



# (12)发明专利

(10)授权公告号 CN 110457069 B

(45)授权公告日 2020.08.18

(21)申请号 201910630145.2

(22)申请日 2017.01.13

(65)同一申请的已公布的文献号  
申请公布号 CN 110457069 A

(43)申请公布日 2019.11.15

(30)优先权数据  
15/014,265 2016.02.03 US

(62)分案原申请数据  
201710025689.7 2017.01.13

(73)专利权人 谷歌有限责任公司  
地址 美国加利福尼亚州

(72)发明人 禹同熾  
安德鲁·埃弗里特·菲尔普斯

(74)专利代理机构 中原信达知识产权代理有限  
责任公司 11219

代理人 周亚荣 邓聪惠

(51)Int.Cl.  
G06F 9/30(2006.01)  
G06F 9/32(2006.01)

(56)对比文件  
US 2004093550 A1,2004.05.13  
WO 0190888 A1,2001.11.29  
WO 2013095601 A1,2013.06.27  
CN 104838357 A,2015.08.12  
CN 103440513 A,2013.12.11

审查员 吴卿

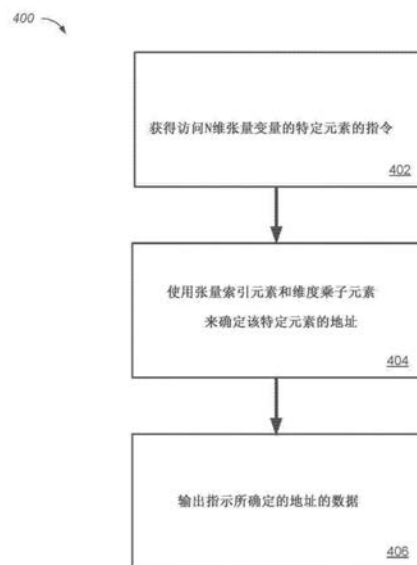
权利要求书4页 说明书12页 附图11页

## (54)发明名称

访问多维张量中的数据

## (57)摘要

本申请涉及访问多维张量中的数据。用于处理用来访问N维张量的指令的方法、系统、和装置,所述装置包括多个张量索引元素和多个维度乘子元素,其中,所述维度乘子元素中的每一个维度乘子元素具有对应张量索引元素。所述装置包括一个或多个处理器,所述一个或多个处理器被配置成获得用于访问N维张量的特定元素的指令,其中,所述N维张量具有跨N个维度中的每一个维度布置的多个元素,并且其中,N是等于或大于1的整数;使用所述多个张量索引元素中的一个或多个张量索引元素和所述多个维度乘子元素中的一个或多个维度乘子元素来确定所述特定元素的地址;以及输出指示用于访问所述N维张量的所述特定元素的所确定的地址的数据。



1. 一种用于处理多维张量输入的装置,所述装置包括:

多个张量索引元素,所述多个张量索引元素包括用于遍历N维张量的多个嵌套循环中的每个嵌套循环的张量索引元素,其中,每个张量索引元素是相应的第一硬件寄存器,所述相应的第一硬件寄存器被配置来存储用于遍历所述N维张量的相应维度的相应嵌套循环的索引值,并且其中,所述N维张量包括跨N个维度中的每一个维度布置的数据元素并且N是等于或大于一的整数,其中,每个索引值是每次该索引值的相应嵌套循环的迭代被执行时更新的值;

多个维度乘子元素,其中,每个维度乘子元素是被配置来存储所述N维张量的相应维度的乘子值的相应的第二硬件寄存器,其中,每个乘子值是所述N维张量的、该乘子值的相应维度的常数值并且在用于遍历该相应维度的相应嵌套循环的每次迭代中保持恒定;以及

一个或多个硬件处理器,所述一个或多个硬件处理器被配置来执行由所述一个或多个硬件处理器可执行的指令集中的一个或多个指令,其中,所述一个或多个指令的执行使得所述一个或多个硬件处理器执行包括以下的操作:

通过以下操作来确定存储器中的、用于存储所述N维张量的数据元素序列的数据值的位置的存储器地址:

对于所述嵌套循环的内循环的每次迭代:

对于所述N维张量中的每一个维度,确定以下的乘积:(i) 用于遍历该维度的嵌套循环的张量索引元素中存储的索引值以及(ii) 该维度的乘子值;以及

基于所述N维张量中的每一个维度的乘积的总和来确定与所述内循环的该次迭代相对应的数据元素的存储器地址;以及

输出指示所述N维张量的所述数据元素序列中的每个数据元素的所确定的存储器地址的数据。

2. 根据权利要求1所述的装置,其中,基于所述N维张量中的每一个维度的乘积的总和来确定与所述内循环的该次迭代相对应的数据元素的存储器地址包括确定以下的总和:(i) 所述N维张量中的每一个维度的乘积的总和以及(ii) 基础存储器地址。

3. 根据权利要求1所述的装置,其中,输出指示每个数据元素的所确定的存储器地址的数据包括:以所确定的存储器地址被确定的顺序来依次输出指示所确定的存储器地址的数据。

4. 根据权利要求1所述的装置,其中,所述一个或多个硬件处理器进一步被配置来进行以下操作:

每次所述内循环完成时,将所述内循环的索引值递增第一递增值;以及

每次嵌套所述内循环的第二循环完成时,将所述第二循环的索引值递增第二递增值。

5. 根据权利要求1所述的装置,其中,所述一个或多个硬件处理器进一步被配置来进行以下操作:

接收用于更新所述内循环的索引值的指令;

在接收到用于更新所述内循环的索引值的所述指令之后,确定所述内循环的索引值和所述内循环的张量边界元素中所存储的张量边界值之间的差满足阈值并且作为响应:

将所述内循环的索引值递增第一递增值。

6. 根据权利要求1所述的装置,其中,所述一个或多个硬件处理器进一步被配置来进行

以下操作：

接收用于更新所述内循环的索引值的指令；

在接收到用于更新所述内循环的索引值的所述指令之后，确定所述内循环的索引值和所述内循环的张量边界元素中所存储的张量边界值之间的差不满足阈值并且作为响应：

将所述内循环的索引值重置为所述内循环的初始值；以及

将嵌套所述内循环的第二循环的索引值递增第二递增值。

7. 根据权利要求1所述的装置，其中，所述一个或多个硬件处理器包括一个或多个算术逻辑单元。

8. 一种用于处理多维张量输入的系统，包括：

一个或多个硬件处理器，所述一个或多个硬件处理器被配置来对N维张量执行线性运算，其中，所述N维张量具有跨N个维度中的每一个维度布置的数据元素，并且其中，N是等于或大于一的整数；

多个张量索引元素，所述多个张量索引元素包括用于遍历所述N维张量的多个嵌套循环中的每个嵌套循环的张量索引元素，其中，每个张量索引元素是相应的第一硬件寄存器，所述相应的第一硬件寄存器被配置来存储用于遍历所述N维张量的相应维度的相应嵌套循环的索引值，并且其中，每个索引值是每次该索引值的相应嵌套循环的迭代被执行时更新的值；

多个维度乘子元素，其中，每个维度乘子元素是被配置来存储所述N维张量的相应维度的乘子值的相应的第二硬件寄存器，其中，每个乘子值是所述N维张量的、该乘子值的相应维度的常数值并且在用于遍历该相应维度的相应嵌套循环的每次迭代中保持恒定；以及

硬件电路，所述硬件电路被配置来进行以下操作：

通过以下操作来确定所述N维张量的数据元素序列的存储器地址：

对于所述嵌套循环的内循环的每次迭代：

对于所述N维张量中的每一个维度，确定以下的乘积：(i) 用于遍历该维度的嵌套循环的张量索引元素中存储的索引值以及(ii) 该维度的乘子值；以及

基于所述N维张量中的每一个维度的乘积的总和来确定与所述内循环的该次迭代相对应的数据元素的存储器地址；以及

输出指示所述N维张量的所述数据元素序列中的每个数据元素的所确定的存储器地址的数据。

9. 根据权利要求8所述的系统，其中，基于所述N维张量中的每一个维度的乘积的总和来确定与所述内循环的该次迭代相对应的数据元素的存储器地址包括确定以下的总和：(i) 所述N维张量中的每一个维度的乘积的总和以及(ii) 基础存储器地址。

10. 根据权利要求8所述的系统，其中，输出指示每个数据元素的所确定的存储器地址的数据包括：以所确定的存储器地址被确定的顺序来依次输出指示所确定的存储器地址的数据。

11. 根据权利要求8所述的系统，其中，所述硬件电路进一步被配置来进行以下操作：

每次所述内循环完成时，将所述内循环的索引值递增第一递增值；以及

每次嵌套所述内循环的第二循环完成时，将所述第二循环的索引值递增第二递增值。

12. 根据权利要求8所述的系统，其中，所述硬件电路进一步被配置来进行以下操作：

接收用于更新所述内循环的索引值的指令；

在接收到用于更新所述内循环的索引值的所述指令之后，确定所述内循环的索引值和所述内循环的张量边界元素中所存储的张量边界值之间的差满足阈值并且作为响应：

将所述内循环的索引值递增第一递增值。

13. 根据权利要求8所述的系统，其中，所述硬件电路进一步被配置来进行以下操作：

接收用于更新所述内循环的索引值的指令；

在接收到用于更新所述内循环的索引值的所述指令之后，确定所述内循环的索引值和所述内循环的张量边界元素中所存储的张量边界值之间的差不满足阈值并且作为响应：

将所述内循环的索引值重置为所述内循环的初始值；以及

将嵌套所述内循环的第二循环的索引值递增第二递增值。

14. 根据权利要求8所述的系统，其中，所述一个或多个硬件处理器包括一个或多个算术逻辑单元。

15. 一种由数据处理装置执行的方法，所述数据处理装置包括一个或多个硬件处理器，所述一个或多个硬件处理器执行所述一个或多个硬件处理器可执行的指令集中的一个或多个指令，所述方法包括：

由所述一个或多个硬件处理器确定N维张量的数据元素序列的存储器地址，其中，所述N维张量包括跨N个维度中的每一个维度布置的数据元素，并且N是等于或大于一的整数，所述确定包括：

对于用于遍历所述N维张量的多个嵌套循环中的内循环的每次迭代：

由所述一个或多个硬件处理器对于所述N维张量中的每一个维度，确定以下的乘积：

(i) 用于遍历该维度的嵌套循环的张量索引元素中存储的索引值以及(ii) 该维度的维度乘子元素中所存储的该维度的乘子值，其中：

每个嵌套循环具有存储在对应的张量索引元素中的对应的索引值，其中，每个张量索引元素是相应的第一硬件寄存器并且每个索引值是每次与该索引值相对应的嵌套循环的迭代被执行时更新的值；以及

每个嵌套循环具有存储在对应的维度乘子元素中的对应的乘子值，其中，每个维度乘子元素是相应的第二硬件寄存器并且每个乘子值是该乘子值的维度的常数值，每个乘子值对于与该乘子值相对应的嵌套循环的每次迭代保持恒定；以及

由所述一个或多个硬件处理器，基于所述N维张量中的每一个维度的乘积的总和来确定与所述内循环的该次迭代相对应的数据元素的存储器地址；以及

由所述一个或多个硬件处理器输出指示所述N维张量的所述数据元素序列中的每个数据元素的所确定的存储器地址的数据。

16. 根据权利要求15所述的方法，其中，基于所述N维张量中的每一个维度的乘积的总和来确定与所述内循环的该次迭代相对应的数据元素的存储器地址包括确定以下的总和：

(i) 所述N维张量中的每一个维度的乘积的总和以及(ii) 基础存储器地址。

17. 根据权利要求15所述的方法，其中，输出指示每个数据元素的所确定的存储器地址的数据包括：以所确定的存储器地址被确定的顺序来依次输出指示所确定的存储器地址的数据。

18. 根据权利要求15所述的方法，进一步包括：

每次所述内循环完成时,将所述内循环的索引值递增第一递增值;以及  
每次嵌套所述内循环的第二循环完成时,将所述第二循环的索引值递增第二递增值。

19. 根据权利要求15所述的方法,进一步包括:

接收用于更新所述内循环的索引值的指令;

