选择单元或写入数据选择单元；

读取数据选择单元，它由与物理存储器单元数量相同的一组选择器组成，每一个选择器用于从多个物理存储器单元中根据数据控制单元的选择信号选择各个读出数据以便并行输出需要读取的数据块的所有数据；

写入数据选择单元，它由与物理存储器单元数量相同的一组选择器组成，每一个选择器用于根据数据控制单元的选择信号选择要写入各个物理存储器单元中的数据以便向物理存储器单元组并行输出需要写入的数据块的所有数据。

更进一步，所述物理存储器单元是单口存储器或多口存储器。

本发明的有益效果主要表现在：1、对存储空间中单位存储元素进行重新定义，

满足实际应用中所需存储空间大小不定的要求；2、存储方便，不会造成存储空间的冗余；3、能满足不同类型数据存取的要求；4、通过对逻辑空间进行有效分配，存储效率高；5、进行统一的交错模式定义，进一步提高存储效率；6、地址计算方法可以采用正负偏移，使用方便、灵活。

（四）附图说明

图 1 是阐述矩阵存储空间以及矩阵降维方法的图；

图 2 是阐述二维子矩阵数据存储空间的图；

图 3 是阐述对于一个 8×8 矩阵的两种不同的交错结果的图；

图 4 是阐述对于一个 4×4 矩阵的两种不同的交错结果的图；

图 5 是阐述矩阵数据存储装置的详细方框图；

图 6 是阐述地址计算单元的详细方框图；

图 7 是阐述一个视频宏块中的亮度信号 Y 和两个色度信号 U、V 在矩阵存储空间中的存储方式的图；

图 8 是阐述一个视频宏块中的亮度信号 Y 和两个色度信号 U、V 在经过降维处理后的逻辑空间的存储方式的图；

图 9 是阐述一个视频宏块中的色度信号 U 在物理存储器中的存储方式的图；

图 10 是阐述需要存取的二维子矩阵数据存储空间中的按行排列的数据块在逻辑空间与物理空间中的分布的图；

图 11 是阐述矩阵数据存储装置在行存储模式下进行读取一行数据的过程的图；

图 12 是阐述 H. 264 标准中需要进行预测的 4×4 亮度块以及其上边的像素组和左边的像素组的图；

图 13 是阐述需要预测的亮度块的上边的像素组和左边的像素组在逻辑空间与物理空间中的分布的图；

图 14 阐述矩阵数据存储装置在行列交错存储模式下进行读取一行数据的过程的图；

图 15 是阐述矩阵数据存储装置在行列交错存储模式下进行读取一列数据的过程的图。

（五）具体实施方式

下面结合附图对本发明作进一步描述。

实施例 1

如图 1 所示，一种矩阵数据存储方法，包含有预定的高维矩阵数据存储空间 1，它经过多级的降维处理最终可以转化为多个二维矩阵数据存储空间 2，每一个二维矩阵数据存储空间 2 可以再分解成一系列的相互逻辑独立的二维子矩阵数据存储空间 3。数据在所述高维矩阵数据存储空间 1 进行存放时采取由高维到低维的方式，在每一个二维子矩阵数据存储空间中进行存放时采取由左到右，由上到下的顺序，按照模进行存放的方式；

将所述二维子矩阵数据存储空间 3 通过数据存取模式的配置映射到多个物理存储器中，如图 2 所示，配置的内容包括：基地址、数据元素宽度、行跨度、列跨度和数据存储模式；

在存取所述二维子矩阵数据存储空间 3 的数据时，指定需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移、需要存取的二维子矩阵数据存储空间中的数据块的排列方向以及需要存取的二维子矩阵数据存储空间中的数据块的长度，从而从多个物理存储器的每一个存储器中的相应地址中读取需要存取的二维子矩阵数据存储空间中的数据块的数据；在读取数据时，从物理存储器中读出的数据要按照需要经过数据置换以使其满足需要的读取数据顺序；在写数据时，数据在送入物理存储器中需要经过数据置换以使其满足物理存储器中需要的存放顺序。

所述数据存储模式包括行存储、列存储和行列交错存储。

在行存储模式下，支持的存取的数据块的排列方向只有行排列，每次所能进行存取的最大数据块长度为不超过物理存储器的存储单元总数。

在列存储模式下，支持的存取的数据块的排列方向为列排列，每次所能进行存取的最大数据块长度为不超过物理存储器的存储单元总数。

在行列交错存储模式下，支持的存取的数据块的排列方向包括行排列和列排列，二者可同时支持。

在行列交错存储模式下，数据进行存取时要进行交错，交错包括下列步骤：

由独立进行交错的数据单元组成的正方形矩阵区域，所述正方形区域是指在行方向上包含的行跨度的个数等于这个区域所包含的行数，所述包含的行跨度的个数又称交错额度，它不能超过物理存储器的存储单元总数所包含的行跨度的个数；将所述正方形区域中的第一行数据次序保持不变；将所述正方形区域中的第二行

数据的最后一个行跨度中的数据元素转移到第一个行跨度相应位置中去，而原来的从第一个行跨度到倒数第二个行跨度中的数据元素依次向后转移一个行跨度；将所述正方形区域中的第三行数据的倒数第二个行跨度中的数据元素转移到第一个行跨度相应位置中去，最后一个行跨度中的数据元素转移到第二个行跨度相应位置中去，而原来的从第一个行跨度到倒数第三个行跨度中的数据元素依次向后转移两个行跨度；依次类推，每进行到下一行，则从行尾移到行首的数据元素增加一个，直到进行到所述正方形区域的最后一行结束；以上所述正方形区域进行交错后得出的新的各行可以进行排列组合，经过排列组合产生的每一种结果均可作为本交错方式的结果。

如图 3 所示，这里描述了对于一个 8×8 矩阵的两种不同的交错结果，如图 4 所示，这里描述了对于一个 4×4 矩阵的两种不同的交错结果。

在行存储模式下，物理地址计算过程为：设基地址为 $base$ ，需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y ，行跨度为 $xstride$ ，列跨度为 $ystride$ ；

需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址 z_i 计算式为 (1)：

$$z_i = base + (x + i) \times xstride + y \times ystride \quad (1)$$

其中， i 为各个数据相对于该数据块的首地址的距离，单位为行跨度；

将全局的物理地址 z_i 进行除以 N 和取 N 的模的运算后得到每一个存储器单元所需要的物理地址信息， N 是指所述物理存储器的模数。

在列存储模式下，物理地址计算过程为：设基地址为 $base$ ，需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y ，行跨度为 $xstride$ ，列跨度为 $ystride$ ；

需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址 z_i 计算式为 (2)：

$$z_i = base + x \times xstride + (y + i) \times ystride \quad (2)$$

其中， i 为各个数据相对于该数据块的首地址的距离，单位为列跨度；

将全局的物理地址 z_i 进行除以 N 和取 N 的模的运算后得到每一个存储器单元所需要的物理地址信息， N 是指所述物理存储器的模数。

在行列交错存储模式下，物理地址计算过程为：设基地址为 $base$ ，需要存取

的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 x 和 y , 行跨度为 $xstride$, 列跨度为 $ystride$;

行列交错存储模式的交错额度为 N , 需要存取的二维子矩阵数据存储空间中的数据块的各个数据在物理存储器中对应的物理地址的计算方法包括以下步骤:

(1)、计算未经修正的数据块中各个数据对应的全局物理地址 z_i ;

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列时, 计算式为 (1):

$$z_i = base + (x+i) \times xstride + y \times ystride \quad (1)$$

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列时, 计算式为 (2):

$$z_i = base + x \times xstride + (y+i) \times ystride \quad (2)$$

其中 i 为各个数据相对于该数据块的首地址的距离, 单位为行跨度;

(2)、同时计算数据块的各个数据在独立进行数据交错的正方形区域中所在位置的行坐标 $R(z_i)$ 和列坐标 $C(z_i)$;

