(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 106970896 A
(43)申请公布日 2017.07.21

(21)申请号 201710201589.5

(22)申请日 2017.03.30

(71)申请人 中国人民解放军国防科学技术大学
地址 410073 湖南省长沙市砚瓦池正街47号

(72)发明人 郭阳 张军阳 刘仲 扈啸 王慧丽 胡敏慧 王子聪

(74)专利代理机构 湖南兆弘专利事务所(普通合伙) 43008
代理人 周长清

(51)Int.Cl.
*G06F 17/16*(2006.01)
*G06F 17/15*(2006.01)

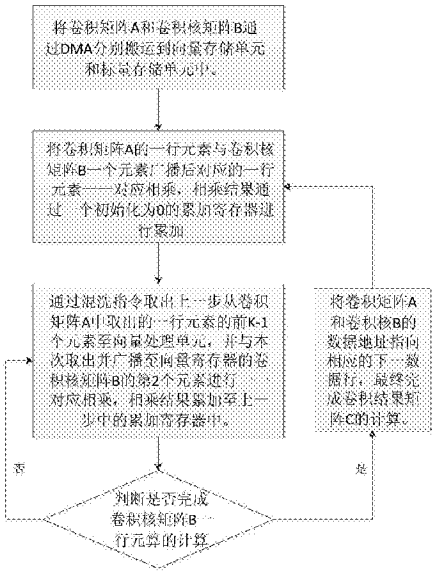
权利要求书1页 说明书5页 附图9页

(54)发明名称

面向向量处理器的二维矩阵卷积的向量化实现方法

(57)摘要

一种面向向量处理器的二维矩阵卷积的向量化实现方法，步骤为：S1:通过DMA控制器将卷积矩阵A和卷积核矩阵B分别搬运到向量存储单元和标量存储单元；S2:将卷积矩阵A的一行元素和卷积核矩阵B的一个元素广播后对应的一行元素一一对应相乘，结果进行累加；S3:通过混洗指令取出卷积矩阵A中取出的一行元素的前K-1个元素至向量处理单元，与本次取出并广播至向量处理单元的卷积核矩阵B的第二个元素进行一一对应相乘，结果累加；S4:判断是否完成一行元素的计算；S5:将两个矩阵的数据地址指向下一数据行，完成矩阵C的第一行元素的计算，通过循环完成整个矩阵C的计算。本发明具有原理简单、操作方便、能大大增加算法并行度并提高计算效率等优点。




CN 106970896 A

1.一种面向向量处理器的二维矩阵卷积的向量化实现方法，其特征在于，步骤为：

S1:输入卷积矩阵A和卷积核矩阵B;通过DMA控制器将卷积矩阵A和卷积核矩阵B分别搬运到向量存储单元和标量存储单元；

S2:将卷积矩阵A的一行元素和卷积核矩阵B的一个元素广播后对应的一行元素一一对应相乘，相乘的结果通过一个初始化为0的累加寄存器进行累加；

S3:通过混洗指令取出步骤S2中从卷积矩阵A中取出的一行元素的前K-1个元素至向量处理单元，并与本次取出并广播至向量处理单元的卷积核矩阵B的第二个元素进行一一对应相乘，结果累加至步骤S2中的累加寄存器中;其中K为并行处理单元的个数；

S4:判断是否完成卷积核矩阵B一行元素的计算，若没有完成，返回步骤S3，若完成则继续步骤S5；

S5:将卷积矩阵A和卷积核矩阵B的数据地址指向相应的下一数据行，最终完成卷积结果矩阵C的第一行元素的计算，通过循环步骤S2～步骤S5最终完成整个卷积结果矩阵C的计算。

2.根据权利要求1所述的面向向量处理器的二维矩阵卷积的向量化实现方法，其特征在于，在上述步骤S1的搬运过程中，通过DMA控制器的配置，卷积矩阵A的每一行组织成一个数据帧，卷积核矩阵B的每一个元素组织成一个数据帧，整个卷积矩阵A共可分成p个数据帧，当所述数据帧的元素个数不等于向量处理器中并行处理单元的个数K的倍数时，在数据帧尾部补0使得每个数据帧的元素个数等于并行处理单元的个数K的倍数。

3.根据权利要求1或2所述的面向向量处理器的二维矩阵卷积的向量化实现方法，其特征在于，对于m×m的卷积矩阵A卷积n×n的卷积核矩阵B的运算，得到(m-n+1)×(m-n+1)的卷积结果矩阵C，且m≧n，结果矩阵C的每一个元素都是由卷积核矩阵B(n×n)的所有元素在卷积矩阵A(m×m)上与卷积核矩阵B同样大小的区域进行点积运算并累加求和的结果。

# 面向向量处理器的二维矩阵卷积的向量化实现方法

## 技术领域

[0001] 本发明主要到涉及到向量处理器以及数据处理领域，特指一种面向向量处理器的二维矩阵卷积的向量化实现方法。

## 背景技术

[0002] 在许多科学计算任务和实际应用中都会涉及到二维矩阵卷积运算，如图像处理、机器学习、深度学习、人工神经网络及通信系统中的信号编解码等，对于不同规模的二维矩阵卷积计算任务，由于涉及到大量的数据访存和乘加运算，需要占用大量的访存和计算时间，并行效率比较差。如何利用向量处理器的多功能部件、多运算单元的特点来简单而高效的实现不同规模的二维矩阵卷积的向量化运算一直是业界的研究热点。

[0003] 在传统的标量处理器上，研究人员已经提出了一些有效的二维矩阵卷积的实现方法，以加速二维矩阵卷积的运算。但是，随着人工智能、深度学习、人工神经网络等学科的兴起，在图像识别、语音识别、文本识别及其他目标识别领域中二维矩阵卷积占有着越来越重要的位置，尤其是在当前目标识别率最高的卷积神经网络模型中，如何加速二维矩阵的卷积运算成为当前研究的热点和难点。随着高密集、实时运算应用的不断涌现，单芯片难以满足这类应用的高密度实时计算需求，因此，向量处理器得到了广泛应用。

[0004] 如图1所示，为一个向量处理器的典型结构，其具有处理器、程序存储器和数据存储器(两者均可以为任意的可访问存储器，包括外部高速缓冲存储器、外部RAM等)。向量处理器的处理器分为标量处理部件和向量处理部件两个部分，通常向量处理部件内有K个并行处理单元(PE)，这些处理单元都有各自的运算部件和寄存器，处理单元间能通过规约指令进行数据交互，如并行处理单元之间的数据相乘、比较等。标量处理单元主要负责流控和逻辑判断指令的处理，而向量处理单元主要负责密集型的数据计算。向量处理单元运算所用的数据由向量数据存储单元提供。一般地，如图2所示，向量数据存储单元的BANK(存储体)的个数与向量处理单元的处理单元个数K是一致的。

## 发明内容

[0005] 本发明要解决的技术问题就在于：针对现有技术存在的技术问题，本发明提供一种原理简单、操作方便、能大大增加算法并行度并提高计算效率的面向向量处理器的二维矩阵卷积的向量化实现方法。