在接收到用于更新所述内循环的索引值的所述指令之后,确定所述内循环的索引值和  
所述内循环的张量边界元素中所存储的张量边界值之间的差满足阈值并且作为响应:

将所述内循环的索引值递增第一递增值。

20. 根据权利要求15所述的方法,进一步包括:

接收用于更新所述内循环的索引值的指令;

在接收到用于更新所述内循环的索引值的所述指令之后,确定所述内循环的索引值和  
所述内循环的张量边界元素中所存储的张量边界值之间的差不满足阈值并且作为响应:

将所述内循环的索引值重置为所述内循环的初始值;以及

将嵌套所述内循环的第二循环的索引值递增第二递增值。

## 访问多维张量中的数据

[0001] 分案说明

[0002] 本申请属于申请日为2017年1月13日的中国发明专利申请No.201710025689.7的分案申请。

### 背景技术

[0003] 本说明书大体涉及使用寄存器以用于计算存储器地址值。寄存器可以被用来处理分支条件。

### 发明内容

[0004] 根据本说明书中描述的主题的一个创新性方面,寄存器可以被用于迭代嵌套循环结构,以计算对应的存储器地址,并且用于允许程序员处理循环边界条件。通常,寄存器可以被布置来维护体系结构状态的若干集合,包括循环边界值、每个循环中的当前循环索引变量、用于计算存储器地址值的维度乘子,以及用于处理分支循环边界的程序计数器值。该布置对于处理一个或多个多维数据结构是有利的。利用这些体系结构状态集合,处理器可以执行将索引变量移动到新的位置、检索存储器地址值、并且跳转至新的分支目标的指令。

[0005] 通常,在本说明书中所描述的主题的一个创新性方面可以体现在用于处理用来访问N维张量的指令的装置中。所述装置包括:多个张量索引元素和多个维度乘子元素,其中,所述维度乘子元素中的每一个维度乘子元素具有对应张量索引元素。所述装置包括一个或多个处理器,所述一个或多个处理器被配置成获得用于访问N维张量的特定元素的指令,其中,所述N维张量具有跨N个维度中的每一个维度布置的多个元素,并且其中,N是等于或大于1的整数;使用所述多个张量索引元素中的一个或多个张量索引元素和所述多个维度乘子元素中的一个或多个维度乘子元素来确定所述特定元素的地址;以及输出指示用于访问所述N维张量的所述特定元素的所确定的地址的数据。

[0006] 这些和其他实施方式每个能够可选地包括以下特征中的一个或多个。例如,所述特定元素的所述地址可以是与所述N维张量的另一个元素的地址偏移。为了确定所述特定元素的所述地址,对于所述一个或多个张量索引元素中的每一个张量索引元素,该相应张量索引元素中所存储的值可以被乘以一个或多个的多个张量索引元素的对应张量索引元素中所存储的值,以生成相应的相乘值。可以通过确定所述相乘值的总和来确定所述地址偏移。

[0007] 所述多个张量索引元素可以被布置到M组张量索引元素中,M是大于1的正整数。每组张量索引元素可以被指派给相区别且不同的多维张量。

[0008] 所述装置可以包括多个张量边界元素,其中,所述多个张量边界元素中的每一个张量边界元素可以具有在所述多个张量索引元素中的对应张量索引元素。所述指令可以表示用于处理包括第一循环和第二循环的嵌套循环的指令,所述第二循环是所述第一循环的外循环,其中,所述第一循环使用第一索引变量来迭代,并且其中,所述第二循环使用第二索引变量来迭代。

[0009] 所述一个或多个处理器可以被配置成将所述多个张量索引元素中所存储的值、所述多个维度乘子元素中所存储的值、以及所述多个张量边界元素中所存储的值设置为初始化值。所述一个或多个处理器可以进一步被配置成将所述多个张量边界元素中的第一张量边界元素中所存储的值设置为用于迭代所述第一循环的所述第一索引变量的上边界值；将所述多个张量边界元素中的第二张量边界元素中所存储的值设置为用于迭代所述第二循环的所述第二索引变量的上边界值；将所述多个维度乘子元素中的第一维度乘子元素中所存储的值设置为预定值；以及将所述多个维度乘子元素中的第二维度乘子元素中所存储的值设置为所述第一索引变量的所述上边界值。

[0010] 为了确定所述N维张量中的元素的地址，所述一个或多个处理器可以被配置成将所述多个张量索引元素中的第一张量索引元素中所存储的值设置为所述第一索引变量的当前值；将所述多个张量索引元素中的第二张量索引元素中所存储的值设置为所述第二索引变量的当前值；将所述第一维度乘子元素中所存储的值乘以所述第一张量索引元素中所存储的值，以获得第一相乘值；将所述第二维度乘子元素中所存储的值乘以所述第二张量索引元素中所存储的值，以获得第二相乘值；以及通过至少对所述第一相乘值和所述第二相乘值求和来确定该地址。

[0011] 所述一个或多个处理器被可以配置成接收用于更新所述第一张量索引元素中所存储的值的指令；在接收到所述用于更新所述第一张量索引元素中所存储的值的指令之后，确定所述第一张量索引元素中所存储的值和所述第一张量边界元素中所存储的值之间的差满足阈值；以及响应于确定所述第一张量索引元素中所存储的值和所述第一张量边界元素中所存储的值之间的差满足所述阈值，更新所述第一张量索引元素中所存储的值。每次所述第一循环完成时，所述第一索引变量被递增第一递增值。为了更新所述第一张量索引元素中所存储的值，所述第一张量索引元素中所存储的值可以被递增所述第一递增值。

[0012] 所述一个或多个处理器可以被配置成接收用于更新所述第一张量索引元素中所存储的值的指令；在接收到用于更新所述第一张量索引元素中所存储的值的指令之后，确定所述第一张量索引元素中所存储的值和所述第一张量边界元素中所存储的值之间的差不满足阈值。响应于确定所述第一张量索引元素中所存储的值和所述第一张量边界元素中所存储的值之间的差不满足所述阈值，所述一个或多个处理器可以被配置成重置所述第一张量索引元素中所存储的值；以及更新所述第二张量索引元素中所存储的值。每次所述第二循环完成时，所述第二索引变量被递增第二递增值。为了更新所述第二张量索引元素中所存储的值，所述第二张量索引元素中所存储的值可以被递增所述第二递增值。

[0013] 所述一个或多个处理器可以包括一个或多个算术逻辑单元。所述张量索引元素和所述维度乘子元素中的每一个是寄存器。所述装置可以包括多个分支目标元素，其中，所述多个分支目标元素中的每一个分支目标元素具有在所述多个张量索引元素中的对应张量索引元素，并且其中，所述多个分支目标元素中的每一个分支目标元素被配置成存储当与该相应分支目标元素对应的循环的迭代完成时执行的指令的地址。

[0014] 本说明书中所描述的主题能够被实现在特定实施例中以便实现以下优点中的一个或多个。使用寄存器以保持跟踪存储器地址值允许程序利用一条指令来迭代深度嵌套循环。存储器地址值能够基于寄存器中所存储的值通过应用简单算术计算来快速确定。使用算术逻辑单元来确定存储器地址值允许减少处理器处的计算周期的数目，并且为其他计算

任务增加处理器带宽。能够以减少的数目的指令来遍历张量。二维寄存器体系结构允许多维张量,每个多维张量具有待被同时跟踪的不同维度。通过具有分支目标表,能够减少分支指令相关的开销。

[0015] 该方面以及其他方面的其他实施方式包括对应的系统、装置、和被配置成执行所述方法的动作并且被编码在计算机存储设备上的计算机程序。一个或多个计算机的系统能够借助于在操作中安装在该系统上使得该系统执行动作的软件、固件、硬件或其组合来配置。一个或多个计算机程序能够借助于具有指令来如此配置,所述指令当被数据处理装置执行时使得该装置执行动作。

[0016] 本说明书中描述的主题的一个或多个实施方式的细节在以下附图和描述中得以陈述。本主题的其他潜在特征、方面、和优势将根据描述、附图、和权利要求而变得显而易见。

## 附图说明

[0017] 图1是示例计算系统的框图。

[0018] 图2A-图2H图示了示例张量遍历单元。

[0019] 图3图示了示例张量遍历单元。

[0020] 图4是图示用于确定多维张量变量的地址的过程的示例的流程图。

[0021] 各附图中相似附图标记和名称指示相似的元素。

## 具体实施方式

[0022] 大体上,当软件算法处理N维张量时,可以使用嵌套循环,其中每个循环负责遍历该N维张量的每一个维度。多维张量可以是矩阵或者多维矩阵。N维张量的每一个维度可以包括一个或多个元素,其中每个元素可以存储相应的数据值。例如,张量可以是程序中的变量,其中该变量可以具有三个维度。第一维度可以具有300个元素的长度,第二维度可以具有1000个元素的长度,并且第三维度可以具有20个元素的长度。以嵌套循环来遍历张量需要计算元素的存储器地址值以加载或存储元素的对应数据值。例如,for循环是嵌套循环,其中由三个循环索引变量跟踪的三个循环能够被嵌套以遍历该三维张量。在一些情况下,处理器可能需要执行循环边界条件 (bound condition),诸如利用外循环索引变量来设置内循环的循环边界。例如,在确定是否退出嵌套循环的最内层 (inner-most) 循环时,程序可以将该最内层循环的循环索引变量的当前值与该嵌套循环的最外层 (outer-most) 循环的循环索引变量的当前值进行比较。

[0023] 这些任务可能需要大量的指令,诸如分支指令和整数算术指令。当每个循环边界很小并且循环数目很大时,计算可能耗费总执行时间的相当一部分,并且严重降级整体性能。例如,处理器可以被设计来处理N维张量输入。处理器可以被实现在2维架构上,其将 (N-2) 个维度留给软件来处理。当N很大时,对N-2个维度的处置对于处理器而言在计算上可能很繁重,因为该计算需要非常高量的计算带宽。处理器的硬件张量遍历单元可以通过降低该处理器在遍历张量时所需处理的维度数目来提高处理器的计算带宽。

[0024] 图1示出了用于遍历张量的示例计算系统100的框图。大体上,计算系统100处理输入104以生成输出116。计算系统100可以被配置成执行线性代数计算。输入104可以是能够



由计算系统100处理的任何合适的的数据。计算系统100包括处理单元102、存储介质104、和张量遍历单元106。

[0025] 一般的,当处理单元102执行用于访问张量的特定元素的指令时,张量遍历单元106确定该张量的特定元素的地址,使得处理单元102可以访问存储介质104以读取表示该特定元素的值的数据114。例如,程序可以包括嵌套循环并且处理单元102可以执行用于根据与该嵌套循环相关联的当前索引变量值来访问该嵌套循环内的二维数组变量的元素的指令。基于与该嵌套循环相关联的当前索引变量值,张量遍历单元106可以确定偏移值,该偏移值表示距该二维数组变量的第一元素的偏移。然后,处理单元102可以使用该偏移值以从存储介质来访问该二维数组变量的特定元素。

[0026] 处理单元102被配置成处理用于在计算系统100内执行的指令,所述指令包括存储在存储介质104中的指令112或者存储在另一存储设备中的其他指令。处理单元102可以包括一个或多个处理器。存储介质104在计算系统100内存储信息。在一些实施方式中,存储介质104是易失性存储器单元。在一些其他实施方式中,存储介质104是非易失性存储器单元。存储介质104也可以是另一形式的计算机可读介质,诸如软盘设备、硬盘设备、光盘设备、或者磁带设备、闪速存储器或者其他类似的固态存储器设备、或者设备阵列,包括存储区域网络或其他配置中的设备。该指令在由处理单元102执行时,使得处理单元102执行一个或多个任务。

[0027] 张量遍历单元106被配置成确定与一个或多个张量相关联的状态。该状态可以包括循环边界值、当前循环索引变量值、用于计算存储器地址值的维度乘子、和/或用于处置分支循环边界的程序计数器值。张量遍历单元106包括张量状态元素122和算术逻辑单元124。张量状态元素122中的每一个可以是存储元件,例如寄存器或任何其他合适的存储电路。在一些实施方式中,张量状态元素122可以在物理上或逻辑上被布置为不同的群组,以下参照图2A-图2H和图3来对其详细的描述。在一些实施方式中,张量状态元素122的群组可以在物理上或逻辑上被布置为多维数组。例如,张量状态元素122的每一个群组可以在物理上或逻辑上被布置为二维数组。算术逻辑单元124包括被配置成执行算术和逻辑操作的一个或多个算术逻辑单元(ALU)。例如,算术逻辑单元124可以包括数字电路,其被配置成执行加法、减法、乘法、除法、逻辑与、逻辑或、或者任何其他合适的算术和逻辑运算。

