

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G06F 9/302

G06F 9/315



[12] 发 明 专 利 说 明 书

[21] ZL 专利号 01810648.X

[45] 授权公告日 2005 年 4 月 13 日

[11] 授权公告号 CN 1196998C

[22] 申请日 2001.8.21 [21] 申请号 01810648.X

[30] 优先权

[32] 2000.10.4 [33] GB [31] 0024311.3

[86] 国际申请 PCT/GB2001/003744 2001.8.21

[87] 国际公布 WO2002/029553 英 2002.4.11

[85] 进入国家阶段日期 2002.12.3

[71] 专利权人 ARM 有限公司

地址 英国剑桥郡

[72] 发明人 D·H·赛姆斯 D·J·西尔

审查员 袁文婷

[74] 专利代理机构 中国专利代理(香港)有限公司

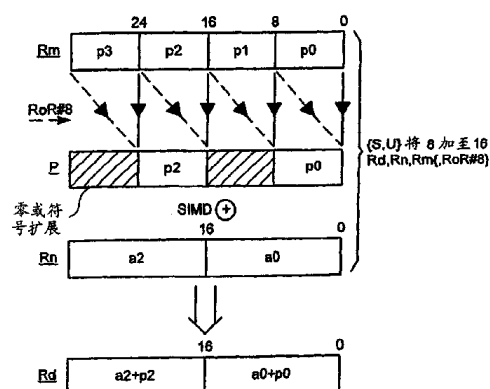
代理人 程天正 王 勇

权利要求书 2 页 说明书 6 页 附图 4 页

[54] 发明名称 单指令多数据处理

[57] 摘要

一种数据处理系统配备有指令(ADD8T016),该指令用符号或零扩展对数据字的非邻接部分解包并将其与响应同一指令而执行的诸如加法之类的单指令多数据的运算操作相结合。所述指令能很好地在具有数据路径(2)的系统内使用,所述数据路径包括位于运算电路(8)上游的位移电路(6)。



1. 一种用于数据处理的设备，该设备包括：

位移电路；

运算电路；以及

5 指令解码器，它响应指令而控制上述位移电路和运算电路去对第一输入数据字和第二输入数据字进行操作，其中，所述操作能产生通过下列措施给出的值：

选择上述第一输入数据字的多个非相邻的多位部分，从而形成多个多位部分，每个部分的位长度均为 A；

10 可选地按共用的位移量将上述多个非相邻的多位部分位移成位移后的位部分；

使上述多个非相邻的多位部分中的每一个部分都从位长度 A 提升至位长度 B，以形成多个提升了的多位部分，使得所述提升了的多位部分可以相邻接，从而形成提升了的数据字 P；以及

15 将来自上述提升的数据字 P 和所述第二输入数据字的位长为 B 的相应位位置部分用作输入操作数，进行多种独立的运算操作，以形成结果数据字。

2. 如权利要求 1 的设备，其特征在于， $B=2*A$ 。

3. 如权利要求 1 的设备，其特征在于，所述多个多位部分位移成
20 位移后的位部分，以使得最低位位置的多位部分从第零个位位置开始扩展。

4. 如权利要求 1 的设备，其特征在于，将上述多位部分从位长度为 A 提升为位长度为 B 包括下列措施之一：

用符号将上述多位部分扩展至 B 位长；以及

25 用零将上述多位部分扩展至 B 位长。

5. 如权利要求 1 的设备，其特征在于，所述多个独立的运算操作是独立的加法操作。

6. 如权利要求 2 的设备，其特征在于，所述第一数据字和所述第二数据字的每一个都具有 C 位位长， $C=N*B$ ，其中，N 是大于 1 的整数。

30 7. 如权利要求 6 的设备，其特征在于， $C=B*2$ 。

8. 如权利要求 1 的设备，其特征在于， $B=16$ 和 $A=8$ 。

9. 如权利要求 1 的设备，其特征在于，所述共用位移量是 $B-A$ 。

10. 如权利要求 1 的设备, 其特征在于, 所述指令是单指令多数数据指令。

11. 如权利要求 1 的设备, 其特征在于, 所述指令将数据值解包操作与运算操作结合起来。

5 12. 如权利要求 1 的设备, 其特征在于, 所述位移电路在所述设备的数据路径中位于所述运算电路的上游。

13. 如权利要求 1 的设备, 其特征在于, 一个能进行操作以便将所述多位部分从位长 A 提升到位长 B 的提升电路以与所述位移电路的一部分相并行的方式被设置, 在执行所述指令时, 所述位移电路能进行
10 操作, 以便为经过该位移电路的数据值提供与所述位移电路在执行其它指令时提供的共用位移量相比时其范围是受限的共用位移量。

14. 一种数据处理的方法, 该方法包括下列步骤:

对指令进行解码并执行, 所述指令能产生通过下列步骤给出的结果:

15 选择第一数据字的多个非相邻的多位部分, 从而形成多个位长度为 A 的多位部分;

可选地按共用的位移量将所述多个非相邻的多位部分位移成位移后的位部分;

使所述多个非相邻的多位部分中的每一个部分都从位长度 A 升至
20 位长度 B, 以形成多个提升了的多位部分, 使得所述提升了的多位部分可以相邻接, 从而形成提升了的数据字 P; 以及