[0006] 为解决上述技术问题，本发明采用以下技术方案：

[0007] 一种面向向量处理器的二维矩阵卷积的向量化实现方法，其步骤为：

[0008] S1:输入卷积矩阵A和卷积核矩阵B;通过DMA控制器将卷积矩阵A和卷积核矩阵B分别搬运到向量存储单元和标量存储单元；

[0009] S2:将卷积矩阵A的一行元素和卷积核矩阵B的一个元素广播后对应的一行元素一一对应相乘，相乘的结果通过一个初始化为0的累加寄存器进行累加；

[0010] S3:通过混洗指令取出步骤S2中从卷积矩阵A中取出的一行元素的前K-1个元素至

向量处理单元，并与本次取出并广播至向量处理单元的卷积核矩阵B的第二个元素进行一一对应相乘，结果累加至步骤S2中的累加寄存器中；其中K为并行处理单元的个数；

[0011] S4:判断是否完成卷积核矩阵B一行元素的计算，若没有完成，返回步骤S3，若完成则继续步骤S5；

[0012] S5:将卷积矩阵A和卷积核矩阵B的数据地址指向相应的下一数据行，最终完成卷积结果矩阵C的第一行元素的计算，通过循环步骤S2～步骤S5最终完成整个卷积结果矩阵C的计算。

[0013] 作为本发明的进一步改进：在上述步骤S1的搬运过程中，通过DMA控制器的配置，卷积矩阵A的每一行组织成一个数据帧，卷积核矩阵B的每一个元素组织成一个数据帧，整个卷积矩阵A共可分成p个数据帧，当所述数据帧的元素个数不等于向量处理器中并行处理单元的个数K的倍数时，在数据帧尾部补0使得每个数据帧的元素个数等于并行处理单元的个数K的倍数。

[0014] 作为本发明的进一步改进：对于m×m的卷积矩阵A卷积n×n的卷积核矩阵B的运算，得到(m-n+1)×(m-n+1)的卷积结果矩阵C，且m≧n，结果矩阵C的每一个元素都是由卷积核矩阵B(n×n)的所有元素在卷积矩阵A(m×m)上与卷积核矩阵B同样大小的区域进行点积运算并累加求和的结果。

[0015] 与现有技术相比，本发明的优点在于：本发明的面向向量处理器的二维矩阵卷积的向量化实现方法，通过DMA完成卷积矩阵A和卷积核矩阵B分别搬移至向量存储体和标量存储体，同时还充分利用向量处理器中的向量部件多个并行处理单元能够同时进行相同运算操作的特点来进行大量的同类型操作，通过配置特殊的混洗模式，大量复用每次取到的卷积矩阵A的数据，从而大大降低卷积矩阵A的访存量，进而大幅度提高二维矩阵卷积的计算效率，且步骤简单，易于实现。

## 附图说明

[0016] 图1是典型的向量处理器结构示意图。

[0017] 图2是向量处理器中的向量数据存储单元的结构示意图。

[0018] 图3是本发明的总流程示意图。

[0019] 图4是本发明中卷积矩阵A在向量数据存储单元中的加载形式及卷积核矩阵B的元素标量广播至向量寄存器的示意图。

[0020] 图5是本发明在具体应用实例2中卷积矩阵A(16×16)在向量存储单元中的存放形式示意图。

[0021] 图6是本发明在具体应用中配置的混洗模式1的实施示意图。

[0022] 图7是本发明在具体应用中配置的混洗模式2的实施示意图。

[0023] 图8是本发明在具体应用实例2中完成卷积结果矩阵C一行元素的实现步骤示意图。

[0024] 图9是本发明在具体应用实例3中卷积矩阵A在向量数据存储单元中的存放形式示意图。

[0025] 图10是本发明在具体应用实例3中完成卷积结果矩阵C一行元素的实现步骤示意图。

## 具体实施方式

[0026] 以下将结合说明书附图和具体实施例对本发明做进一步详细说明。

[0027] 如图3和图4所示，本发明的面向向量处理器的二维矩阵卷积的向量化实现方法，其步骤为：

[0028] S1:输入卷积矩阵A和卷积核矩阵B;通过DMA控制器将卷积矩阵A和卷积核矩阵B分别搬运到向量存储单元和标量存储单元；

[0029] S2:将卷积矩阵A的一行元素和卷积核矩阵B的一个元素广播后对应的一行元素一一对应相乘，相乘的结果通过一个初始化为0的累加寄存器进行累加；

[0030] S3:通过混洗指令取出步骤S2中从卷积矩阵A中取出的一行元素的前K-1个元素至向量处理单元，并与本次取出并广播至向量处理单元的卷积核矩阵B的第二个元素进行一一对应相乘，结果累加至步骤S2中的累加寄存器中;其中K为并行处理单元的个数；

[0031] S4:判断是否完成卷积核矩阵B一行元素的计算，若没有完成，返回步骤S3，若完成则继续步骤S5；

[0032] S5:将卷积矩阵A和卷积核矩阵B的数据地址指向相应的下一数据行，最终完成卷积结果矩阵C的第一行元素的计算，通过循环步骤S2～步骤S5最终完成整个卷积结果矩阵C的计算。

[0033] 在上述步骤S1的搬运过程中，通过DMA控制器的配置，卷积矩阵A的每一行组织成一个数据帧，卷积核矩阵B的每一个元素组织成一个数据帧，整个卷积矩阵A共可分成p个数据帧，当所述数据帧的元素个数不等于向量处理器中并行处理单元的个数K的倍数时，在数据帧尾部补0使得每个数据帧的元素个数等于并行处理单元的个数K的倍数。

[0034] 在具体应用实例1中，本发明面向向量处理器的二维矩阵卷积的向量化实现方法，其详细流程为：

[0035] S101、输入卷积矩阵A和卷积核矩阵B;通过DMA控制器将卷积矩阵A和卷积核矩阵B分别搬运到向量存储单元和标量存储单元；

[0036] 即:通过DMA控制器的配置，可以将卷积矩阵A的每一行组织成一个数据帧，卷积核矩阵B的每一个元素组织成一个数据帧，整个卷积矩阵A共可分成p个数据帧。当数据帧的元素个数不等于向量处理器中并行处理单元的个数K的倍数时，在数据帧尾部补0使得每个数据帧的元素个数等于并行处理单元的个数K的倍数。

[0037] S102、将卷积矩阵A的一行元素和卷积核矩阵B的第1个元素广播后对应的一行元素一一对应相乘，相乘的结果通过一个初始化为0的累加寄存器进行累加。

[0038] S103、通过混洗指令取出步骤S102中从卷积矩阵A中取出的那一行元素的前K-1个元素至向量处理单元，并与本次取出并广播至向量寄存器的卷积核矩阵B的第2个元素进行一一对应相乘，结果累加至步骤S102中的累加寄存器中。

[0039] S104、判断是否完成卷积核矩阵B一行元素的计算，若没有完成，返回步骤S103，若完成则继续步骤S105。

[0040] S105、将卷积矩阵A和卷积核矩阵B的数据地址指向相应的下一数据行，最终完成卷积结果矩阵C的第一行元素的计算，通过循环步骤S102～步骤S105，最终完成整个卷积结果矩阵C的计算。

[0041] 对于m×m的卷积矩阵A卷积n×n的卷积核矩阵B的运算，可以得到(m-n+1)×(m-n+1)的卷积结果矩阵C，且m≧n，结果矩阵C的每一个元素都是由卷积核矩阵B(n×n)的所有元素在卷积矩阵A(m×m)上与卷积核矩阵B同样大小的区域进行点积运算并累加求和的结果。