[0028] 图2A示出了张量遍历单元的张量状态元素200的示例集合。张量状态元素200可以对应于张量遍历单元106的张量状态元素122。张量状态元素200包括张量索引元素202的群组、张量边界元素204的群组、和维度乘子元素206的群组。张量索引元素202可以在物理上或逻辑上被布置为具有M行和N列的2维数组,其中M和N为大于或等于1的整数。在一些实施方式中,张量索引元素202的每一行可以表示张量的张量索引信息。例如,如果程序定义了两个数组变量V1和V2,则张量遍历单元可以指派行202a和行202b来分别跟踪数组变量V1和V2的张量索引信息。在一些实施方式中,张量索引元素202的每一列可以表示与张量相关联的嵌套循环索引变量值的张量索引信息。例如,如果程序定义了具有三个循环的嵌套循环以用于访问变量V1,其中,该嵌套循环的每个循环由嵌套循环索引变量i、j、和k来索引,张量遍历单元可以指派张量索引元素 $X_{1,1}$ 、 $X_{1,2}$ 、和 $X_{1,3}$ 来分别跟踪嵌套循环索引变量i、j、和k。下面参照图2D-图2H来更具体地描述嵌套索引信息。

[0029] 张量边界元素204可以在物理上或逻辑上被布置为具有与张量索引元素202相同

的维度的2维数组,其中张量边界元素204中的每一个元素具有在张量索引元素202中的对应元素。在一些实施方式中,张量边界元素204的每一行可以表示张量的张量边界信息。例如,如果程序定义了两个数组变量V1和V2,则张量遍历单元可以指派行204a和行204b来分别跟踪数组变量V1和V2的张量边界信息。在一些实施方式中,张量索引元素204的每一列可以表示与张量相关联的嵌套循环索引变量值的张量边界信息。例如,如果程序定义了具有三个循环的嵌套循环以用于访问变量V1,其中,该嵌套循环的每个循环由嵌套循环索引变量i、j、和k来索引,张量遍历单元可以指派张量边界元素Y<sub>1,1</sub>、Y<sub>1,2</sub>、和Y<sub>1,3</sub>来分别跟踪与嵌套循环索引变量i、j、和k相关联的张量边界信息。下面参照图2C来更具体地描述嵌套边界信息。

[0030] 维度乘子元素206可以在物理上或逻辑上被布置为具有与张量索引元素202相同的维度的2维数组,其中维度乘子元素206中的每一个元素具有在张量索引元素202中的对应的元素。在一些实施方式中,维度乘子元素206的每一行可以表示张量的维度乘子信息。例如,如果程序定义了两个数组变量V1和V2,则张量遍历单元可以指派行206a和行206b来分别跟踪数组变量V1和V2的维度乘子信息。在一些实施方式中,维度乘子元素206的每一列可以表示与张量相关联的嵌套循环索引变量值的维度乘子信息。例如,如果程序定义了具有三个循环的嵌套循环以用于访问变量V1,其中,该嵌套循环的每个循环由嵌套循环索引变量i、j、和k来索引,张量遍历单元可以指派维度乘子元素Z<sub>1,1</sub>、Z<sub>1,2</sub>、和Z<sub>1,3</sub>来分别跟踪与嵌套循环索引变量i、j、和k相关联的维度乘子信息。下面参照图2D来更具体地描述维度乘子信息。

[0031] 图2B-图2H示出了张量遍历单元可以如何使用张量状态元素200来处理张量的示例。参照图2B,程序212可以被存储在存储介质104中或能够由处理单元102执行的另一存储介质中。程序212指定具有第一维度——4和第二维度——2的字符数组变量V。程序212指定用于遍历该变量V的嵌套for循环,其中该for循环以由嵌套循环索引变量i跟踪的外循环来遍历V的第一维度并且以由嵌套循环索引变量j跟踪的内循环来遍历V的第二维度。在一些实施方式中,张量状态元素200可以在程序的开始时被初始化。例如,处理器可以执行初始化张量状态元素200的指令“InitializeElements(初始化元素)”。该指令可以是可由处理器执行的指令集合中的硬件指令。在一些实施方式中,在初始化之后,张量状态元素200中的每一个元素被设置为预定值。此处,在初始化之后,张量状态元素200的每一个元素可以被设置为零。

[0032] 图2C图示了根据程序212来设置张量边界元素204中的张量边界信息。例如,处理器可以执行设置张量边界元素204中的张量边界信息的指令“SetTensorBound(设置张量边界)”。在一些实施方式中,张量边界信息表示对应的嵌套循环索引变量的上边界。此处,内嵌套循环索引变量i具有上边界4,并且外嵌套循环索引变量j具有上边界2。在设置张量边界元素204中的张量边界信息后,元素214(即Y<sub>1,2</sub>)被设置为4,并且元素216(即Y<sub>1,1</sub>)被设置为2。

[0033] 图2D图示了根据程序212来设置维度乘子元素206中的维度乘子信息。例如,处理器可以执行设置维度乘子元素206中的维度乘子信息的指令“SetDimensionMultiplier(设置维度乘子)”。在一些实施方式中,维度乘子信息表示张量索引元素202中的对应元素可以被乘的乘子。此处,字符变量V具有第一维度4和第二维度2。如果元素V[0][0]被存储在基础

存储器地址P处并且元素 $V[0][1]$ 被存储在存储器地址P+1处,则元素 $V[1][0]$ 被存储在存储器地址P+2处。因此,元素 $V[i][j]$ 的对应存储器地址是 $P+(i \times 2)+(j \times 1)$ 。因此,外循环的维度乘子信息是2,并且内循环的维度乘子信息是1。在设置维度乘子元素206中的维度乘子信息之后,元素222(即 $Z_{1,2}$ )被设置为2,并且元素224(即 $Z_{1,1}$ )被设置为1。

[0034] 图2E图示了根据程序212来访问元素 $V[0][0]$ 。例如,处理器可以执行定位与所访问的元素相对应的存储器地址的指令“LocateTensor(定位张量)”。在一些实施方式中,该指令可以包括基础存储器地址。例如,指令“LocateTensor”可以包括作为变量V的第一元素的元素 $V[0][0]$ 的存储器地址。在一些实施方式中,该指令可以包括与待访问的张量相对应的行号码。例如,指令“LocateTensor”可以包括与变量V相对应的行号码。此处,行号码为1。

[0035] 在一些实施方式中,响应于接收到该指令,算术逻辑单元(例如,算术逻辑单元106)通过计算张量索引元素202中的每一个元素中所存储的值与对应的维度乘子元素206中所存储的值之间的乘法积的和来确定存储器地址偏移。此处,算术逻辑单元用元素 $Z_{1,N}$ 中所存储的值来乘元素 $X_{1,N}$ 中所存储的值、用元素 $Z_{1,N-1}$ 中所存储的值来乘元素 $X_{1,N-1}$ 中所存储的值、...、用元素 $Z_{1,2}$ 中所存储的值来乘元素 $X_{1,2}$ 中所存储的值、并且用元素 $Z_{1,1}$ 中所存储的值来乘元素 $X_{1,1}$ 中所存储的值。然后,算术逻辑单元对所有相乘的积一起求和以确定与正访问的元素相对应的存储器地址。由于张量索引元素202中的所有值均为零,因此存储器地址偏移为零,如框230所指示。然后,通过将基础存储器地址(即P)添加到存储器地址偏移(即0)而确定存储器地址并且基于所确定的存储器地址来访问存储介质中的所存储数据,处理器能够访问元素 $V[0][0]$ 。

[0036] 图2F图示了根据程序212来访问元素 $V[0][1]$ 。例如,在程序已经迭代过第一内循环之后,嵌套循环索引变量被更新(即 $i=0$ 并且 $j=0$ ),处理器可以在程序进入第二内循环时执行更新张量索引元素202的指令“IterateTensor(迭代张量)”(即 $i=0$ 并且 $j=1$ )。在一些实施方式中,张量遍历单元通过将具有最低索引的张量索引元素递增预定值来更新张量索引元素202。例如,参照图2F,张量遍历单元可以将张量索引元素228(即 $X_{1,1}$ )递增1。在一些其它实施方式中,张量遍历单元可以通过将具有最低索引的张量索引元素递减预定值来更新张量索引元素202。

[0037] 在一些实施方式中,张量遍历单元将张量索引元素202中的所更新值与张量边界元素204中的值进行比较,并且确定是否已经满足阈值。例如,张量遍历单元可以将张量索引元素228中的所更新值(即1)与张量边界元素216中的值(即2)进行比较,并且确定张量索引元素228中的所更新值是否小于张量边界元素216中的值。此处,张量索引元素228中的所更新值小于张量边界元素216中的值,并且张量索引元素228中的所更新值仍然是1。

[0038] 然后,通过执行指令“LocateTensor”来定位与 $V[0][1]$ 相对应的存储器地址,处理器能够访问元素 $V[0][1]$ 。参照图2F,张量索引元素202中除了张量索引元素228外的所有值均为零。张量索引元素228具有值1,并且算术逻辑单元可以确定存储器地址偏移为1,如框232所指示。然后,通过将基础存储器地址(即P)添加到存储器地址偏移(即1)而确定存储器地址并且基于所确定的存储器地址来访问存储介质中的所存储数据,处理器能够访问元素 $V[0][1]$ 。

[0039] 图2G图示了根据程序212来访问元素 $V[1][0]$ 。例如,在程序已经第二次迭代过内循环之后,嵌套循环索引变量被更新(即 $i=1$ 并且 $j=0$ ),并且处理器可以执行更新张量索

引元素202的指令“IterateTensor”。例如,张量遍历单元可以将张量索引元素228(即 $X_{1,1}$ )递增1。张量遍历单元可以将张量索引元素228中的所更新值(即2)与张量边界元素216中的值(即2)进行比较,并且确定张量索引元素228中的所更新值不小于张量边界元素216中的值。在一些实施方式中,响应于确定阈值没有被满足,张量遍历单元可以将所更新值续带(carry over)到下一张量索引元素并且重置当前的张量索引元素。例如,由于张量索引元素228中的所更新值不小于张量边界元素216中的值,因此张量遍历单元可以将张量索引元素226递增1,并且将张量索引元素228重置回0。

[0040] 然后,通过执行指令“LocateTensor”来定位与 $V[1][0]$ 相对应的存储器地址,处理器能够访问元素 $V[1][0]$ 。参照图2G,张量索引元素202中除了张量索引元素226外的所有值为零。张量索引元素226具有值1,并且算术逻辑单元可以确定存储器地址偏移为2,如框234所指示。然后,通过将基础存储器地址(即P)添加到存储器地址偏移(即2)而确定存储器地址并且基于所确定的存储器地址来访问存储介质中的所存储数据,处理器能够访问元素 $V[1][0]$ 。

[0041] 图2H图示了根据程序212来访问元素 $V[3][1]$ 。例如,程序可以继续迭代进行内循环和外循环直到嵌套循环索引变量处于其上边界( $i=3$ 并且 $j=1$ )。通过执行指令“LocateTensor”来定位与 $V[3][1]$ 相对应的存储器地址,处理器能够访问元素 $V[3][1]$ 。参照图2H,张量索引元素202中除了张量索引元素226和228外的所有值为零。张量索引元素226具有值3,并且张量索引元素228具有值1。算术逻辑单元可以确定存储器地址偏移为7,如框236所指示。然后,通过将基础存储器地址(即P)添加到存储器地址偏移(即7)而确定存储器地址并且基于所确定的存储器地址来访问存储介质中的所存储数据,处理器能够访问元素 $V[3][1]$ 。