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列时, 计算式为 (3)、(4):

$$R(z_i) = \begin{cases} (x+i)\%N \times xstride & (x+i \geq 0) \\ (N+(x+i)\%N) \times xstride & (x+i < 0) \end{cases} \quad (3)$$

$$C(z_i) = \begin{cases} y\%N & (y \geq 0) \\ N+y\%N & (y < 0) \end{cases} \quad (4)$$

当需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列时, 计算式为 (5)、(6):

$$R(z_i) = \begin{cases} (x\%N) \times xstride & (x \geq 0) \\ (N+x\%N) \times xstride & (x < 0) \end{cases} \quad (5)$$

$$C(z_i) = \begin{cases} (y+i)\%N & (y+i \geq 0) \\ N+(y+i)\%N & (y+i < 0) \end{cases} \quad (6)$$

(3)、按照不同的交错方式得到修正表, 依据前面求出的 $R(z_i)$ 和 $C(z_i)$ 得出数据块的各个数据在交错过程中移动的偏移量, 利用这个偏移量对 z_i 进行修正, 从而得出需要存取的二维子矩阵数据存储空间中的数据块中的各个数据在物理存储器中对应的全局物理地址;

(4)、将全局的物理地址 z_i 进行除以 N 和取 N 的模的运算后得到每一个存储器

单元所需要的物理地址信息， N 是指所述物理存储器的模数。

所述需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移，其中偏移量可以是一个正数，也可以是一个负数，数值以字节大小为单位。

实施例 2

一个由多个物理存储器单元并联组成的矩阵数据存储装置，矩阵数据存储装置的逻辑存储空间包含有预定的高维矩阵数据存储空间，所述高维矩阵数据存储空间经过多级的降维处理最终可以转化为多个二维矩阵数据存储空间，每一个二维矩阵数据存储空间可以再分解成一系列的相互逻辑独立的二维子矩阵数据存储空间。如图 5 所示，所述矩阵数据存储装置包括：

配置寄存器，包括基址寄存器，数据元素字节宽度寄存器，行跨度寄存器，列跨度寄存器，数据存储模式寄存器，交错额度寄存器。

地址计算单元组，由数量与物理存储器单元相同的地址计算单元组成，每一个地址计算单元具有一个自己的模块号，所述地址计算单元用于根据配置寄存器中的内容和模块号，按照控制器选择的数据通路，从控制器读取需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据所对应的物理地址。

地址转换模块，用于把地址计算单元组生成的各路物理地址信息来转化成为各个物理存储器单元的地址信息，输送到各个物理存储器单元；同时根据模块号为 0 的地址计算单元产生的物理地址信息计算需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的单元号，并将这一信息输送到数据置换模块。

数据置换模块，用来在读取数据时，按照地址转换模块产生的需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的信息对物理存储器单元组产生的一组数据进行置换，以使其满足需要的读取数据顺序；以及在写数据时，按照地址转换模块产生的需要存取的二维子矩阵数据存储空间中的数据块的首地址所对应的物理存储器单元的信息对送入物理存储器单元组的一组数据进行置换以使其满足物理存储器中需要的存放顺序。

物理存储器单元组，由一组并行的物理存储器单元 $50 \sim 57$ 组成，每一个物理

存储器单元的位宽均为字节。

控制器，从程序存储器中读取指令，并对指令进行解析转化为对配置寄存器的配置信息，或是将从指令中解析出的需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移、需要存取的二维子矩阵数据存储空间中的数据块的排列方向输送到地址计算单元组，同时根据需要存取的二维子矩阵数据存储空间中的数据块的长度决定使能地址计算单元的数量，根据需要存取的二维子矩阵数据存储空间中的数据块的排列方向选择地址计算单元的数据通路，将这些信息也输送到地址计算单元组；

总线，连接多个物理存储器单元组的数据通道。

如图6所示，所述地址计算单元包括一个物理地址计算单元，用于根据控制器选择的数据通路和模块号 i ，所述的模块号 i 与各个数据相对于该数据块的首地址的距离 i 取值相同；以及由控制器输入的需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移 x 和列偏移 y ，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据所对应的全局物理地址，这个物理地址在行存储模式下将直接输出，在行列交错存储模式下将输入到地址修正单元进行修正；还包括一个行坐标计算单元，用于在行列交错存储模式下，根据自身的模块号 i 和需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移 x ，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据在独立进行数据交错的正方形区域中所在的行坐标；还包括一个列坐标计算单元，用于在行列交错存储模式下，根据自身的模块号 i 和需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的列偏移 y ，生成需要存取的二维子矩阵数据存储空间中的数据块的相应位置数据在独立进行数据交错的正方形区域中所在的列坐标；还包括一个地址修正单元，用于在行列交错存储模式下，根据物理地址计算单元生成的物理地址、行坐标计算单元生成的数据在独立进行数据交错的正方形区域中所在的行坐标、列坐标计算单元生成的数据在独立进行数据交错的正方形区域中所在的列坐标进行修正，最终生成修正后的物理地址；还包括一个输出数据选择器，用来选择是输出地址修正单元的结果还是直接输出物理地址计算单元的计算结果。

所述数据置换模块包括一个数据控制单元，接受来自地址转换模块的首字节

所在的物理存储器单元模块号，将其转化为对数据选择单元的选择信号，输送给读取数据选择单元或者是写入数据选择单元；还包括一个读取数据选择单元，它由与物理存储器单元数量相同的一组选择器组成，每一个选择器用于从多个物理存储器单元中根据数据控制单元的选择信号选择各个读出数据以便并行输出需要读取的数据块的所有数据；还包括一个写入数据选择单元，它由与物理存储器单元数量相同的一组选择器组成，每一个选择器用于根据数据控制单元的选择信号选择要写入各个物理存储器单元中的数据以便向物理存储器单元组并行输出需要写入的数据块的所有数据。

实施例 3

本实施例将矩阵数据存取方法应用在视频信号处理中，通常将一帧视频数据分为若干个宏块，每一个独立的宏块由亮度信号 Y 和两个色度信号 U 和 V 按照 4: 2: 0 的比例组成。其中，亮度信号为一个 16×16 的矩阵，两个色度信号分别为 8×8 的矩阵。利用本发明矩阵存储装置可以有效的实现对这种宏块数据进行存储。

将每一个独立宏块看作是一个四维矩阵数据存储空间，其中一维为时间，如图 7 所示，在沿着时间 t 轴依次排列着一组相互独立的矩阵数据存储空间 1，每一个三维数据空间 1 由三个相互独立的二维矩阵数据存储空间 2 组成，每一个二维矩阵数据存储空间 2 存放着 Y、U、V 中的一种信息，其中多个 Y 数据块或者多个 U 数据块或者多个 V 数据块可组成一个二维子矩阵数据存储空间。

把这个四维矩阵数据存储空间通过降维处理，转化为多个二维矩阵数据存储空间 2 组成的一个总二维矩阵数据存储空间，即一个平面逻辑空间，如图 8 所示，在每个二维矩阵数据存储空间 2 中存放着 Y、U、V 中的一种信息。在规划二维矩阵数据存储空间 2 的空间大小时可以将用来存放 Y 的二维矩阵数据存储空间的长和宽设计为用来存放 U、V 的二维矩阵数据存储空间的长和宽的两倍，通过这样的一种方法可以使逻辑空间达到充分的饱和。

在向物理存储器中进行存放时，存放顺序为在逻辑空间的同一个二维子矩阵数据存储空间中，先将第一行的数据从左到右依次存入物理存储器，然后转到下一行的最左端，再按照从左到右的顺序依次存放，如图 9 所示，以色度信号 U 的 8×8 数据块为例，先把所有同一行的 U 数据块中的第一行的数据按照箭头方向存入物理存储器，然后按照箭头方向存储所有同一行的 U 数据块中的第二行的数据，依次类推，直到 8 行数据存储完毕。