[0042] 在具体应用实例2中，本发明面向向量处理器的二维矩阵卷积的向量化实现方法，用来计算规模为16×16的卷积矩阵A卷积规模为5×5的卷积核矩阵B(向量处理单元个数K为16)，如图8所示，其详细流程为：

[0043] S1001、输入卷积矩阵A(16×16)和卷积核矩阵B(5×5)；通过DMA搬运卷积矩阵A和卷积核矩阵B分别到向量存储单元和标量存储单元，卷积矩阵A在向量单元的存放方式如图5所示，卷积核矩阵B则连续存放在标量存储单元中。

[0044] S1002、将卷积矩阵A的一行元素和卷积核矩阵B的一个元算分别加载到向量处理单元和标量处理单元中，由于卷积矩阵A的规模为16×16，该向量处理器有16个同构处理单元，因此一次加载卷积矩阵A的一行元素至向量寄存器，加载卷积核矩阵B的第1个元素至标量寄存器，之后通过广播指令将该标量寄存器广播至向量寄存器中，将加载卷积矩阵A一行元素的向量寄存器与卷积核矩阵B广播后对应的向量寄存器对应相乘并累加至初始化为0的累加寄存器中；之后，通过特殊配置的混洗模式1将第一次取出的卷积矩阵A第一行16个元素的前15个元素取出至一个向量寄存器中，并将不足16个元素的位置写0(混洗模式1的实现方式如图6所示)；取卷积核矩阵B的第2个元素至标量寄存器，同样进行广播操作至向量寄存器中，并与此时的具有卷积矩阵A前15个元素的向量寄存器一一对应向乘，相乘结果累加至同样的累加寄存器中，由于卷积核矩阵B的规模为5×5，因此该循环共计进行5次(卷积核矩阵的列数)，进而完成卷积核矩阵B第一行元素的计算。

[0045] S1003、将卷积矩阵A和卷积核矩阵B的数据地址指向相应的下一数据行，返回步骤S1002完成卷积核矩阵B第2行元素的计算，该行计算结果同样累加至相同的累加寄存器中，由于卷积核矩阵B的规模为5×5，因此步骤S1003共计循环5次(卷积核矩阵的行数)完成卷积结果矩阵C(12×12)第一行12个元素的计算。

[0046] S1004、重复步骤S1002、S1003 12次(16-5+1)最终完成整个卷积结果C矩阵所有元素的计算。

[0047] 在具体应用实例3中，本发明面向向量处理器的二维矩阵卷积的向量化实现方法，用来计算规模为8×8的卷积矩阵A卷积规模为4×4的卷积核矩阵B(向量处理单元个数K为16)，如图9和图10所示，其详细流程为：

[0048] S10001、如图6所示，通过DMA搬运卷积矩阵A和卷积核矩阵B分别到向量存储单元和标量存储单元，这个过程与实施例1操作相同。

[0049] S10002、将卷积矩阵A的两行元素和卷积核矩阵B的第1个元素分别加载到向量处理单元和标量处理单元中，这里由于卷积矩阵A的规模为8×8，而向量处理单元的个数K为16，因此，一次可以加载卷积矩阵A的两行数据，由于是由卷积核的规模来控制核心循环的次数，因此即使加载的向量数据不是处理单元个数K的整数倍，也不需要额外进行补0操作。由于本次加载的是卷积矩阵A的前两行数据，首先取卷积核矩阵B的第1个元素至标量寄存器并通过标向量广播指令广播至向量寄存器中，通过乘加指令完成内层循环的第一次累加，继而通过混洗模式1对第1次取到的卷积矩阵A的向量寄存器进行混洗操作(混洗方法与实施例2相同)，接着取卷积核矩阵B的第2个元素并广播至向量寄存器，一一对应相乘后累

加至第一次的累加寄存器中，循环4次完成卷积核矩阵B一行元素的计算。

[0050] S10003、为了提高卷积矩阵A的数据复用率，通过混洗模式2将步骤2中取到的卷积矩阵A的前两行数据的第2行数据放置到另一个向量寄存器中，并将不足16个元素的处理单元置0(混洗模式2的实现方式如图7所示)，继而进入步骤S10002，循环4次完成卷积核矩阵B第二行元素的计算。

[0051] S10004、顺移到卷积矩阵A的下两行和卷积核矩阵的后两行，重复步骤S10002、S10003，完成卷积结果矩阵C(5×5)第一行元素的计算，重复步骤S10002～S1004共计5次循环，最终完成整个卷积结果矩阵C(5×5)的计算。

[0052] 以上仅是本发明的优选实施方式，本发明的保护范围并不仅局限于上述实施例，凡属于本发明思路下的技术方案均属于本发明的保护范围。应当指出，对于本技术领域的普通技术人员来说，在不脱离本发明原理前提下的若干改进和润饰，应视为本发明的保护范围。