[0042] 然后,处理器可以执行更新张量索引元素202的指令“IterateTensor”。例如,张量遍历单元可以将张量索引元素228(即 $X_{1,1}$ )递增1。张量遍历单元可以将张量索引元素228中的所更新值(即2)与张量边界元素216中的值(即2)进行比较,并且确定张量索引元素228中的所更新值不小于张量边界元素216中的值。由于张量索引元素228中的所更新值不小于张量边界元素216中的值,因此张量遍历单元可以将张量索引元素226递增1,并且将张量索引元素228重置回0。张量遍历单元可以将张量索引元素226中的所更新值(即 $3+1=4$ )与张量边界元素214中的值(即4)进行比较,并且确定张量索引元素226中的所更新值不小于张量边界元素214中的值。由于张量索引元素228中的所更新值不小于张量边界元素214中的值,因此张量遍历单元可以将下一张量索引元素(即 $X_{1,3}$ )递增1,并且将张量索引元素226重置回0。然而,由于对应的张量边界元素(即 $Y_{1,3}$ )为0,因此阈值未被满足。张量遍历单元将继续重置张量索引元素202的第一行直到该第一行处的全部张量索引元素被重置为0为止。

[0043] 在以上参照图2B-图2H所描述的示例中,对元素V的张量遍历的顺序为 $V[0][0]$ 、 $V[0][1]$ 、 $V[1][0]$ 、 $V[1][1]$ 、 $V[2][0]$ 、 $V[2][1]$ 、 $V[3][0]$ 、和 $V[3][1]$ 。在一些实施方式中,通过将元素214(即 $Y_{1,2}$ )中的张量边界信息设置为2并将元素216(即 $Y_{1,1}$ )中的张量边界信息设置为4,并且将元素222(即 $Z_{1,2}$ )中的维度乘子信息设置为1并将元素224(即 $Z_{1,1}$ )中的维度乘子信息设置为2,程序可以以 $V[0][0]$ 、 $V[1][0]$ 、 $V[2][0]$ 、 $V[3][0]$ 、 $V[0][1]$ 、 $V[1][1]$ 、 $V[2][1]$ 、和 $V[3][1]$ 的顺序来遍历元素V。

[0044] 图3示出了张量遍历单元的张量状态元素300的示例集合。张量状态元素300可以

对应于张量遍历单元106的张量状态元素122或者张量状态元素200。张量遍历单元300包括张量索引元素302的群组、张量边界元素304的群组、维度乘子元素306的群组、以及分支目标元素308的群组。分支目标元素308可以在物理上或逻辑上被布置为2维数组,其中分支目标元素308中的每一个元素具有在张量索引元素302中的对应元素。在一些实施方式中,分支目标元素308的每一行可以表示张量的分支目标信息。在一些实施方式中,分支目标元素308的每一列可以表示与张量相关联的嵌套循环索引变量值的分支目标信息。在一些实施方式中,对于N维张量——其中存在N列的张量索引元素302、张量边界元素304、以及维度乘子元素306,可以存在N+1列的对应的分支目标308,因为存在满足边界阈值的N+1个情况。

[0045] 在一些实施方式中,分支目标信息可以包括在嵌套循环中完成循环迭代时待由处理器执行的指令的存储器地址。例如,当张量索引元素中的值被更新时,处理器可以根据在对应的分支目标元素中所存储的存储器地址来执行指令。参照图3,程序312可以被存储在存储介质104中或者能够由处理单元102执行的另一存储介质中。程序312指定具有第一维度——4和第二维度——2的字符数组变量V。程序312指定用于遍历该变量V的嵌套for循环,其中该for循环以由嵌套循环索引变量i跟踪的外循环来遍历V的第一维度并且以由嵌套循环索引变量j跟踪的内循环来遍历V的第二维度。程序312还指定用于退出内循环的条件,其要求处理器执行minimum函数(最小函数)指令以将嵌套循环索引变量i与号码2进行比较。在一些实施方式中,minimum函数指令的存储器地址可以被存储在分支目标元素324中,使得如果存在从张量索引元素326向张量索引元素328的续带,将由处理器来执行该minimum函数指令以确定内循环的分支条件并且设置该内循环的张量边界信息。类似地,不同指令的存储器地址可以被存储在分支目标元素322中,使得如果在不具有向张量索引元素328的续带的情况下存在张量索引元素326的递增,将由处理器来执行该不同指令。

[0046] 图4是图示了确定多维张量变量的地址的过程400的示例的流程图。过程400可以由一个或多个计算机的系统来执行,例如计算系统100。该系统包括具有多个张量索引元素和多个维度乘子元素的张量遍历单元,其中该维度乘子元素中的每一个元素具有对应的张量索引元素。在一些实施方式中,该系统可以包括多个张量边界元素,其中该多个张量边界元素中的每一个元素可以具有在多个张量索引元素中的对应的张量索引元素。在一些实施方式中,该系统包括一个或多个算术逻辑单元。

[0047] 系统获得用于访问N维张量的特定元素的指令,其中该N维张量具有跨N个维度中的每一个维度布置的多个元素,并且其中N是等于或大于1的整数(402)。例如,处理单元102可以执行用于访问张量的特定元素的指令。

[0048] 在一些实施方式中,指令可以表示用于处理包括第一循环和第二循环的嵌套循环的指令,该第二循环是该第一循环的外循环,其中该第一循环使用第一索引变量来迭代,并且其中该第二循环使用第二索引变量来迭代。例如,程序212可以被存储在存储介质104中或者能够由处理单元102执行的另一存储介质中。程序212可以指定具有第一维度——4和第二维度——2的字符数组变量V。程序212可以指定用于遍历该变量V的嵌套for循环,其中该for循环以由嵌套循环索引变量i跟踪的外循环来遍历V的第一维度并且以由嵌套循环索引变量j跟踪的内循环来遍历V的第二维度。

[0049] 系统使用多个张量索引元素中的一个或多个张量索引元素和多个维度乘子元素中的一个或多个维度乘子元素来确定该特定元素的地址(404)。在一些实施方式中,该特定

元素的地址可以是距N维张量的另一个元素的地址偏移。对于一个或多个张量索引元素中的每一个张量索引元素,系统可以将相应的张量索引元素中所存储的值乘以一个或多个张量索引元素中的对应张量索引元素中所存储的值,以生成相应的相乘值。系统可以通过确定该相乘值的总和来确定地址偏移。例如,算术逻辑单元(例如算术逻辑单元106)可以通过计算张量索引元素202中的每一个元素中所存储的值与对应的维度乘子元素206中所存储的值之间的乘法积的总和来确定存储器地址偏移。

[0050] 在一些实施方式中,系统可以将多个张量索引元素中所存储的值、多个维度乘子元素中所存储的值、和多个张量边界元素中所存储的值设置为初始化值。例如,处理器可以执行初始化张量状态元素200的指令“InitializeElements”。在初始化之后,张量状态元素200中的每一个元素可以被设置为零。

[0051] 在一些实施方式中,系统可以将多个张量边界元素中的第一张量边界元素中所存储的值设置为用于迭代第一循环的第一索引变量的上边界值。系统可以将多个张量边界元素中的第二张量边界元素中所存储的值设置为用于迭代第二循环的第二索引变量的上边界值。例如,处理器可以执行初始化张量状态元素200的指令“InitializeElements”。在初始化之后,张量状态元素200中的每一个元素可以被设置为零。系统可以将多个维度乘子元素中的第一维度乘子元素中所存储的值设置为预定值。系统可以将多个维度乘子元素中的第二维度乘子元素中所存储的值设置为第一索引变量的上边界值。例如,处理器可以执行设置张量边界元素204中的张量边界信息的指令“SetTensorBound”。在设置索引边界元素204中的索引边界信息后,元素214(即 $Y_{1,2}$ )被设置为4,并且元素216(即 $Y_{1,1}$ )被设置为2。

[0052] 在一些实施方式中,系统可以将多个张量索引元素中的第一张量索引元素中所存储的值设置为该第一索引变量的当前值。系统可以将多个张量索引元素中的第二张量索引元素中所存储的值设置为该第二索引变量的当前值。系统可以将第一维度乘子元素中所存储的值乘以第一张量索引元素中所存储的值以获得第一相乘值。系统可以将第二维度乘子元素中所存储的值乘以第二张量索引元素中所存储的值以获得第二相乘值。系统可以通过至少对该第一相乘值和该第二相乘值求和来确定地址。例如,算术逻辑单元用元素 $Z_{1,N}$ 中所存储的值来乘元素 $X_{1,N}$ 中所存储的值、用元素 $Z_{1,N-1}$ 中所存储的值来乘元素 $X_{1,N-1}$ 中所存储的值、...、用元素 $Z_{1,2}$ 中所存储的值来乘元素 $X_{1,2}$ 中所存储的值、并且用元素 $Z_{1,1}$ 中所存储的值来乘元素 $X_{1,1}$ 中所存储的值。然后,算术逻辑单元对所有相乘的积一起求和以确定与正访问的元素相对应的存储器地址。

[0053] 在一些实施方式中,系统可以接收更新第一张量索引元素中所存储的值的指令。例如,处理器可以执行更新张量索引元素202的指令“IterateTensor”。在接收到更新第一张量索引元素中所存储的值的指令后,系统可以确定在第一张量索引元素中所存储的值与第一张量边界元素中所存储的值之间的差满足阈值。响应于确定在第一张量索引元素中所存储的值与第一张量边界元素中所存储的值之间的差满足阈值,系统可以更新该第一张量索引元素中所存储的值。例如,参照图2F,张量遍历单元可以将张量索引元素228(即 $X_{1,1}$ )递增1。张量索引元素228中的所更新值小于张量边界元素216中的值,并且张量索引元素228中所更新值仍然是1。

[0054] 在一些实施方式中,每次第一循环完成时,第一索引变量可以被递增第一递增值。

系统可以将第一张量索引元素中所存储的值递增第一递增值。例如,在程序进入第二内循环(即 $i=0$ 并且 $j=1$ )之后,张量遍历单元可以将张量索引元素228(即 $X_{1,1}$ )递增1。

[0055] 在一些实施方式中,系统可以接收更新在第一张量索引元素中所存储的值的指令。例如,处理器可以执行更新张量索引元素202的指令“IterateTensor”。在接收到更新第一张量索引元素中所存储的值的指令之后,系统可以确定在第一张量索引元素中所存储的值与第一张量边界元素中所存储的值之间的差不满足阈值。响应于确定在第一张量索引元素中所存储的值与第一张量边界元素中所存储的值之间的差不满足阈值,系统可以重置该第一张量索引元素中所存储的值,并且更新第二张量索引元素中所存储的值。例如,张量遍历单元可以将张量索引元素228(即 $X_{1,1}$ )递增1。张量遍历单元可以将张量索引元素228中的所更新值(即2)与张量边界元素216中的值(即2)进行比较,并且确定张量索引元素228中的所更新值不小于张量边界元素216中的值。由于张量索引元素228中的所更新值不小于张量边界元素216中的值,因此张量遍历单元可以将张量索引元素226递增1,并且将张量索引元素228重置回0。

[0056] 在一些实施方式中,每第二循环完成时,第二索引变量可以被递增第二递增值。系统可以将第二张量索引元素中所存储的值递增第二递增值。例如,在程序进入外循环的下一循环(即 $i=1$ 并且 $j=0$ )之后,张量遍历单元可以将张量索引元素226(即 $X_{1,2}$ )递增1。

[0057] 系统输出指示用于访问N维张量的特定元素的所确定地址的数据(406)。例如,处理单元102可以使用存储器地址偏移值来访问存储介质中的N维数组变量的特定元素。

[0058] 在一些实施方式中,系统包括多个分支目标元素,其中该多个分支目标元素中的每一个元素具有在多个张量索引元素中的对应张量索引元素,并且其中该多个分支目标元素中的每一个元素被配置成存储在与相应的分支目标元素对应的循环的迭代完成时执行的指令的地址。例如,张量遍历单元300包括存储分支目标信息的分支目标元素308的群组。分支目标元素可以包括在嵌套循环的循环迭代完成时由处理器执行的指令的存储器地址。