第一、行储模式下的矩阵装置的数据存取地址计算

在信号处理的快速傅立叶变换、离散余弦变换等运算中，都会遇到同时取一行数据的运算，下面将以一个从存储器中同时取一行 4 个数据至寄存器中的例子来说明本矩阵装置在行存储模式下的使用方法。

设行跨度为 1，列跨度为 16，数据元素宽度为 1，需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 3 和 1，需要存取的二维子矩阵数据存储空间中的数据块的排列方向为行排列，需要存取的二维子矩阵数据存储空间中的数据块的长度为 4，则这样一个数据块在逻辑空间中所处的位置如图 10a 所示。

设物理存储器单元组由 8 个并行的物理存储器单元 50~57 组成，设基地址为 4，则需要存取的二维子矩阵数据存储空间中的数据块在物理空间所处的位置如图 10b 所示。

用户在进行数据存取之前需要对矩阵存取装置的配置寄存器进行配置，方式是用户通过指令将配置信息输送到控制器，然后由控制器对指令进行解析用得出的配置信息完成对各个配置寄存器的配置。配置后在下一次配置之前的所有的数据存取都依照本次配置执行。

在这个例子中配置完成后各个配置寄存器值分别为数据存储模式为行存储模式，行跨度寄存器为 1，列跨度寄存器为 16，数据元素宽度寄存器为 1。

然后用户就可以通过指令来完成数据的存取，在本例中，控制器通过对存取指令的解析可以得到需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 4 和 1，需要存取的二维子矩阵数据存储空间中的数据块的排列方向只可为行排列，需要存取的二维子矩阵数据存储空间中的数据块的长度为 4。控制器将以上信息输送到各个地址计算单元进行物理地址的计算，同时生成地址计算单元的数据通路选择信号以控制其计算的进行。

地址计算单元组中的前四个地址计算单元被控制器使能，开始执行物理地址计算操作，如图 6 所示，地址计算单元得到控制器送来的地址信息和数据通路选择信号参照公式 (1) 完成以下运算：

物理地址计算单元执行：

$$z_0 = 4 + (3 + 0) \times 1 + 1 \times 16 = 23$$

$$z_1 = 4 + (3+1) \times 1 + 1 \times 16 = 24$$

$$z_2 = 4 + (3+2) \times 1 + 1 \times 16 = 25$$

$$z_3 = 4 + (3+3) \times 1 + 1 \times 16 = 26$$

其中下标 $0 \sim 4$ 代表地址计算单元的模块号, 每一个物理地址计算单元只执行自己对应模块号的运算。

由于没有进行行列交错, 所以行坐标计算单元和列坐标计算单元在这里不进行工作, 数据选择器 6 选通信号为 1, 将物理地址计算单元的计算结果直接输出。

如图 11 所示, 地址计算单元的计算结果输送到地址转换模块。地址转换模块将计算结果进行除以 8 和模 8 的处理, 这里的 8 是指物理存储器单元组由 8 个并行的物理存储器单元组成, 得到下表 2:

	除以 8	模 8	意义
$z_0 = 23$	2	7	输出物理存储器单元 7 的地址为 2 的数据
$z_1 = 24$	3	0	输出物理存储器单元 0 的地址为 3 的数据
$z_2 = 25$	3	1	输出物理存储器单元 1 的地址为 3 的数据
$z_3 = 26$	3	2	输出物理存储器单元 2 的地址为 3 的数据

表 2

如图 11 所示, 地址转换模块将转换结果分别送到相应的物理存储器单元 $50 \sim 57$, 物理存储器单元根据地址信息输出数据到地址置换模块。

与此同时, 地址转换模块将 z_0 所对应的物理存储器单元号 7 输送到地址置换模块, 地址置换模块中的数据控制单元将物理存储器单元号 7 转化成为对读取数据选择单元的选择器的选择信号送往读取数据选择单元, 读取数据选择单元通过这些选择信号将由物理存储单元输入的数据进行转换, 将物理存储器单元 57 的数据调整到输出的第一个字节, 如图 11 所示, 然后把读出的一行 4 个数据同过总线送往寄存器。

至此, 这次读数据的操作全部完成。

第二、列存储模式下的矩阵装置的数据存取地址计算

其过程与行存储模式下基本相同, 地址计算公式参照 (2)。

第三、行列交错存储模式下的矩阵装置的数据存取地址计算

以下以 H. 264 标准中帧内预测中 4×4 亮度块的预测为例来说明本矩阵装置在行列交错存储模式下的使用方法。

图 12 显示了需要进行预测的 4×4 亮度块 7 以及其上边的像素组 8 和左边的像素组 9，将要被预测的亮度块 7 可通过像素组 8 和像素组 9 中的数据进行预测，这里就需要既要同时存取处于同一行的像素组 8 又要能同时存取处于同一列的像素组 9。使用本发明的矩阵装置可以对这种存取操作进行加速。

设行跨度为 1，列跨度为 16，数据元素宽度为 1，图 12 中点 M 相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 3 和 1，相应地，点 A 相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 4 和 1，则这样一个数据块在逻辑空间中所处的位置如图 13a 所示。

设物理存储器单元组由 8 个并行的物理存储器单元 50~57 组成，设基地址为 0，则需要存取的二维子矩阵数据存储空间中的数据块在物理空间所处的位置如图 13b 所示，从图中可以看出，由于在行列交错存储模式下，数据从逻辑空间转换到物理空间中要进行交错，本例中交错额度为 8。因此，在图 13b 中，在逻辑空间中处于同一列的 M、I、J、K、L 五个像素点已经被错开分布在五个不同的列中。

用户在进行数据存取之前需要对矩阵存取装置的配置寄存器进行配置，方式是用户通过指令将配置信息输送到控制器，然后由控制器对指令进行解析用得出的配置信息完成对各个配置寄存器的配置。配置后在下一次配置之前的所有的数据存取都依照本次配置执行。

在这个例子中配置完成后各个配置寄存器值分别为数据存储模式为行列交错存储模式，行跨度寄存器为 1，列跨度寄存器为 16，数据元素宽度寄存器为 1，交错额度寄存器为 8。

然后用户就可以通过指令来完成数据的存取。

首先说明对于像素组 8 的读取方式，控制器通过对存取指令的解析可以得到需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 4 和 0，需要存取的二维子矩阵数据存储空间中的数据块的排列方向只可为行排列，需要存取的二维子矩阵数据存储空间中的数据块的长度为 8。控制器将以上信息输送到各个地址计算单元进行物理地址的计算，同时生成地址计算单元的数据通路选择信号以控制其计算的进行。

地址计算单元组中的八个地址计算单元均被控制器使能，开始执行物理地址计算操作，如图 6 所示，地址计算单元得到控制器送来的地址信息和数据通路选择信号完成以下运算：

物理地址计算单元参照公式 (1) 执行：

$$z_0 = 0 + (4 + 0) \times 1 + 1 \times 16 = 20$$

$$z_1 = 0 + (4+1) \times 1 + 1 \times 16 = 21$$

$$z_2 = 0 + (4+2) \times 1 + 1 \times 16 = 22$$

$$z_3 = 0 + (4+3) \times 1 + 1 \times 16 = 23$$

$$z_4 = 0 + (4+4) \times 1 + 1 \times 16 = 24$$

$$z_5 = 0 + (4+5) \times 1 + 1 \times 16 = 25$$

$$z_6 = 0 + (4+6) \times 1 + 1 \times 16 = 26$$

$$z_7 = 0 + (4+7) \times 1 + 1 \times 16 = 27$$

其中下标 0~7 代表地址计算单元的模块号, 每一个物理地址计算单元只执行自己对应模块号的运算。

行坐标计算单元参照公式 (3) 执行:

$$R(z_0) = (4+0) \% 8 \times 1 = 4$$

$$R(z_1) = (4+1) \% 8 \times 1 = 5$$

$$R(z_2) = (4+2) \% 8 \times 1 = 6$$

$$R(z_3) = (4+3) \% 8 \times 1 = 7$$

$$R(z_4) = (4+4) \% 8 \times 1 = 0$$

$$R(z_5) = (4+5) \% 8 \times 1 = 1$$

$$R(z_6) = (4+6) \% 8 \times 1 = 2$$

$$R(z_7) = (4+7) \% 8 \times 1 = 3$$

其中下标 0~7 代表地址计算单元的模块号, 每一个行坐标计算单元只执行自己对应模块号的运算。

列坐标计算单元由于需要存取的二维子矩阵数据存储空间中的数据块的排列方向只可为行排列, 因此所有的列坐标计算单元同时参照公式 (4) 执行

$$C(z) = 0 \% 8 = 1$$

物理地址计算单元、行坐标计算单元、列坐标计算单元的结果输送到地址修正单元进行地址的修正。表 3 为在交错额度 M 为 8, 行跨度 X 为 1 的情况下地址修正单元对物理地址计算单元生成的物理地址进行修正表, 表 3 是表 1 的一个具体实施例。地址的修正参照表 3 进行:

R C	0	1	2	3	4	5	6	7
0	+0	+0	+0	+0	+0	+0	+0	+0

1	+1	+1	+1	+1	+1	+1	+1	-7
2	+2	+2	+2	+2	+2	+2	-6	-6
3	+3	+3	+3	+3	+3	-5	-5	-5
4	+4	+4	+4	+4	-4	-4	-4	-4
5	+5	+5	+5	-3	-3	-3	-3	-3
6	+6	+6	-2	-2	-2	-2	-2	-2
7	+7	-1	-1	-1	-1	-1	-1	-1

表 3

例如对于像素 B 物理地址的修正包括如下步骤:根据像素 B 在独立进行交错的数据单元组成的正方形矩阵区域中所处的行坐标 5 和列坐标 1 查表得到+1,即在物理地址计算单元得到的 $z_1 = 21$ 的基础上再加 1 就得到了修正后的像素 B 所对应的物理地址 22。

按照同样的方法,各个地址修正单元分别将修正后的像素 $A \sim H$ 的物理地址输送到相应的数据选择器 6,如图 6 所示,数据选择器 6 选通信号为 0,将地址修正单元的结果输出。

如图 14 所示,地址计算单元的计算结果输送到地址转换模块。地址转换模块将计算结果进行除以 8 和模 8 的处理,这里的 8 是指物理存储器单元组由 8 个并行的物理存储器单元组成,得到下表 4:

	除以 8	模 8	意义
$z_0 = 21$	2	5	输出物理存储器单元 5 的地址为 2 的数据
$z_1 = 22$	2	6	输出物理存储器单元 6 的地址为 2 的数据
$z_2 = 23$	2	7	输出物理存储器单元 7 的地址为 2 的数据
$z_3 = 16$	2	0	输出物理存储器单元 0 的地址为 2 的数据
$z_4 = 25$	3	1	输出物理存储器单元 1 的地址为 3 的数据
$z_5 = 26$	3	2	输出物理存储器单元 2 的地址为 3 的数据
$z_6 = 27$	3	3	输出物理存储器单元 3 的地址为 3 的数据
$z_7 = 28$	3	4	输出物理存储器单元 4 的地址为 3 的数据

表 4

如图 14 所示,地址转换模块将转换结果分别送到相应的物理存储器单元 $50 \sim 57$,物理存储器单元根据地址信息输出数据到地址置换模块。

与此同时，地址转换模块将 z_0 所对应的物理存储器单元号 5 输送到地址置换模块，地址置换模块中的数据控制单元将物理存储器单元号 5 转化成为对读取数据选择单元的选择器的选择信号送往读取数据选择单元，读取数据选择单元通过这些选择信号将由物理存储单元输入的数据进行转换，将物理存储器单元 55 的数据调整到输出的第一个字节，如图 14 所示，然后把读出的一行 8 个数据同过总线送往寄存器。

再来说明对于像素组 8 的读取方式，控制器通过对存取指令的解析可以得到需要存取的二维子矩阵数据存储空间中的数据块的首地址相对于二维子矩阵数据存储空间的基地址的行偏移和列偏移分别为 3 和 1，需要存取的二维子矩阵数据存储空间中的数据块的排列方向为列排列，需要存取的二维子矩阵数据存储空间中的数据块的长度为 5。控制器将以上信息输送到各个地址计算单元进行物理地址的计算，同时生成地址计算单元的数据通路选择信号以控制其计算的进行。

地址计算单元组中的前五个地址计算单元均被控制器使能，开始执行物理地址计算操作，如图 6 所示，地址计算单元得到控制器送来的地址信息和数据通路选择信号完成以下运算：

物理地址计算单元参照公式 (2) 执行：

$$z_0 = 0 + 3 \times 1 + (1 + 0) \times 16 = 19$$

$$z_1 = 0 + 3 \times 1 + (1 + 1) \times 16 = 35$$

$$z_2 = 0 + 3 \times 1 + (1 + 2) \times 16 = 51$$

$$z_3 = 0 + 3 \times 1 + (1 + 3) \times 16 = 67$$

$$z_4 = 0 + 3 \times 1 + (1 + 4) \times 16 = 83$$

其中下标 0~5 代表地址计算单元的模块号，每一个物理地址计算单元只执行自己对应模块号的运算。

行坐标计算单元由于需要存取的二维子矩阵数据存储空间中的数据块的排列方向只可为列排列，因此所有的行坐标计算单元同时参照公式 (5) 执行：

$$R(z) = (3 \% 8) \times 1 = 3$$

列坐标计算单元参照公式 (6) 执行：

$$C(z_0) = (1 + 0) \% 8 = 1$$

$$C(z_1) = (1 + 1) \% 8 = 2$$

$$C(z_2) = (1 + 2) \% 8 = 3$$

$$C(z_3) = (1+3)\%8 = 4$$

$$C(z_4) = (1+4)\%8 = 5$$

物理地址计算单元、行坐标计算单元、列坐标计算单元的结果输送到地址修正单元进行地址的修正。地址修正参照表 3，例如对于像素 I 物理地址的修正包括如下步骤：根据像素 I 在独立进行交错的数据单元组成的正方形矩阵区域中所处的行坐标 3 和列坐标 2 查表得到+2，即在物理地址计算单元得到的 $z_1 = 35$ 的基础上再加 2 就得到了修正后的像素 I 所对应的物理地址 37。

按照同样的方法，各个地址修正单元分别将修正后的像素 M、J、K、L 的物理地址输送到相应的数据选择器 6，如图 6 所示，数据选择器 6 选通信号为 0，将地址修正单元的结果输出。

如图 15 所示，地址计算单元的计算结果输送到地址转换模块。地址转换模块将计算结果进行除以 8 和模 8 的处理，这里的 8 是指物理存储器单元组由 8 个并行的物理存储器单元组成，得到下表 5：