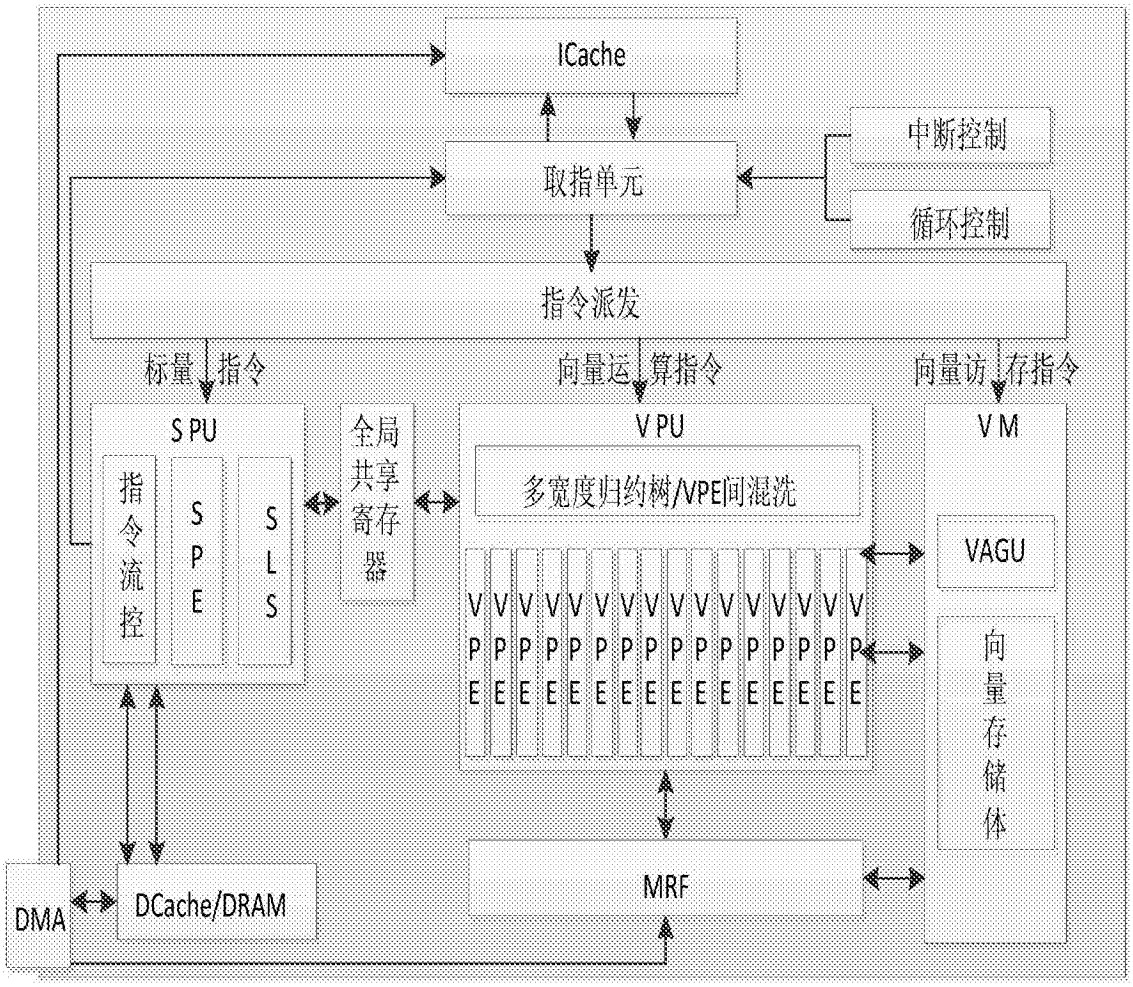

ICache
中断控制
取指单元
循环控制
指令派发
标量指令
向量运算指令
向量访存指令
S PU
指令流控
S P E
S L S
全局共享寄存器
V PU
多宽度归约树/VPE间混洗
V P E
V M
VAGU
向量存储体
MRF
DMA
DCache/DRAM