[0059] 本说明书中描述的主题和功能性操作的实施例能够以数字电子电路、以有形体现的计算机软件或固件、以硬件——包括本说明书中公开的结构及其结构等同物、或者以它们中的一个或多个的组合来实现。本说明书中描述的主题的实施例能够被实现为一个或多个计算机程序,即计算机程序指令的一个或多个模块,其被编码在有形非暂时性程序载体上,以用于由数据处理装置执行或控制数据处理装置的操作。替选地或另外地,程序指令能够被编码在人工生成的传播信号上,该信号例如机器生成的电、光、或电磁信号,其被生成以对信息进行编码以用于传输到合适的接收器装置以供数据处理装置执行。计算机存储介质能够是机器可读存储设备、机器可读存储基底、随机或串行存取存储器设备、或它们中的一个或多个的组合。

[0060] 术语“数据处理装置”涵盖用于处理数据的各种装置、设备、和机器,包括例如可编程处理器、计算机、或者多个处理器或计算机。该装置能够包括专用逻辑电路,例如FPGA(现场可编程门阵列)或ASIC(专用集成电路)。除了硬件之外,该装置还能够包括创建用于所讨论的计算机程序的执行环境的代码,例如构成处理器固件、协议栈、数据库管理系统、操作系统、或它们中的一个或多个的组的代码。

[0061] 计算机程序(也称为程序、软件、软件应用、模块、软件模块、脚本、或代码)能够以任何形式的编程语言编写,包括编译或解释语言、声明性或过程性语言,并且其能够以任何



形式部署,包括作为独立程序或作为适于在计算环境中使用的模块、组件、子例程、或其它单元。计算机程序可以但不必对应于文件系统中的文件。程序能够被存储在保持其他程序或数据——例如存储在标记语言文档中的一个或多个脚本——的文件的一部分中、专用于所讨论的程序的单个文件中、或者多个协作文件——例如存储一个或多个模块、子程序、或代码部分的文件中。计算机程序能够被部署为在一个计算机上或在位于一个地点或跨多个地点分布并通过通信网络互连的多个计算机上执行。

[0062] 本说明书中所描述的过程和逻辑流程能够由一个或多个可编程计算机来执行,所述可编程计算机执行一个或多个计算机程序以通过在输入数据上操作并且生成输出来执行功能。该过程和逻辑流程也能够由专用逻辑电路执行并且装置也能够被实现为专用逻辑电路,所述专用逻辑电路例如FPGA(现场可编程门阵列)、ASIC(专用集成电路)、或者GPGPU(通用图形处理单元)。

[0063] 适合于执行计算机程序的处理器包括例如通用微处理器或专用微处理器或者这两者,或者任何其他种类的中央处理单元。通常,中央处理单元将从只读存储器或随机存取存储器或这两者接收指令和数据。计算机的必需元件是用于实施或执行指令的中央处理单元和用于存储指令和数据的一个或多个存储器设备。通常,计算机还将包括用于存储数据的一个或多个大容量存储设备——例如磁盘、磁光盘、或光盘,或者可操作地耦合以从其接收数据或向其传输数据、或者这两者。然而,计算机不需要具有这样的设备。此外,计算机能够被嵌入另一设备中,例如移动电话、个人数字助理(PDA)、移动音频或视频播放器、游戏控制台、全球定位系统(GPS)接收器、或便携式存储设备——例如通用串行总线(USB)闪存驱动器,仅举数例。

[0064] 适合于存储计算机程序指令和数据的计算机可读介质包括所有形式的非易失性存储器、介质和存储器设备,包括例如半导体存储器设备,例如EPROM、EEPROM、和闪速存储器设备;磁盘,例如内部硬盘或可移动盘;磁光盘;以及CD ROM和DVD-ROM盘。处理器和存储器能够由专用逻辑电路补充或并入专用逻辑电路中。

[0065] 为了提供与用户的交互,本说明书中描述的主题的实施方式能够被实现在具有以下的计算机上:用于向用户显示信息的显示设备——例如CRT(阴极射线管)或LCD(液晶显示器)监视器;以及键盘和指示设备——例如鼠标或轨迹球,用户通过其能够向计算机提供输入。其他种类的设备也能够被用于提供与用户的交互;提供给用户的反馈能够是任何形式的感官反馈,例如视觉反馈、听觉反馈、或触觉反馈;并且能够接收来自用户的处于任何形式的输入,包括声音、语音、或触觉输入。此外,计算机能够通过向用户使用的设备发送文档以及从其接收文档来与该用户交互,例如通过响应于从用户的客户端设备上的web浏览器接收的请求而向该web浏览器发送网页。

[0066] 在本说明书中描述的主题的实施例能够被实现在计算系统中,该计算系统包括例如作为数据服务器的后端组件,或者包括例如应用服务器的中间件组件,或者包括例如具有图形用户界面或Web浏览器的客户端计算机的前端组件——用户通过该Web浏览器能够与本说明书中所描述的主题的实施方式交互,或者包括一个或多个这样的后端组件、中间件组件、或前端组件的任何组合。系统的组件能够通过例如通信网络的任何形式或介质的数字数据通信来互连。通信网络的示例包括局域网(“LAN”)和广域网(“WAN”),例如互联网。

[0067] 计算系统能够包括客户端和服务器。客户端和服务器通常彼此远离并且一般通过



通信网络交互。客户端和服务器的关系依靠在相应计算机上运行并且彼此具有客户端-服务器关系的计算机程序而产生。

[0068] 虽然本说明书包含许多具体实施方式细节,但是这些不应被解释为对任何发明或可以要求保护的内容的范围的限制,而是应当被解释为对特定发明的特定实施例特有的特征的描述。本说明书中在单独实施例的场境下所描述的某些特征也能够单个实施例中组合实现。相反,在单个实施例场境下所描述各个特征也能够多个实施例中单独地或以任何合适的子组合来实现。此外,虽然上面可能将特征描述为以某些组合来起作用并且甚至最初如此要求保护,但是在一些情况下来自所要求保护的组合的一个或多个特征能够从组合中被删去,并且所要求保护的组合可以涉及子组合或子组合的变型。

[0069] 类似地,虽然在附图中以特定顺序描绘了操作,但是这不应被理解为需要以所示的特定顺序或以连续顺序来执行这样的操作、或者需要执行所有所图示的操作才能达到期望的结果。在某些情况下,多任务和并行处理可以是有利的。此外,上述实施例中的各种系统组件的分离不应被理解为在所有实施例中都需要这样的分离,并且应当理解,所描述的程序组件和系统通常能够一起集成在单个软件产品中或封装到多个软件产品内。

[0070] 已经描述了本主题的特定实施例。其他实施例落入所附权利要求书的范围内。例如,权利要求书中所记载的动作能够以不同的顺序执行并且仍然达到期望的结果。作为一个示例,附图中描绘的过程不一定需要所示的特定顺序或连续顺序来达到期望的结果。在某些实施方式中,多任务和并行处理可以是有利的。