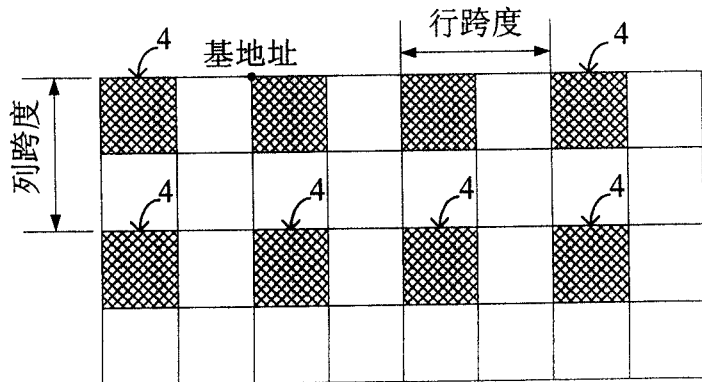
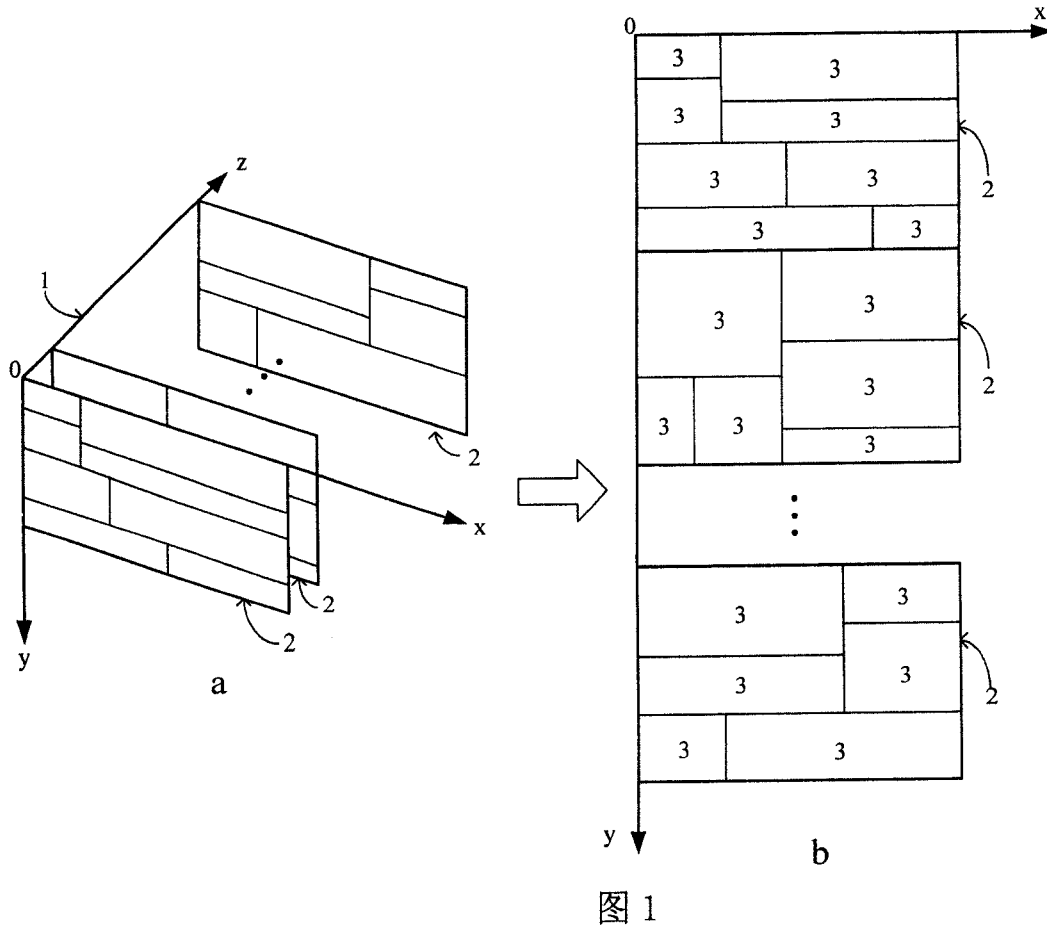
	除以 8	模 8	意义
$z_0 = 20$	2	4	输出物理存储器单元 4 的地址为 2 的数据
$z_1 = 37$	4	5	输出物理存储器单元 5 的地址为 4 的数据
$z_2 = 54$	6	6	输出物理存储器单元 6 的地址为 6 的数据
$z_3 = 71$	8	7	输出物理存储器单元 7 的地址为 8 的数据
$z_4 = 80$	10	0	输出物理存储器单元 0 的地址为 10 的数据

表 5

如图 15 所示，地址转换模块将转换结果分别送到相应的物理存储器单元 50~57，物理存储器单元根据地址信息输出数据到地址置换模块。

与此同时，地址转换模块将 z_0 所对应的物理存储器单元号 4 输送到地址置换模块，地址置换模块中的数据控制单元将物理存储器单元号 4 转化成为对读取数据选择单元的选择器的选择信号送往读取数据选择单元，读取数据选择单元通过这些选择信号将由物理存储单元输入的数据进行转换，将物理存储器单元 54 的数据调整到输出的第一个字节，如图 14 所示，然后把读出的一行 5 个数据同过总线送往寄存器。