图1

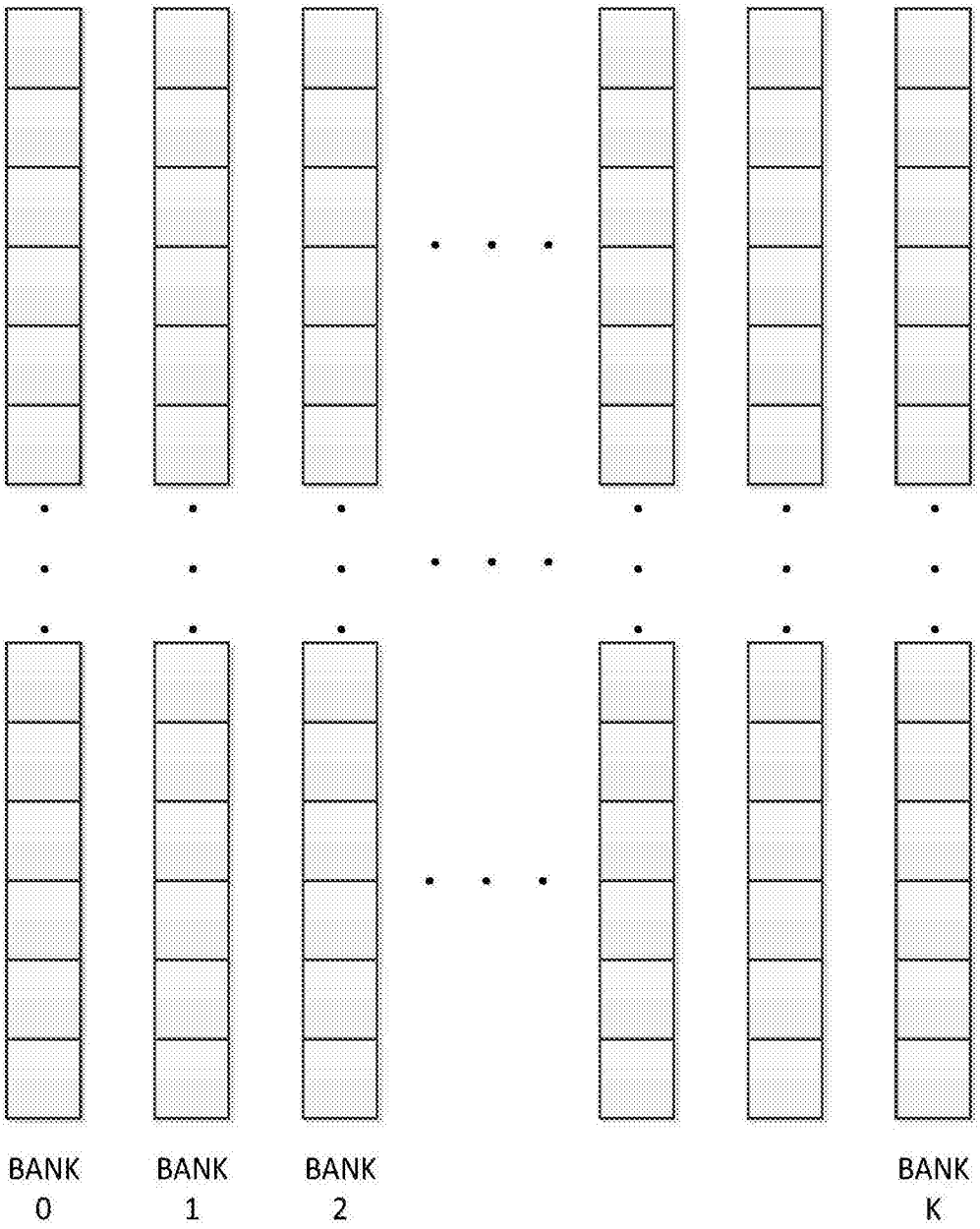



图2

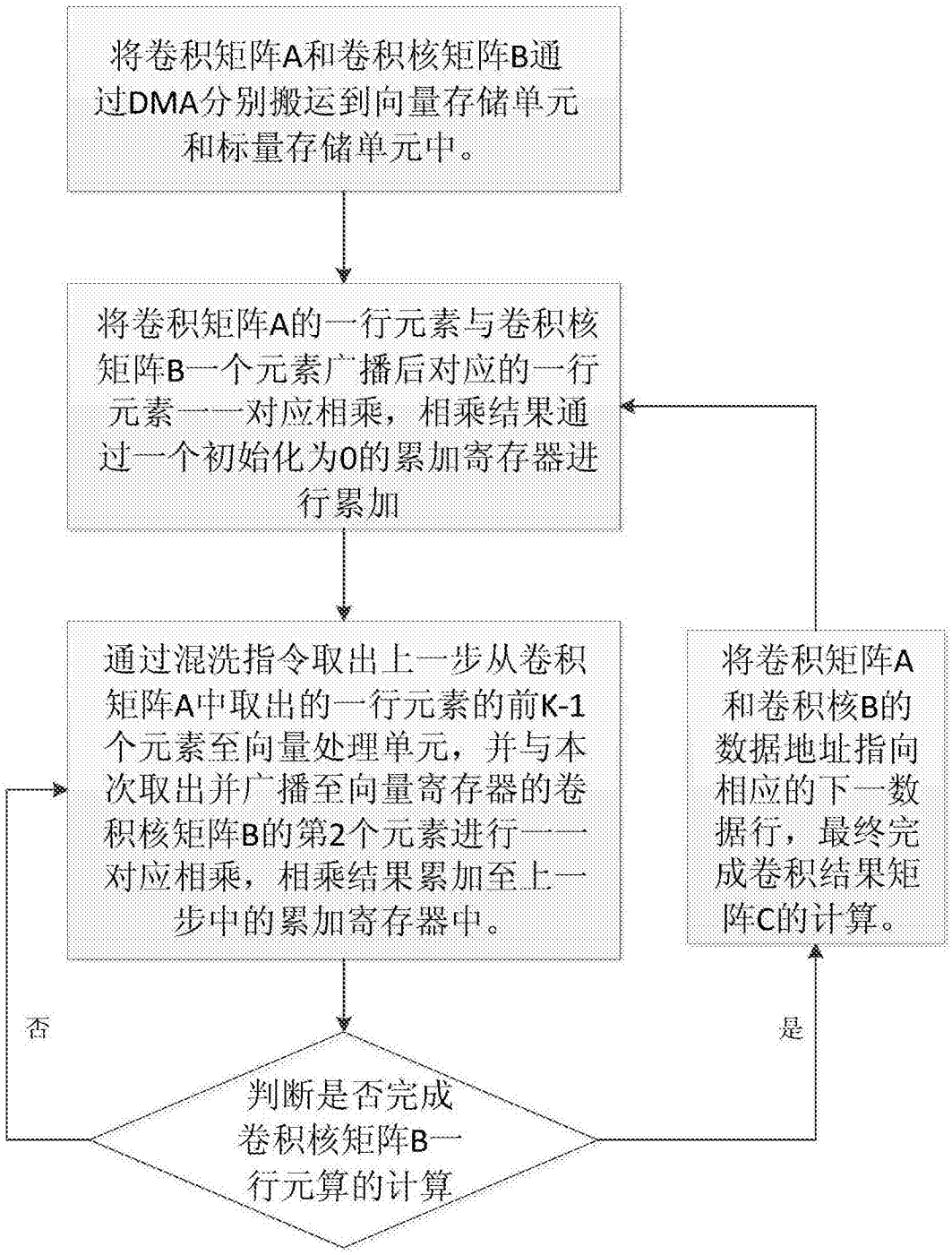

将卷积矩阵A和卷积核矩阵B通过DMA分别搬运到向量存储单元和标量存储单元中。
将卷积矩阵A的一行元素与卷积核矩阵B一个元素广播后对应的一行元素一一对应相乘，相乘结果通过一个初始化为0的累加寄存器进行累加
通过混洗指令取出上一步从卷积矩阵A中取出的一行元素的前K-1个元素至向量处理单元，并与本次取出并广播至向量寄存器的卷积核矩阵B的第2个元素进行一一对应相乘，相乘结果累加至上一步中的累加寄存器中。
将卷积矩阵A和卷积核B的数据地址指向相应的下一数据行，最终完成卷积结果矩阵C的计算。
否
是
判断是否完成卷积核矩阵B一行元算的计算

图3

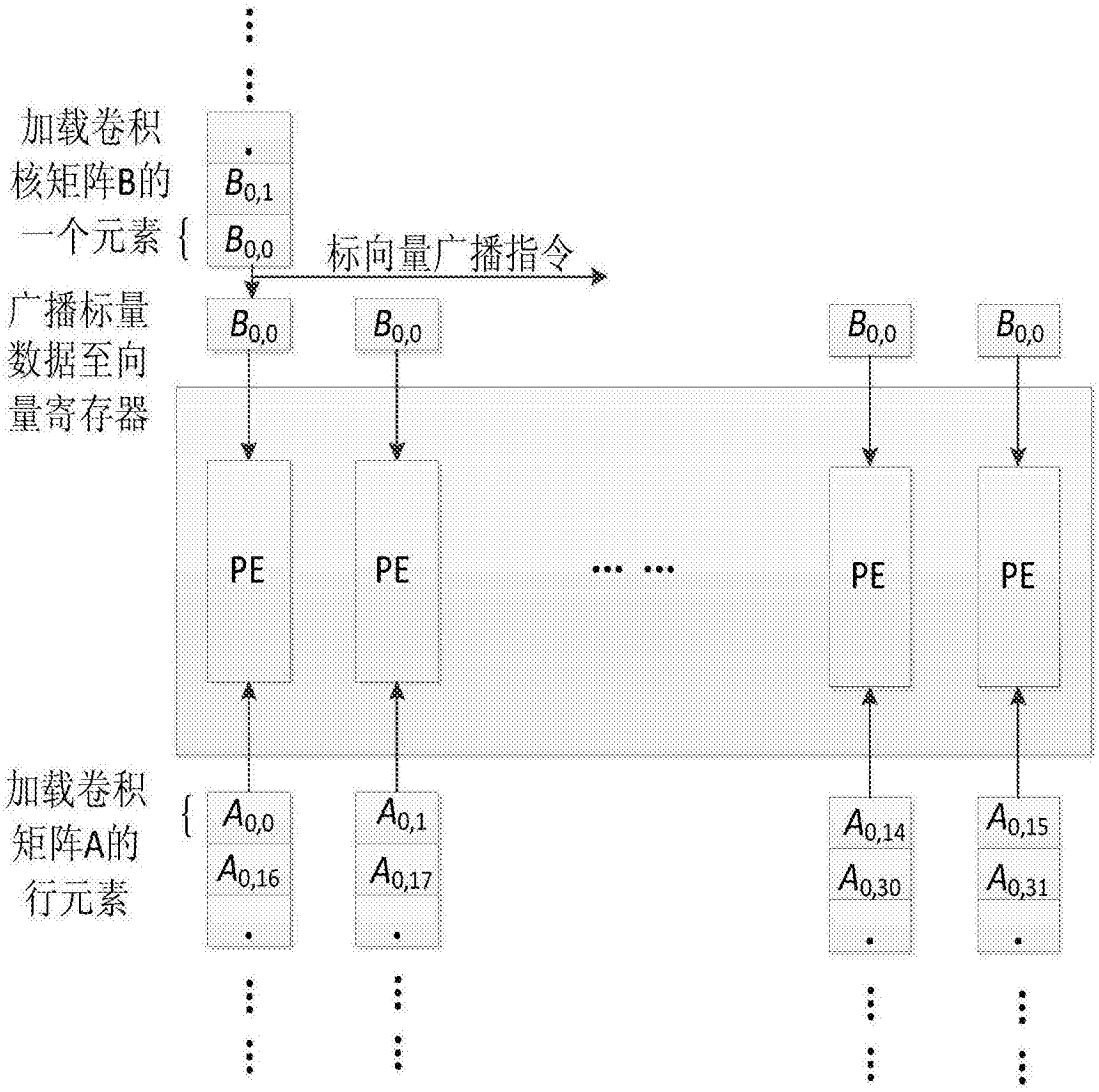

加载卷积
核矩阵B的
一个元素
$B_{0,1}$
$B_{0,0}$
标向量广播指令
广播标量
数据至向
量寄存器
$B_{0,0}$
$B_{0,0}$
$B_{0,0}$
$B_{0,0}$
PE
PE
PE
PE
加载卷积
矩阵A的
行元素
$A_{0,0}$
$A_{0,16}$
$A_{0,1}$
$A_{0,17}$
$A_{0,14}$
$A_{0,30}$
$A_{0,15}$
$A_{0,31}$