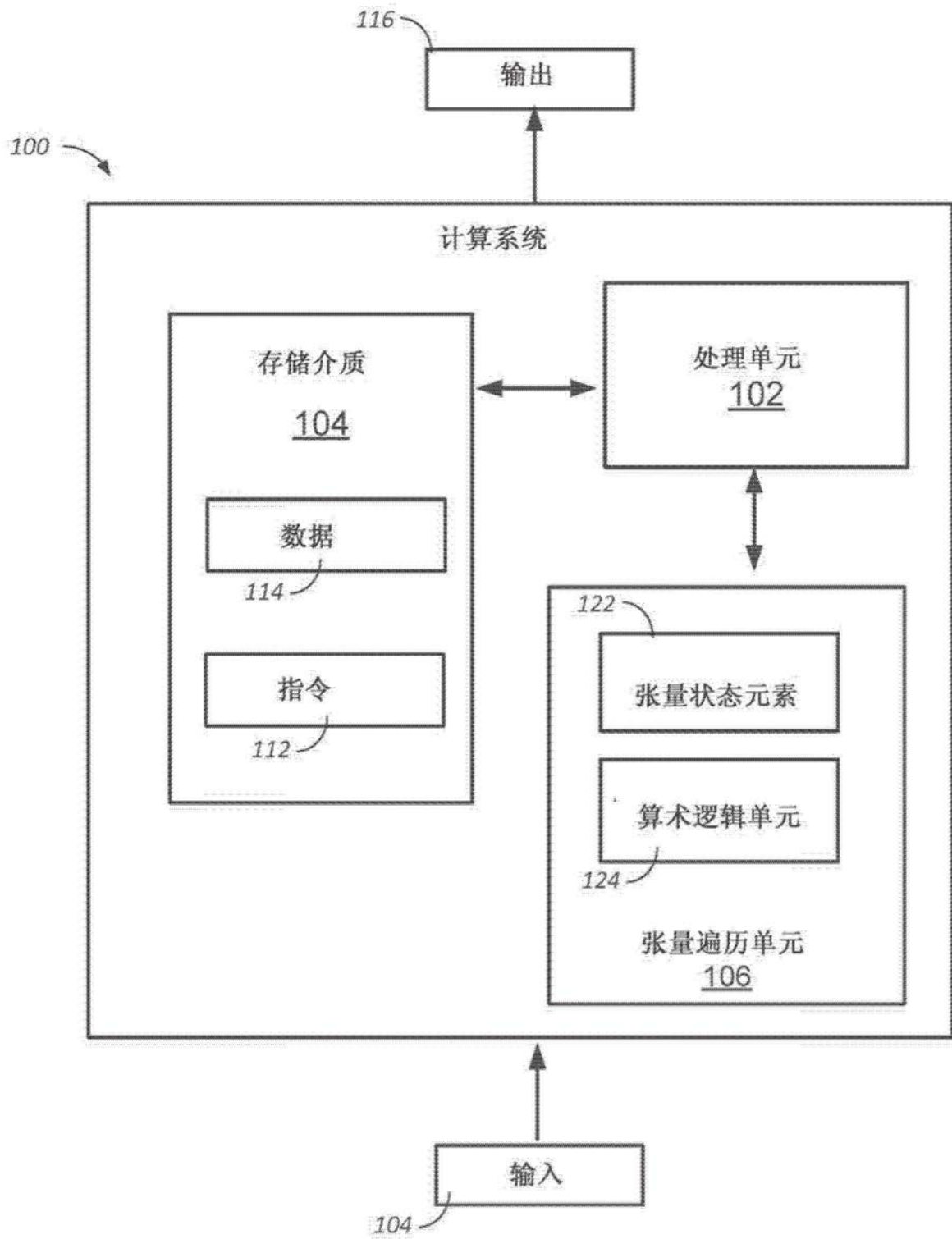


图1

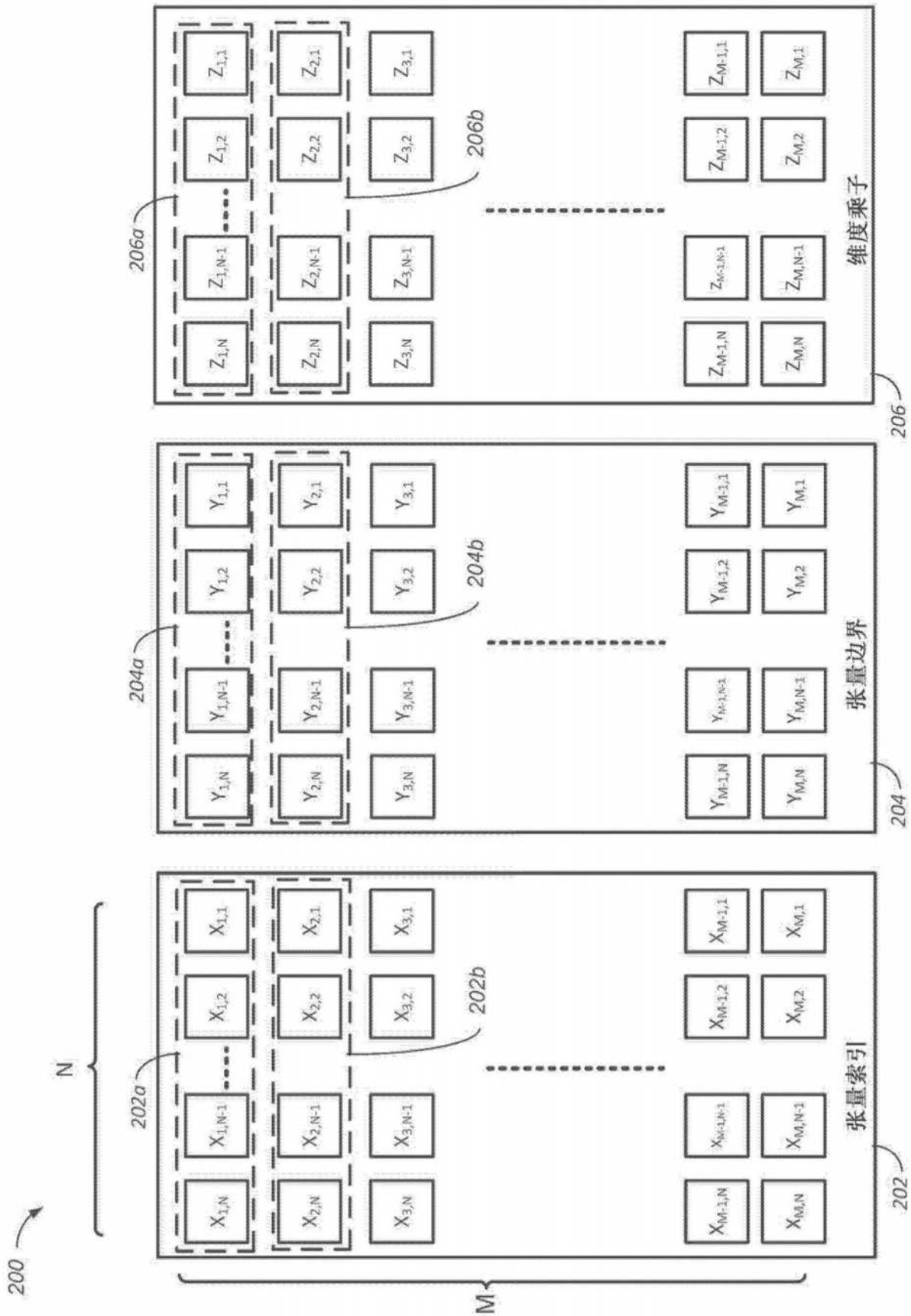


图2A

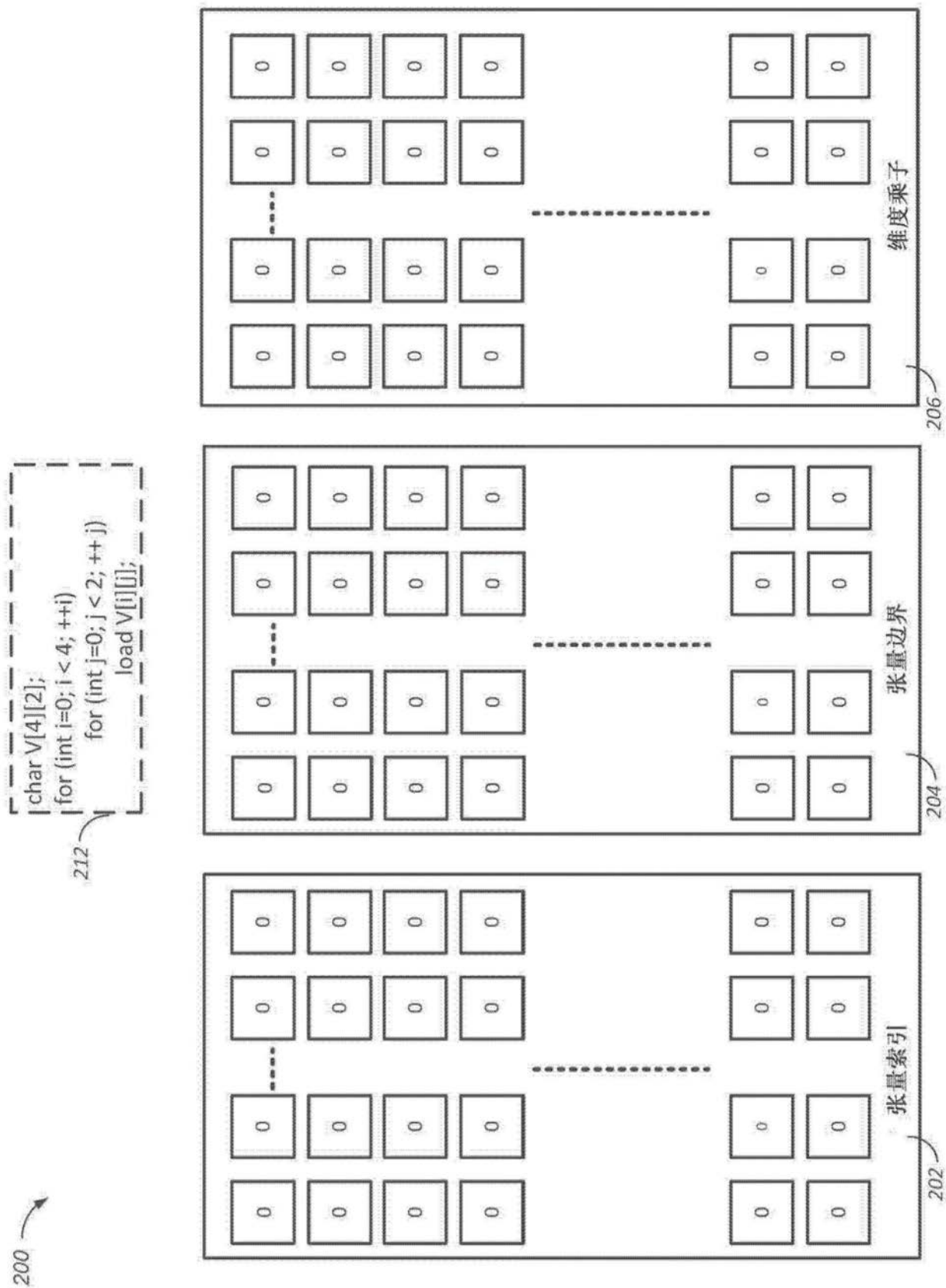


图2B

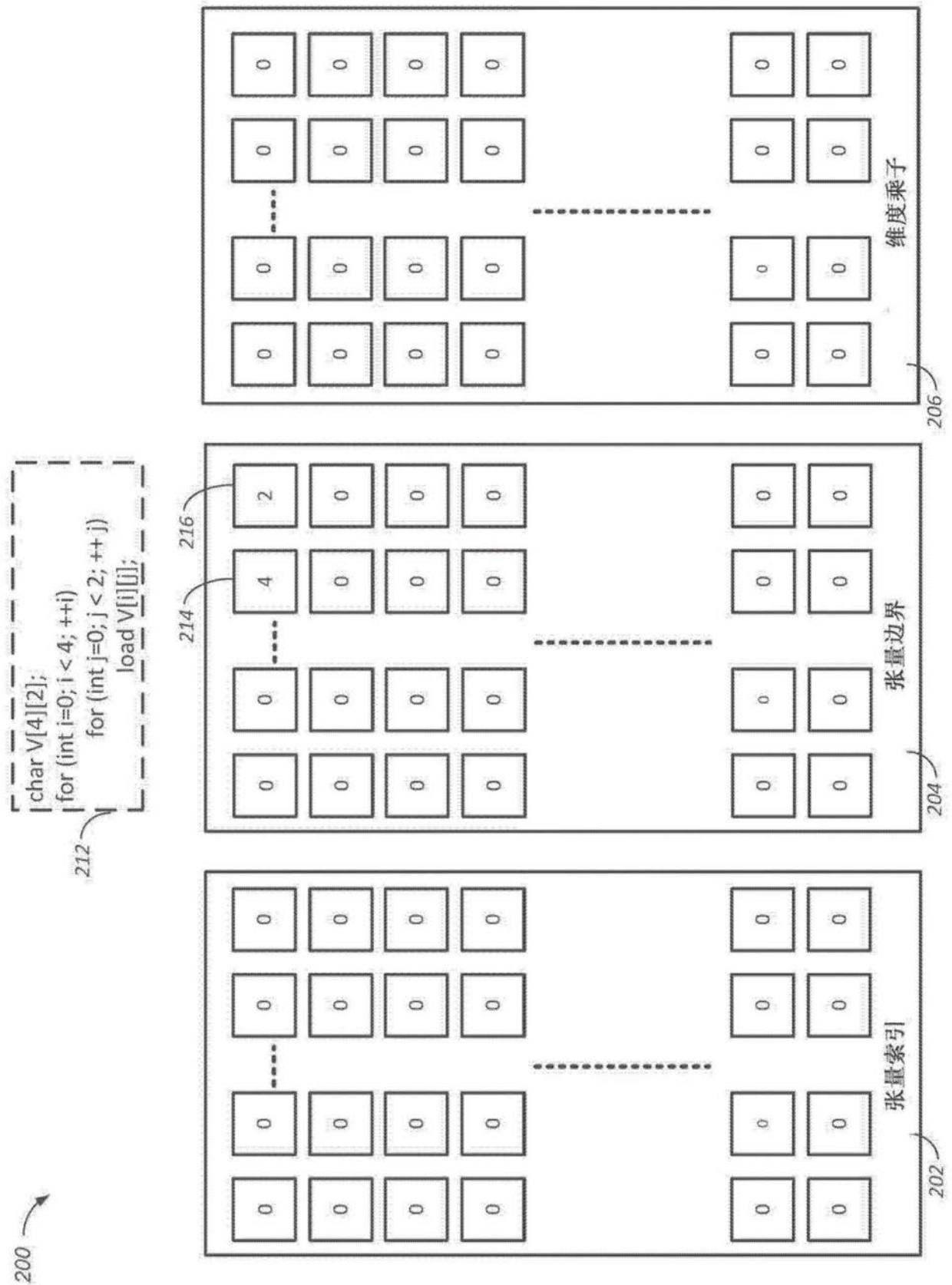