至此，这次读数据的操作全部完成。



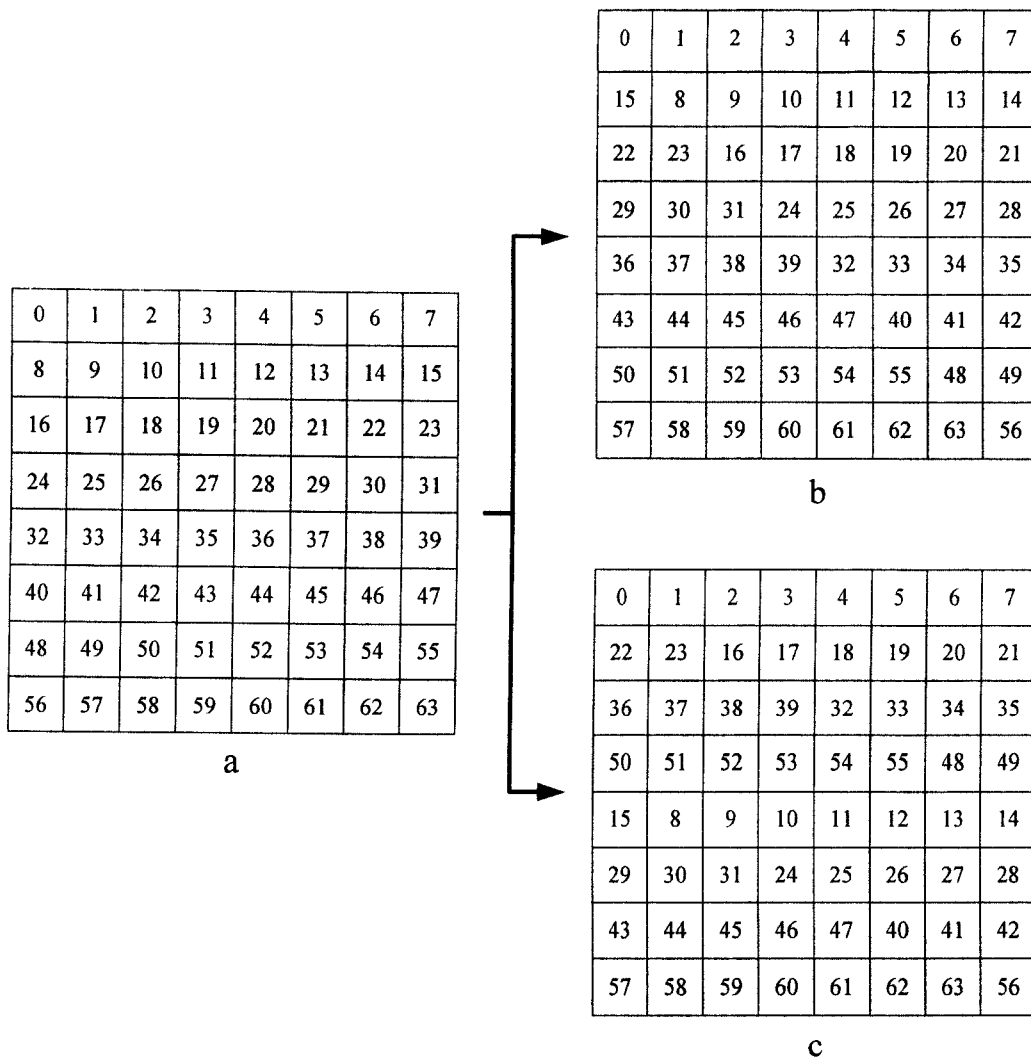


图 3

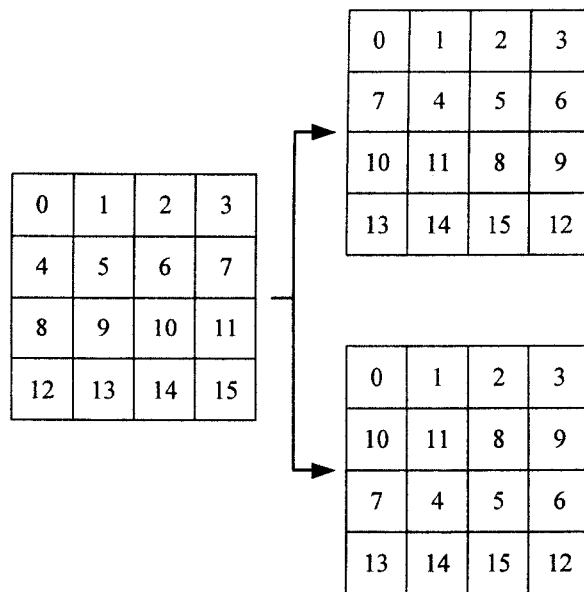


图 4

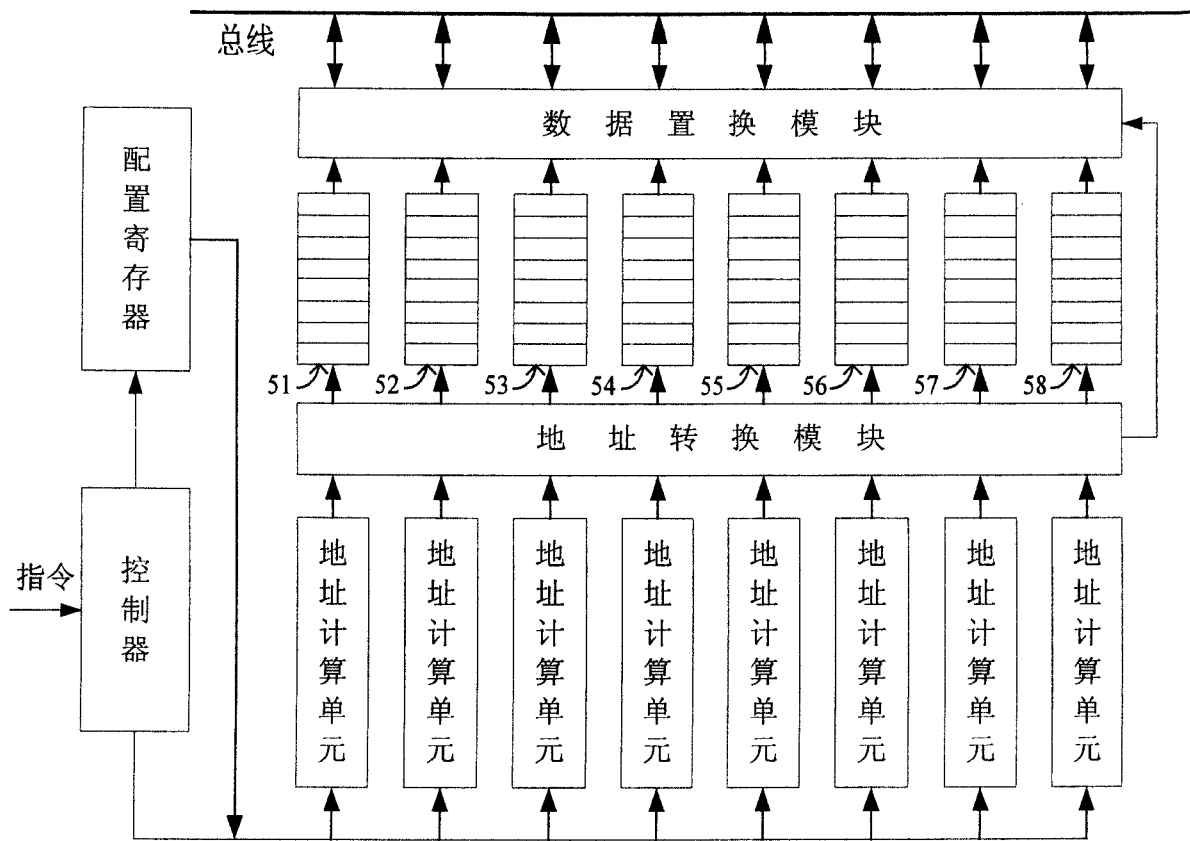


图 5

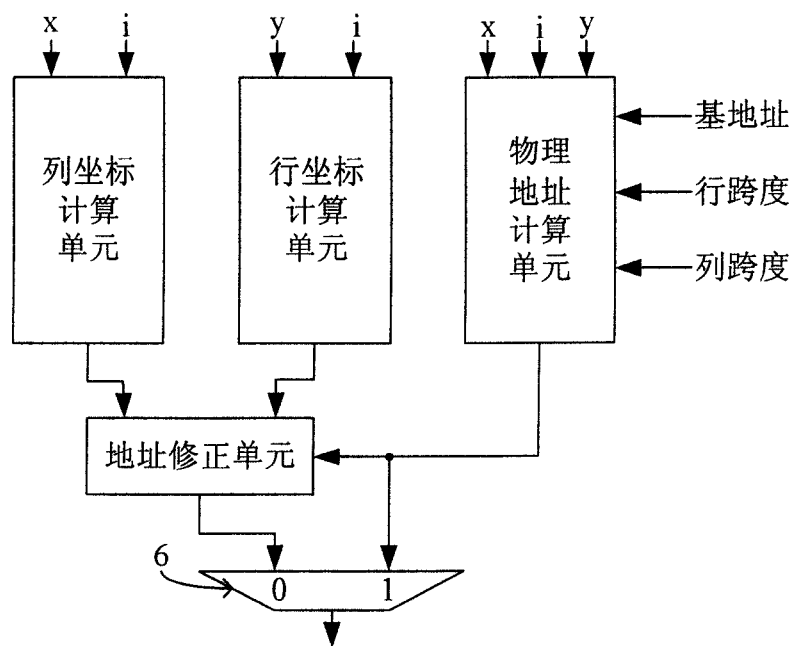


图 6

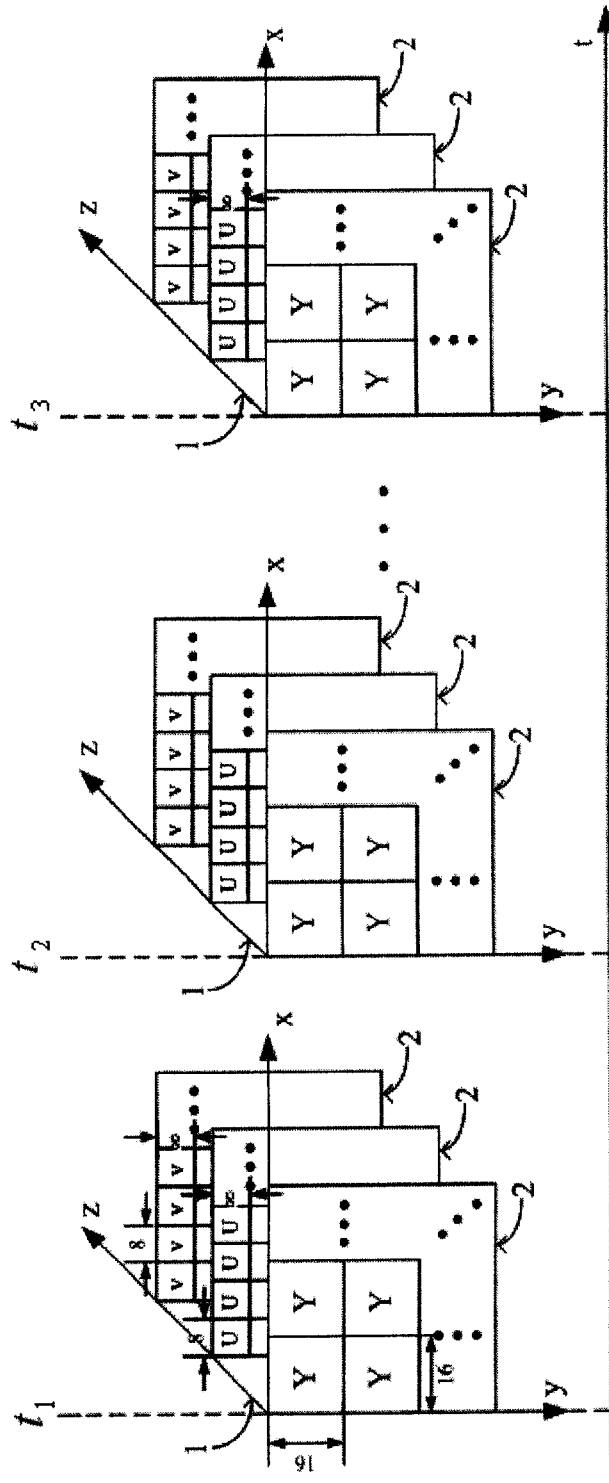


图 7

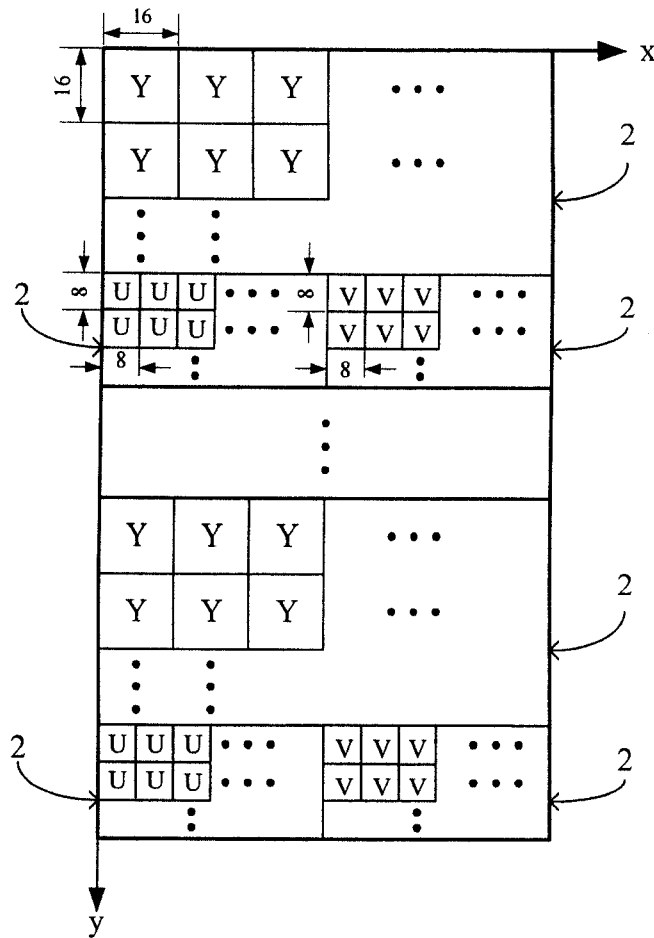


图 8

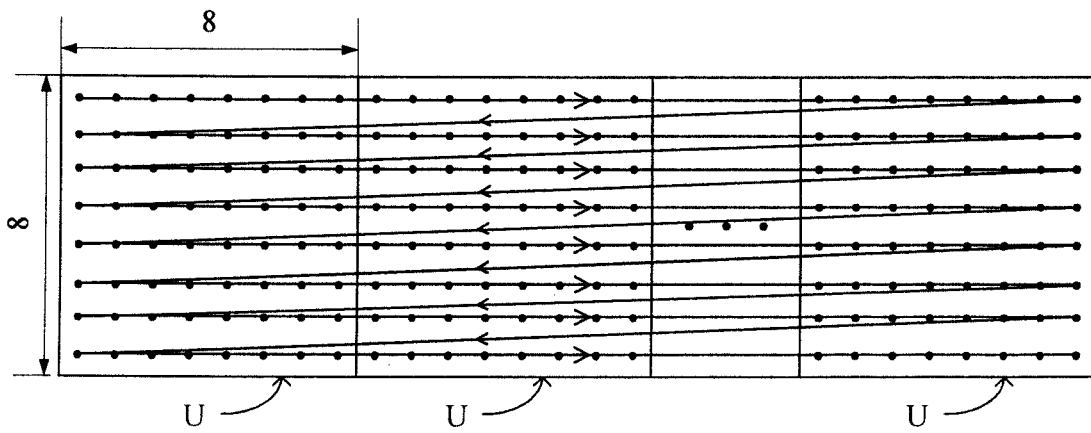


图 9

基地址

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47

a

基地址