图4

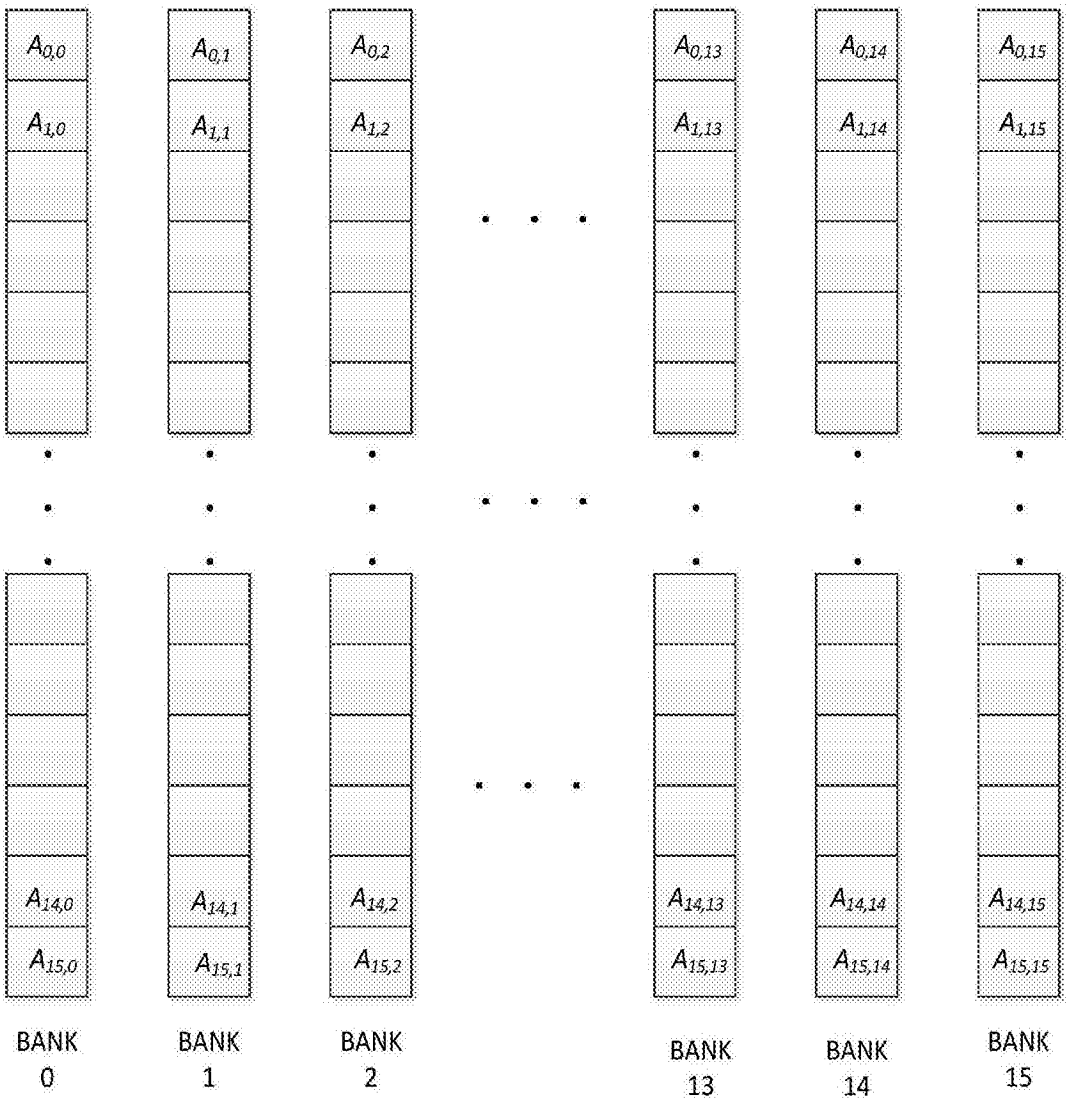

$A_{0,0}$
$A_{1,0}$
$A_{14,0}$
$A_{15,0}$
BANK 0
$A_{0,1}$
$A_{1,1}$
$A_{14,1}$
$A_{15,1}$
BANK 1
$A_{0,2}$
$A_{1,2}$
$A_{14,2}$
$A_{15,2}$
BANK 2
$A_{0,13}$
$A_{1,13}$
$A_{14,13}$
$A_{15,13}$
BANK 13
$A_{0,14}$
$A_{1,14}$
$A_{14,14}$
$A_{15,14}$
BANK 14
$A_{0,15}$
$A_{1,15}$
$A_{14,15}$
$A_{15,15}$
BANK 15

图5

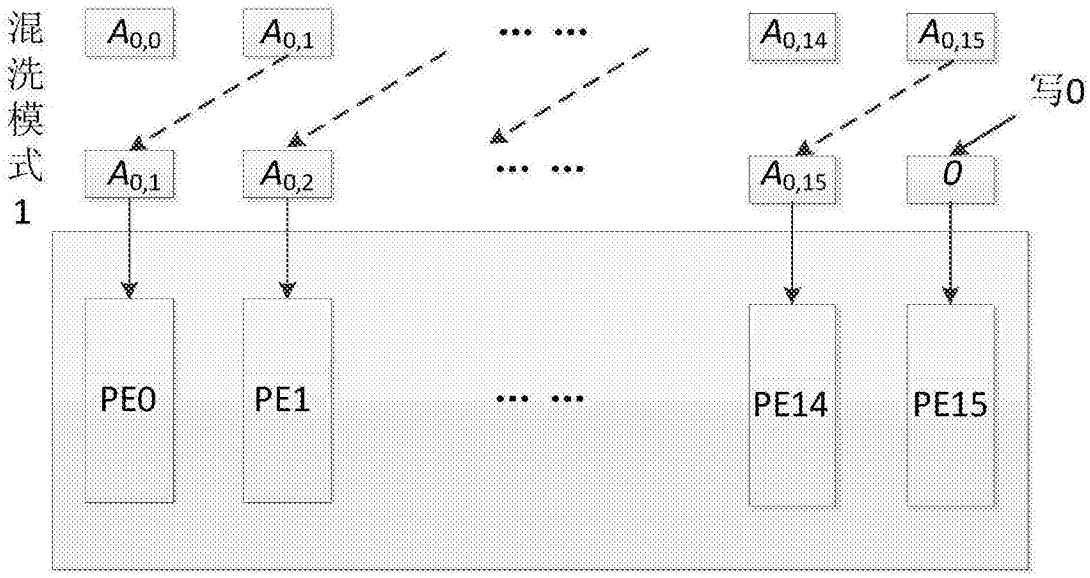

混洗模式1
$A_{0,0}$
$A_{0,1}$
$A_{0,14}$
$A_{0,15}$
写0
$A_{0,1}$
$A_{0,2}$
$A_{0,15}$
0
PE0
PE1
PE14
PE15