图2C

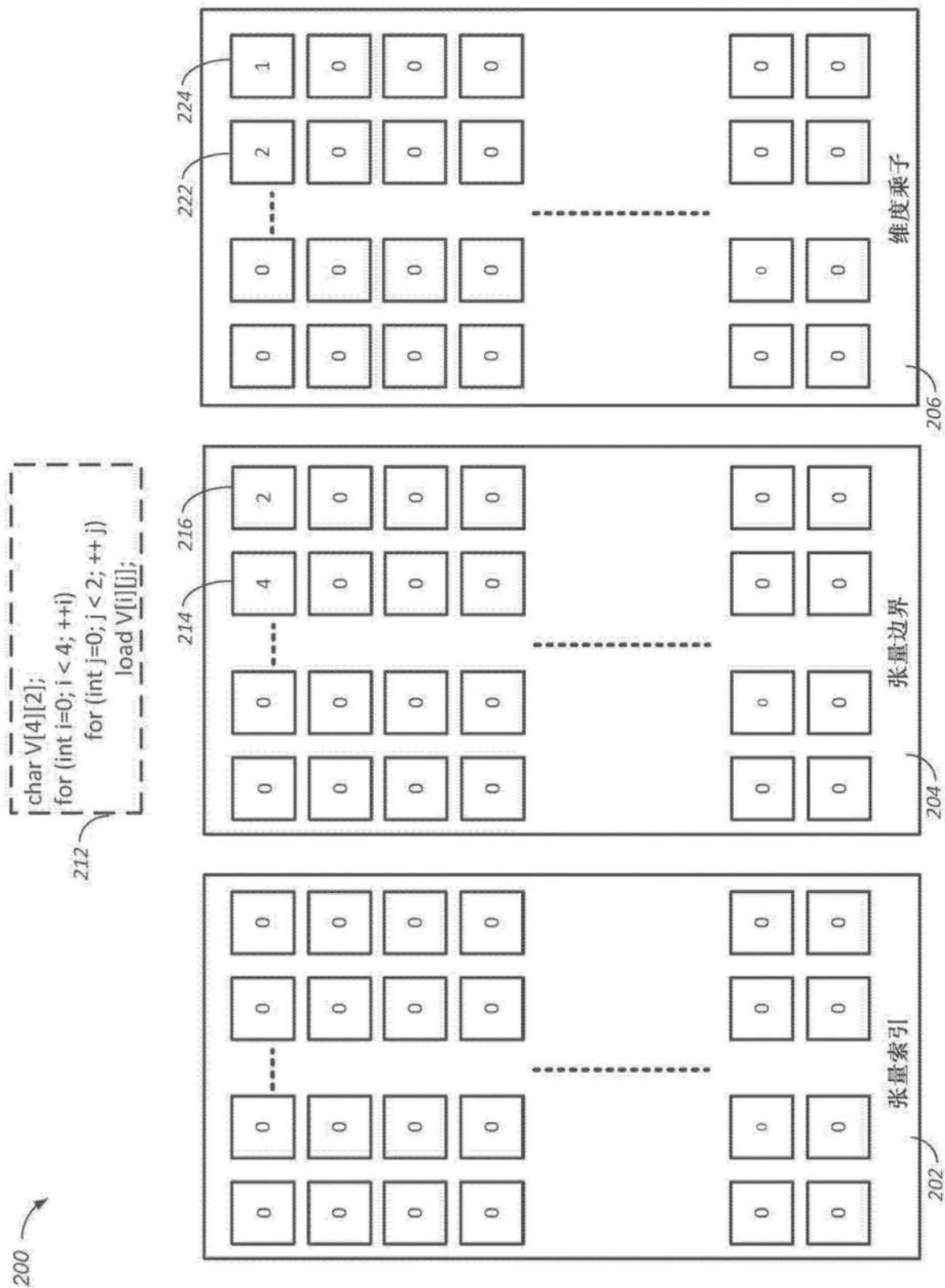


图2D



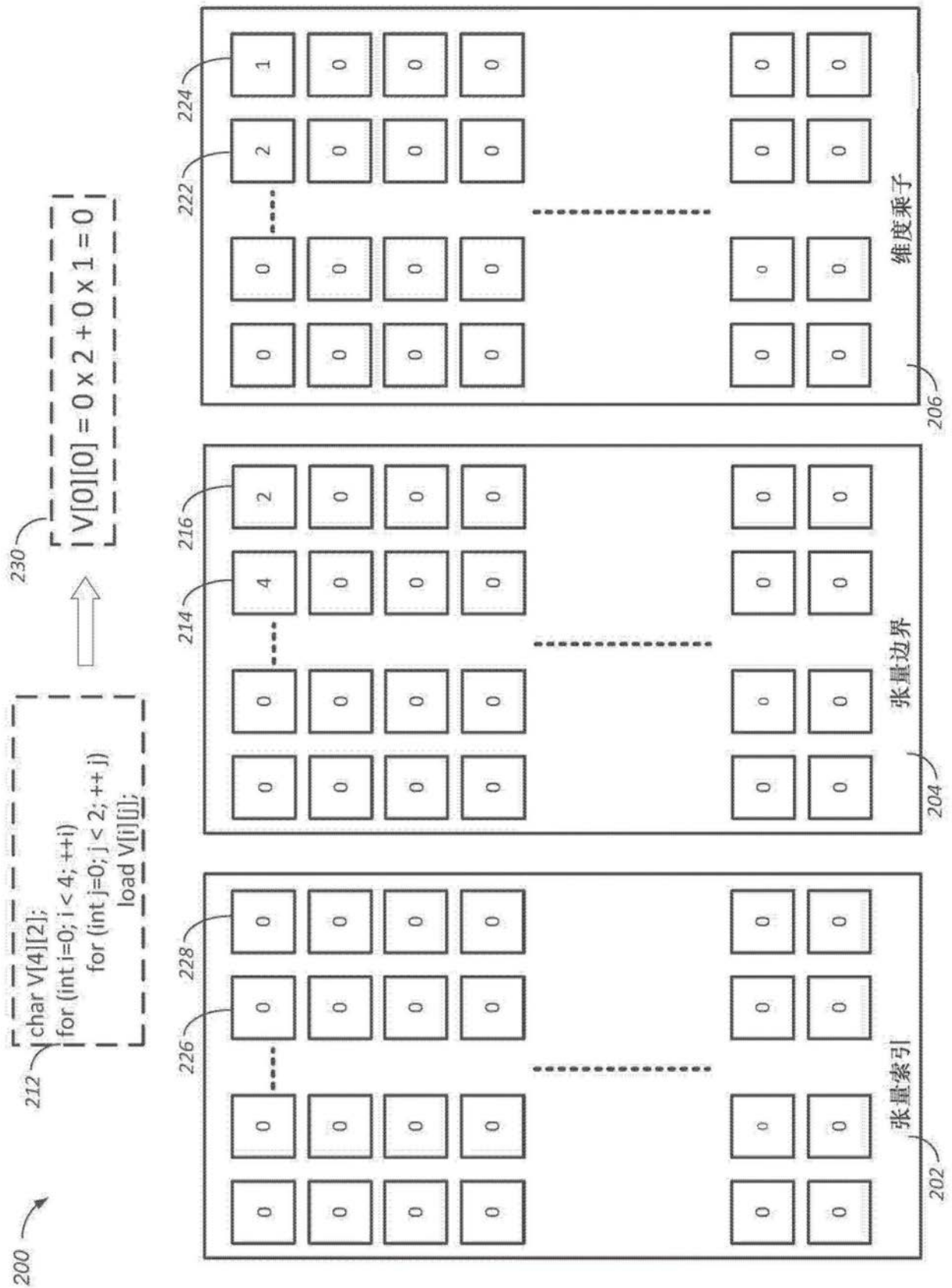


图2E

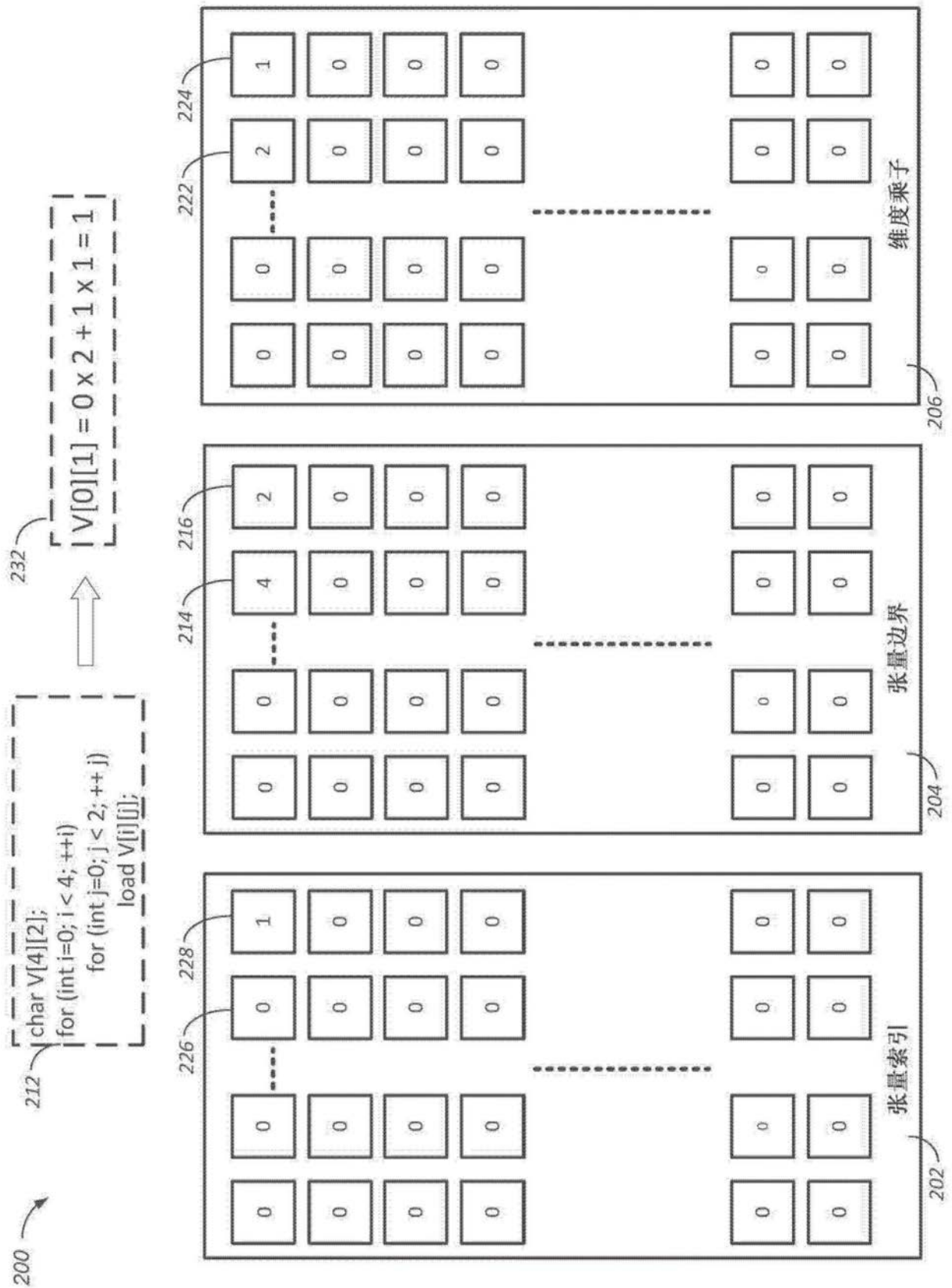


图2F