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47

b

图 10

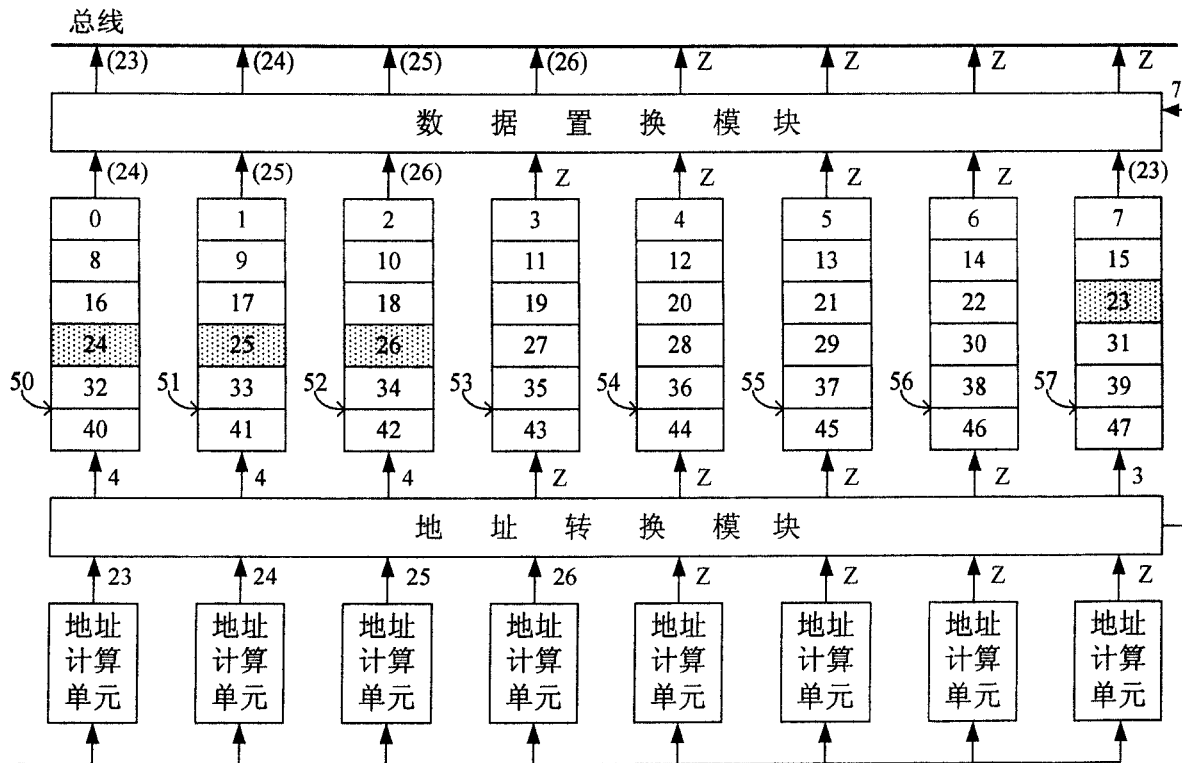


图 11

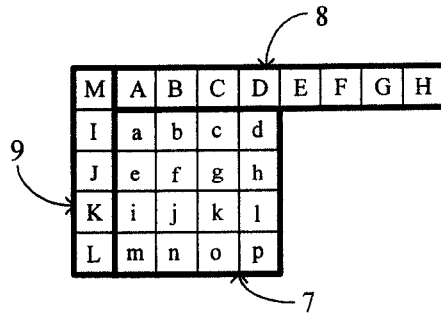


图 12

基地址

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	M	A	B	C	D	E	F	G	H	28	29	30	31
32	33	34	I	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	J	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	K	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	L	84	85	86	87	88	89	90	91	92	93	94	95