图6

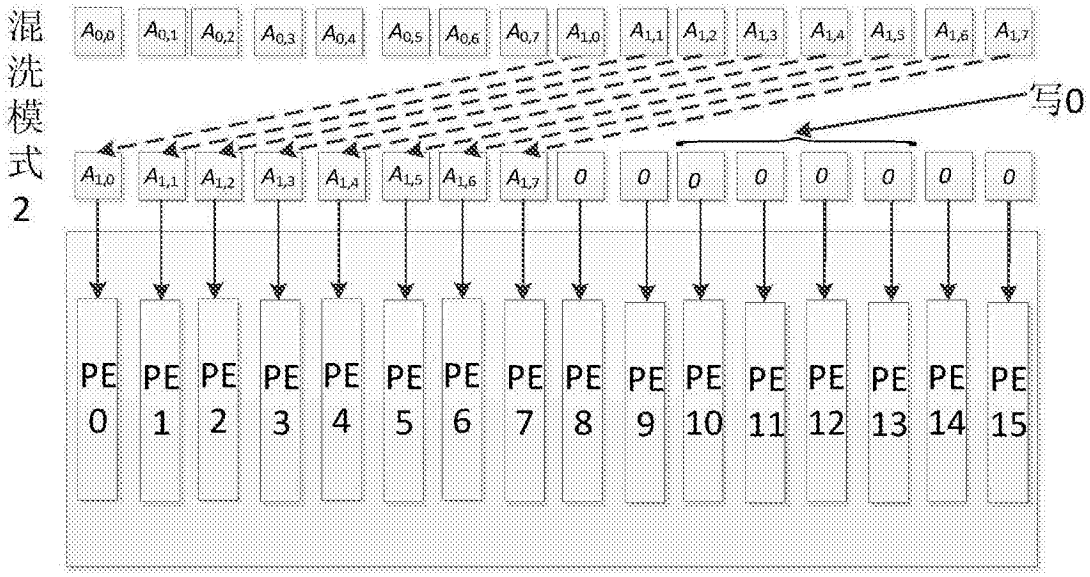

混洗模式2
$A_{0,0}$ $A_{0,1}$ $A_{0,2}$ $A_{0,3}$ $A_{0,4}$ $A_{0,5}$ $A_{0,6}$ $A_{0,7}$ $A_{1,0}$ $A_{1,1}$ $A_{1,2}$ $A_{1,3}$ $A_{1,4}$ $A_{1,5}$ $A_{1,6}$ $A_{1,7}$
写0
$A_{1,0}$ $A_{1,1}$ $A_{1,2}$ $A_{1,3}$ $A_{1,4}$ $A_{1,5}$ $A_{1,6}$ $A_{1,7}$ 0 0 0 0 0 0 0 0
PE 0 PE 1 PE 2 PE 3 PE 4 PE 5 PE 6 PE 7 PE 8 PE 9 PE 10 PE 11 PE 12 PE 13 PE 14 PE 15

图7

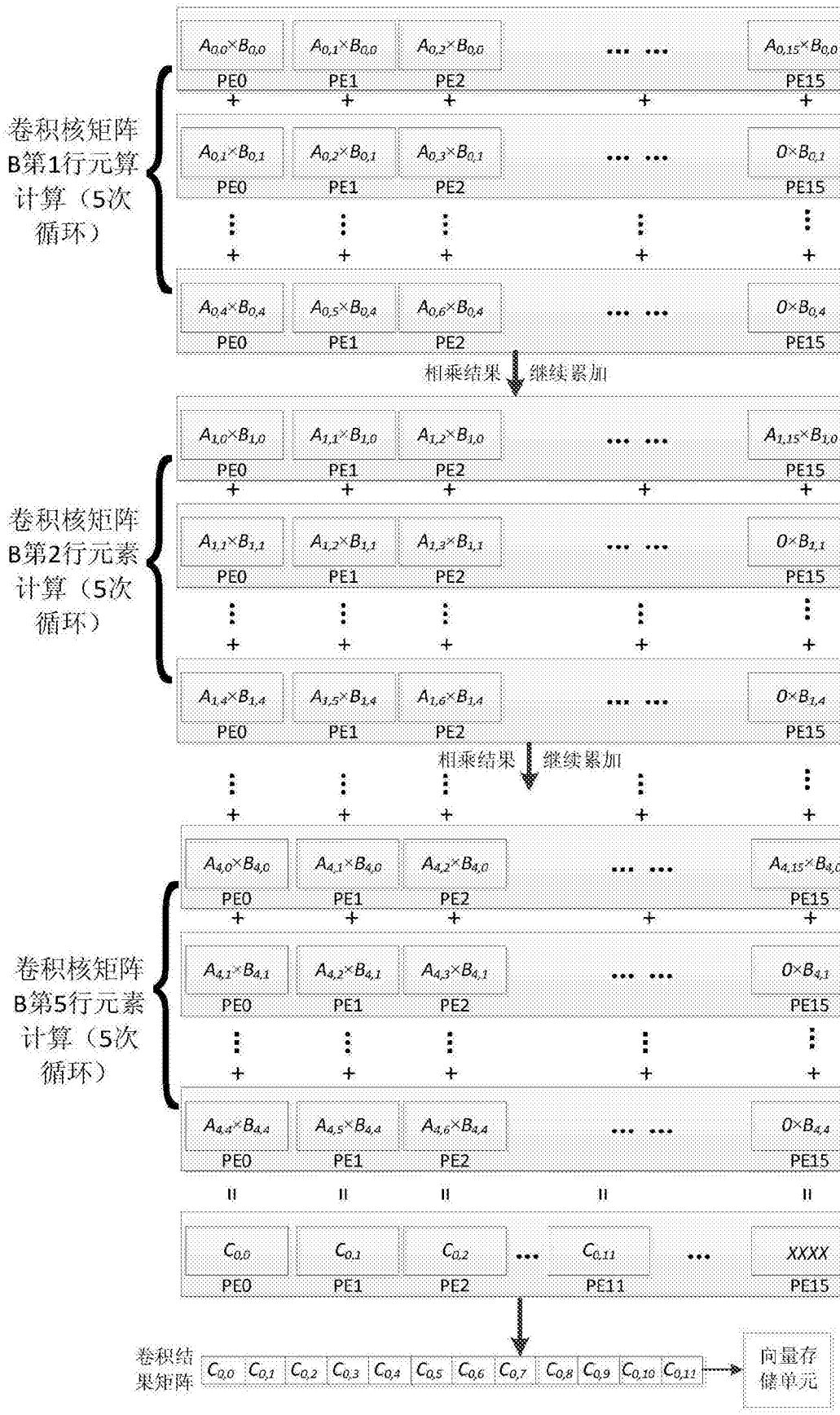

卷积核矩阵B第1行元算计算（5次循环）
卷积核矩阵B第2行元素计算（5次循环）
卷积核矩阵B第5行元素计算（5次循环）
相乘结果
继续累加
PE0
PE1
PE2
PE11
PE15
XXXX
卷积结果矩阵
向量存储单元