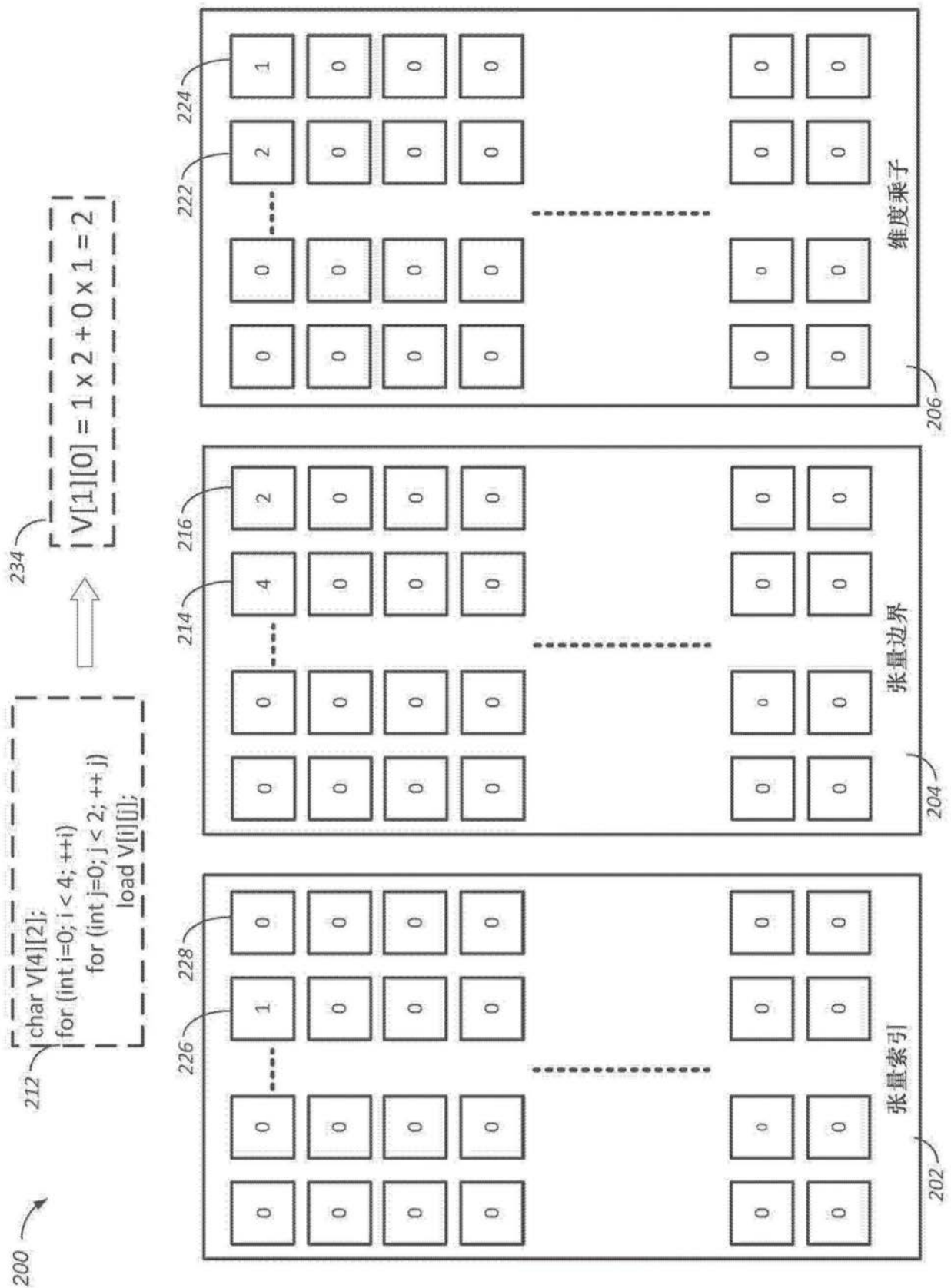


图20

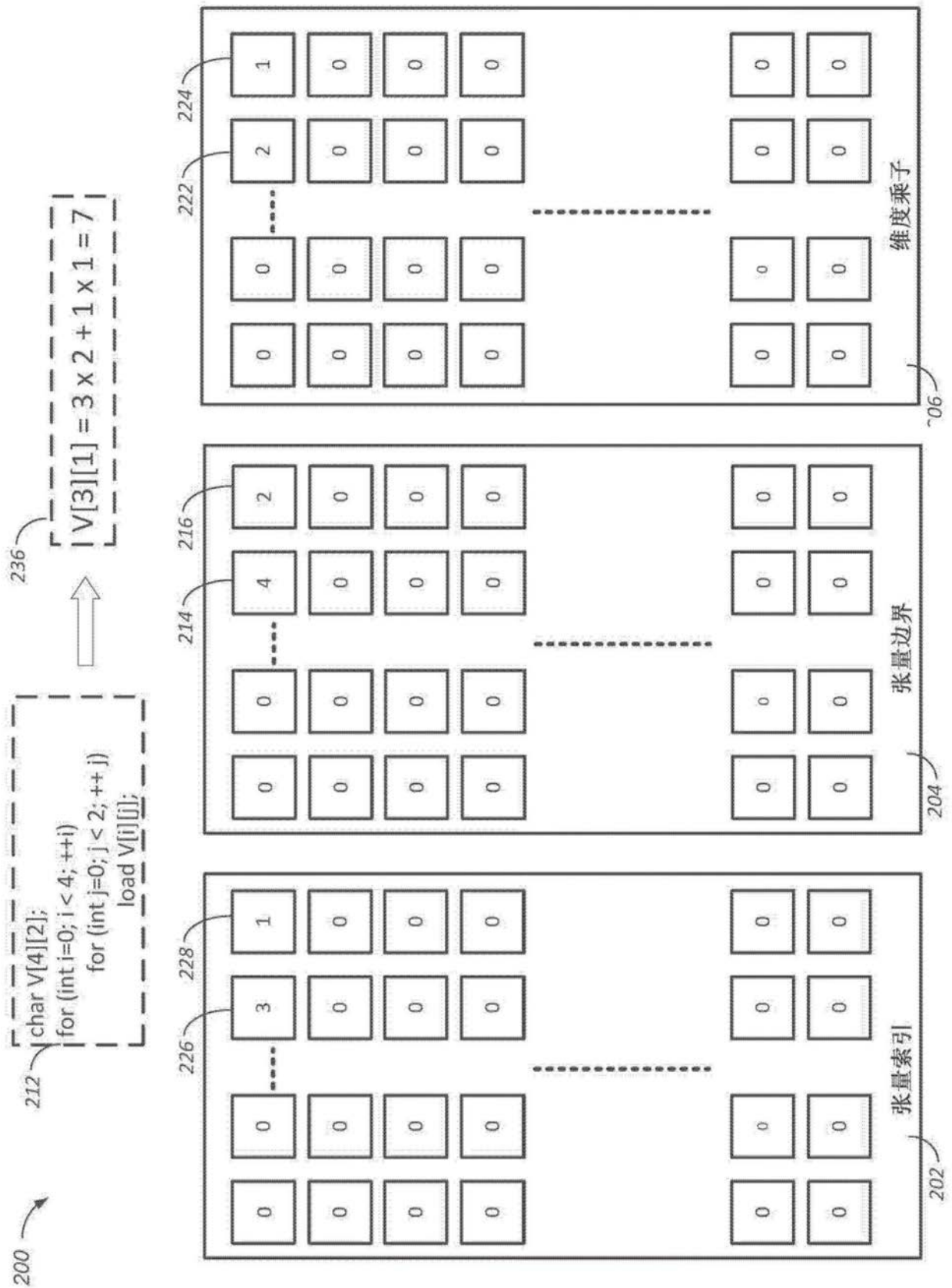


图2H

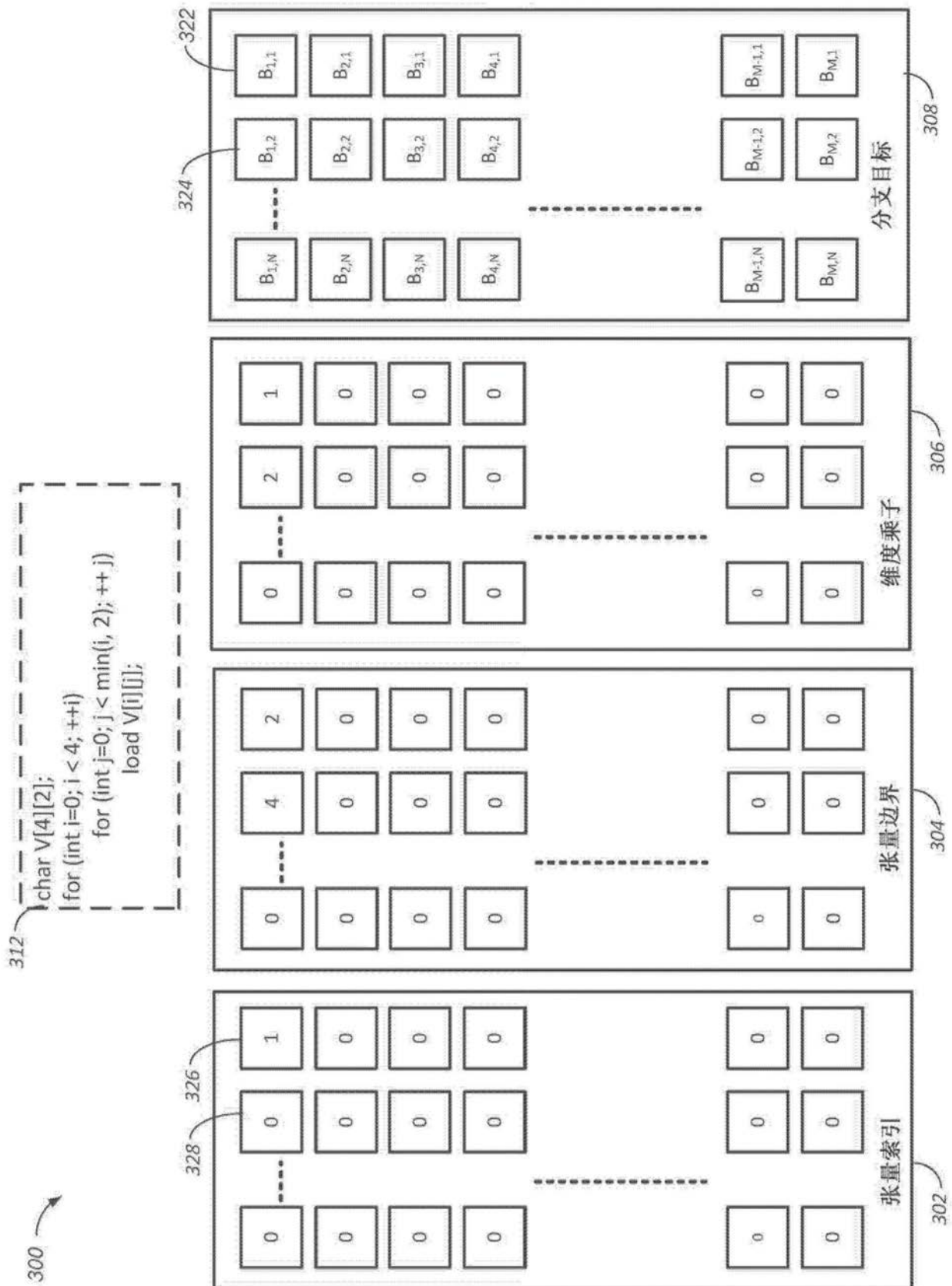


图3

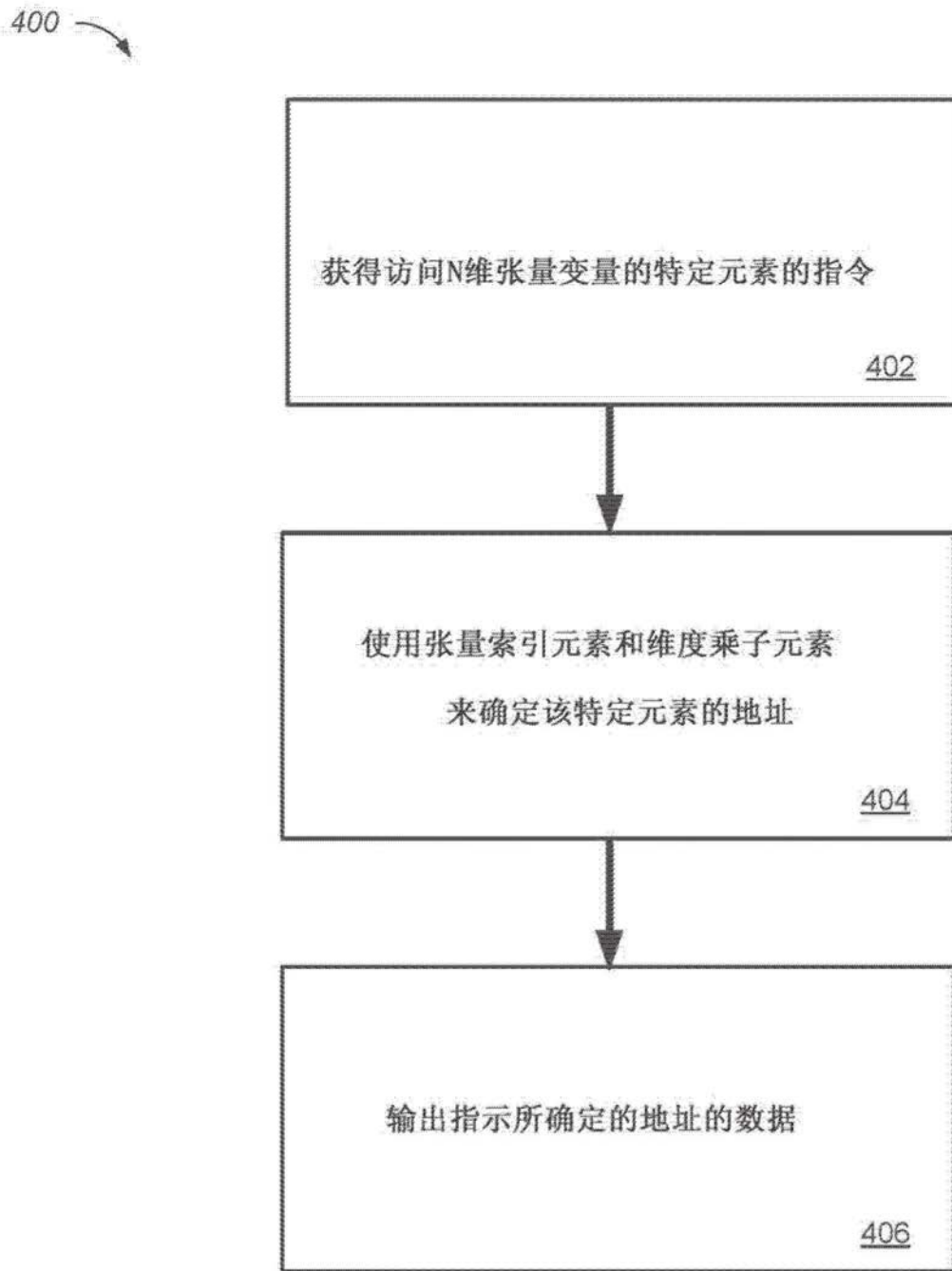


图4