a

基地址

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
D	16	17	18	M	A	B	C
31	E	F	G	H	28	29	30
38	39	32	33	34	I	36	37
46	47	40	41	42	43	44	45
53	54	55	48	49	50	J	52
61	62	63	56	57	58	59	60
68	69	70	71	64	65	66	K
76	77	78	79	72	73	74	75
L	84	85	86	87	80	81	82
91	92	93	94	95	88	89	90

b

图 13

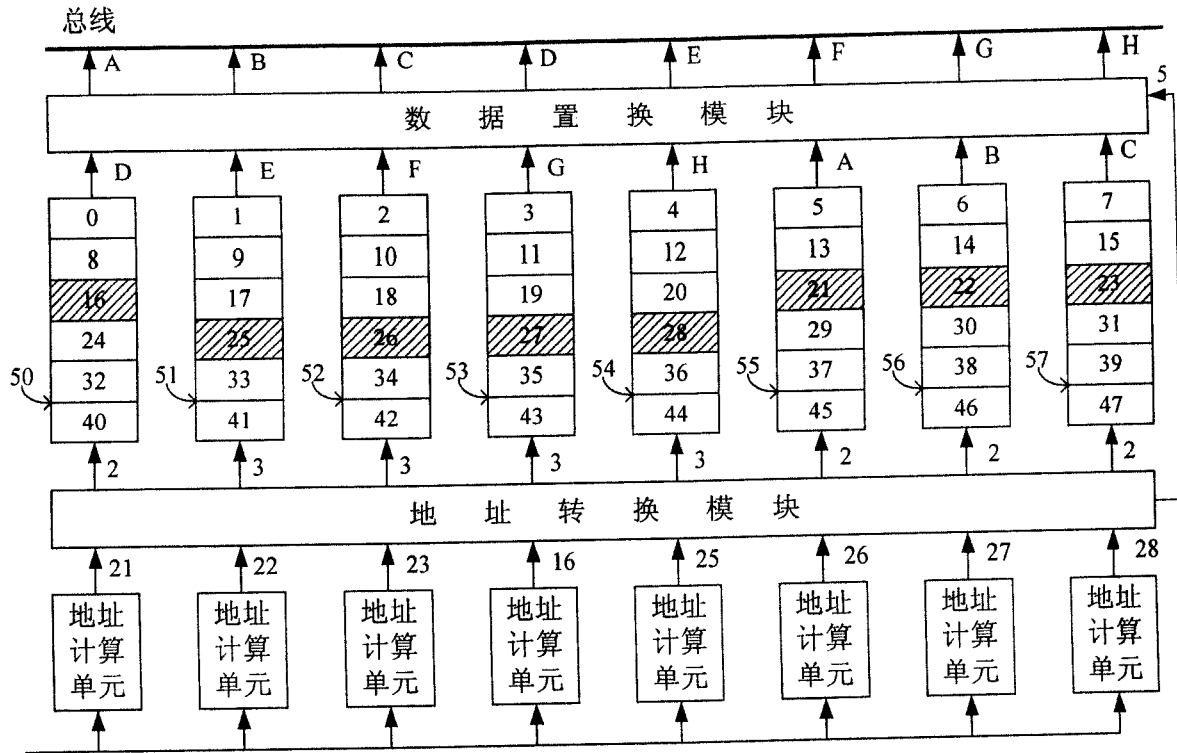


图 14

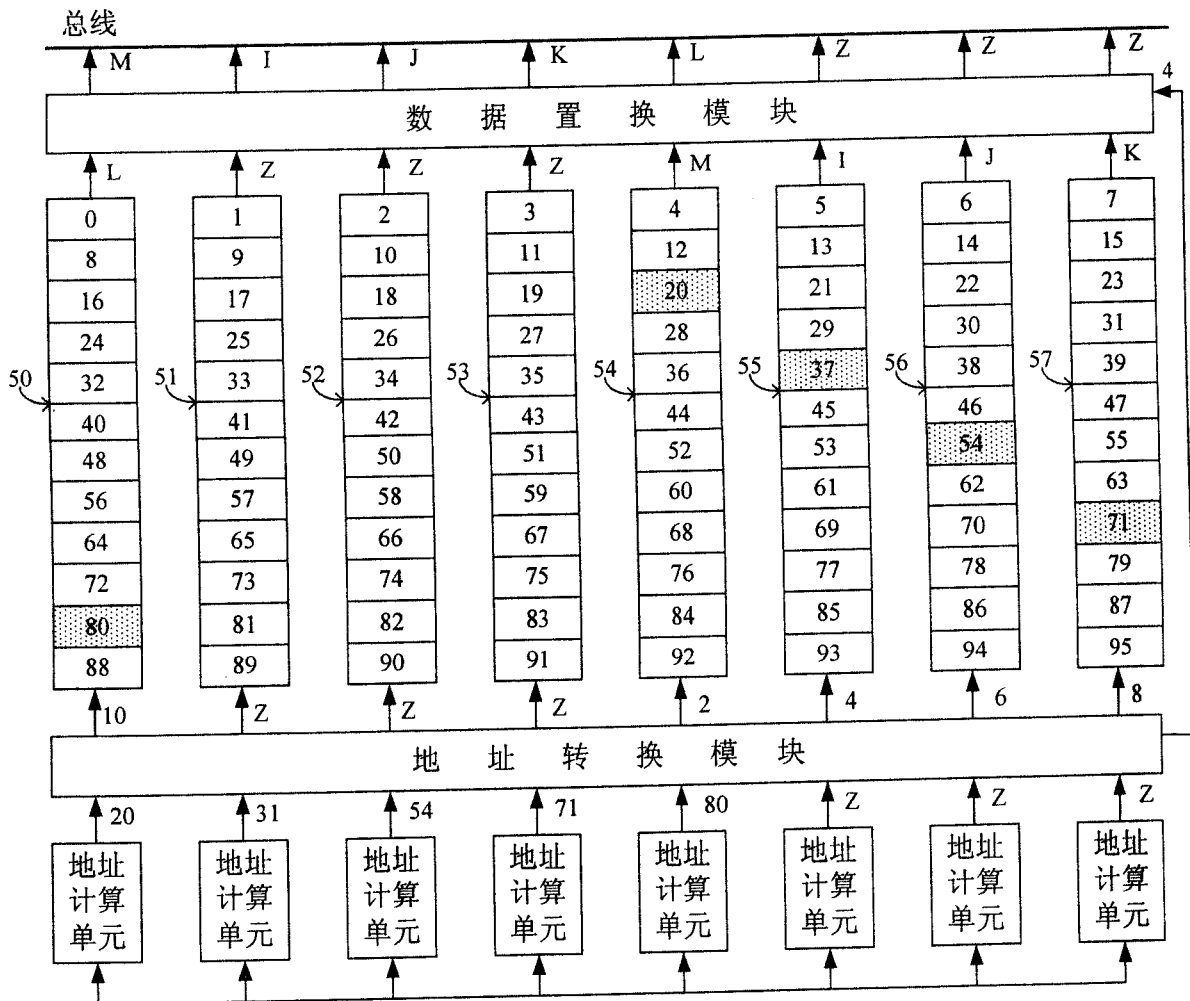


图 15