图8

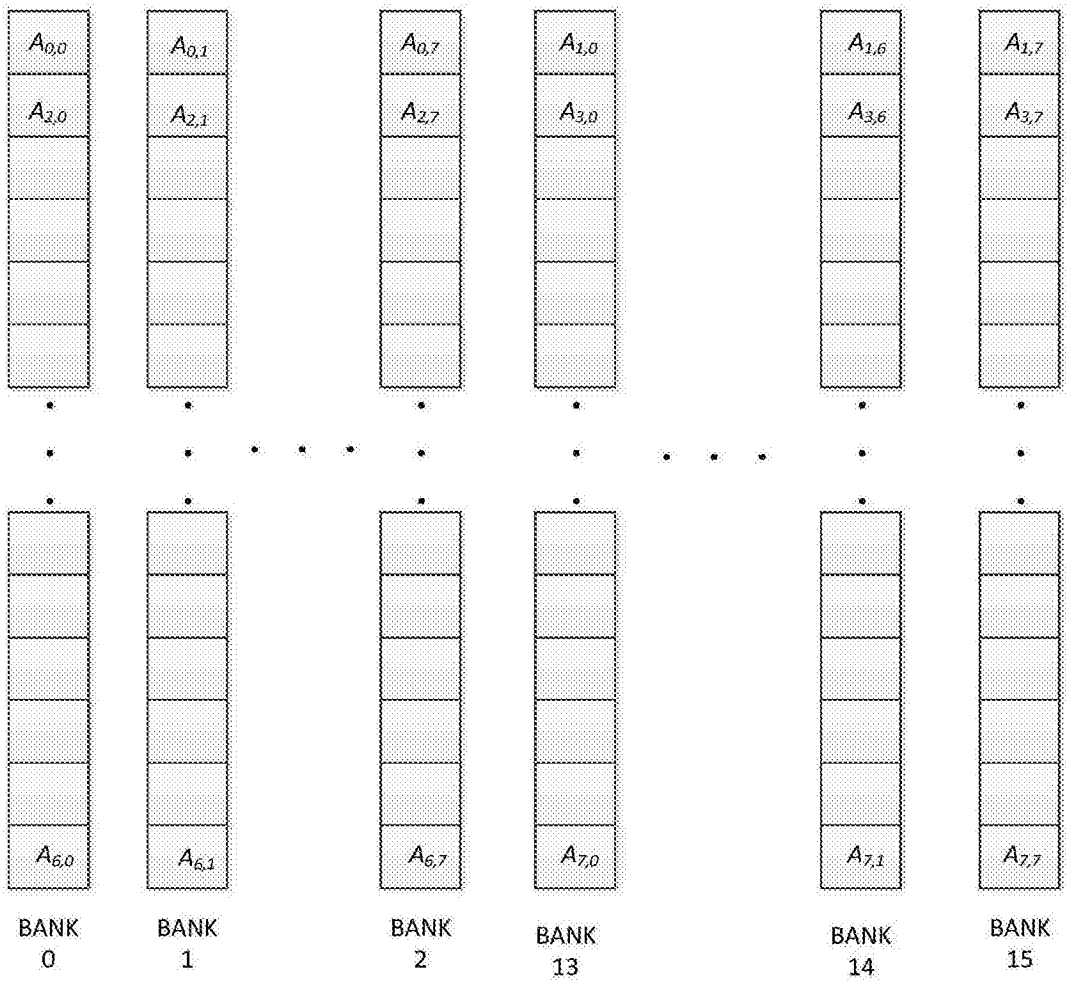

$A_{0,0}$
$A_{2,0}$
$A_{0,1}$
$A_{2,1}$
$A_{0,7}$
$A_{2,7}$
$A_{1,0}$
$A_{3,0}$
$A_{1,6}$
$A_{3,6}$
$A_{1,7}$
$A_{3,7}$
$A_{6,0}$
$A_{6,1}$
$A_{6,7}$
$A_{7,0}$
$A_{7,1}$
$A_{7,7}$
BANK 0
BANK 1
BANK 2
BANK 13
BANK 14
BANK 15

图9

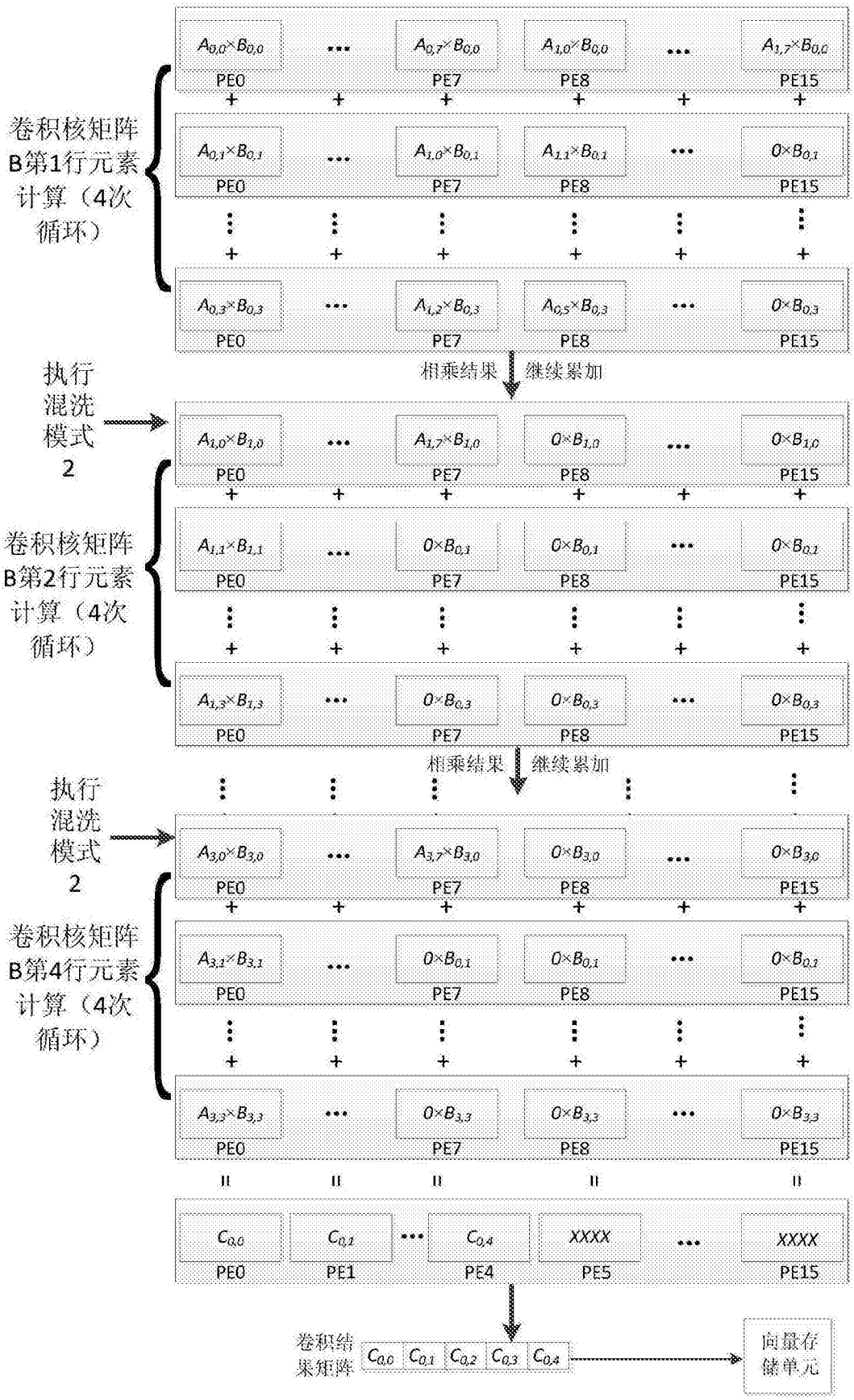

卷积核矩阵B第1行元素计算（4次循环）
执行混洗模式2
卷积核矩阵B第2行元素计算（4次循环）
相乘结果
继续累加
执行混洗模式2
卷积核矩阵B第4行元素计算（4次循环）
PE0
PE1
PE4
PE5
PE7
PE8
PE15
XXXX
卷积结果矩阵
向量存储单元

图10