将来自所述提升的数据字 P 和第二数据字的位长为 B 的相应位位置部分用作输入操作数, 进行多种独立的运算操作, 以形成结果数据字。

单指令多数据处理

技术领域

- 5 本发明涉及数据处理系统领域。具体地说，本发明涉及这样的数据处理系统，在其中需要提供单指令多数据类型的操作。

背景技术

- 单指令多数据操作是一种周知的技术，利用这种技术，可根据单指令来处理的数据字事实上表示了这些数据字中的多数据值，其中特定的处理可根据相应的数据值来独立地进行。这种类型的指令可提高数据处理系统运行的效率并在减少代码数量和提高处理操作速度方面特别有用。这种技术通常但不排它地应用于这样的领域：对表示诸如在数字信号处理应用中的物理信号的数据值进行处理。

- 在扩展数据处理系统的数据处理能力时，一个重要的考虑是能用
15 来支持额外处理能力的任何规模、复杂性、成本和能耗开销的程度。能增加处理能力同时减少所导致的额外开销的措施是很有优势的。

发明内容

- 从一个方面来看，本发明提供了用于数据处理的设备，该设备包括：位移电路；运算电路；以及，指令解码器，它响应指令而控制上述位移电路和运算电路去对数据字 R_n 和数据字 R_m 进行操作，其中，
20 所述操作能产生通过以下步骤得到的值：选择上述数据字 R_m 的多个非相邻的多位部分，从而形成多个多位部分，每个部分的位长度均为 A ；可选地按共用的位移量将上述多个多位部分位移成位移后的位部分；使上述多个多位部分中的每一个部分都从位长度 A 升至位长度 B ，以形成多个提升了的多位部分，使得所述提升了的多位部分可以相邻接，
25 从而形成提升了的数据字 P ；以及，把来自上述提升的数据字 P 和数据字 R_n 的位长为 B 的相应位位置部分用作输入操作数，以便进行多种独立的运算操作，从而形成结果数据字 R_d 。

- 本发明提供了数据处理系统中的一种新数据处理指令，该指令可以
30 把数据字中保存的数据值进行解包并且对解包后的数据值执行单指令多数据型的运算操作。本发明认识到，与对相邻数据值解包的通常的解包指令相比，通过解包，能以更少的额外开销实现数据字中非相

邻的数据值。具体地说，可避免需要额外的数据路径，这些路径会使先前相邻的数据值的位位置分散。相反，例如可以使用现有的掩蔽和字位移电路。此外，简化解包功能使得单指令还能在不产生处理周期限制问题的情况下根据操作数提供运算操作。

5 尽管在总体形式上本发明可应用于选择与被提升的长度相比的任意长度的非相邻多位部分，但是，特别有效且方便的实现形式是这样的，其中，选定的多位部分是提升的多位部分的长度的一半，并且，提升的多位部分可在提升的数据字内相邻接，以便使得长度与输入给上述操作的数据字的长度相等。

10 应该注意，可按多种不同的方式来提升选定的多位部分的长度。两种特别有用的提升方式是通过符号扩展或者通过先导零扩展。

 应该注意，与解包相结合的运算操作可呈多种不同的形式。但是，最佳实施例是这样的，其中，运算操作是独立于相应的提升的多位部分而执行的加法操作。这种指令在多种实际数据处理环境中（诸如作为 MPEG 运动补偿计算的一个部分在计算绝对差之和时）特别有用。

15 如前所述，本发明能以更有效的方式使用数据处理系统中的现有处理资源。这在下面的系统中特别是这样，在这种系统中，位移电路在数据路径中设置在运算电路的上游。这种结构能在运算操作之前用任何可选的位移来进行解包。

20 在最佳实施例中，为了在不增加额外处理周期时间限制的情况下提供预定的功能，与位移电路的一部分相并行地设置一个负责（例如通过符号扩展或先导零扩展）提升选定的多位部分的长度的提升电路，并且，限制所指定的共用位移量的范围，使得位移电路的第一部分可与提升电路结合使用，以便在不使数据值传递所花费的时间延长超过已为在其它操作中经过整个位移电路传递所留出的时间的情况下执行所需的操作。

 从另一个方面来看，本发明提供了一种数据处理的方法，该方法包括下列步骤：对指令进行解码并执行，所述指令能产生通过下列措施给出的结果：选择上述数据字 R_m 的多个非相邻的多位部分，从而形成多个位长度为 A 的多位部分；可选地按共用的位移量将上述多个多位部分位移成位移后的位部分；使上述多个多位部分中的每一个部分都从位长度 A 升至位长度 B ，以形成多个提升了的多位部分，使得所述

提升了的多位部分可以相邻接，从而形成提升了的数据字 P；以及，将来自上述提升的数据字 P 和数据字 Rn 的位长为 B 的相应位位置部分用作输入操作数，进行多种独立的运算操作，以形成结果数据字 Rd。

本发明还提供了一种诸如包括具有上述操作形式的数据处理指令之类的计算机程序产品，它存储有计算机程序，该程序根据上述技术控制通用计算机。

发明内容

以下仅以举例的方式参照附图说明本发明的实施例，附图中：

图 1 概略地示出了第一 SIMD 型数据处理指令的活动；

10 图 2 概略地示出了能很好适用于执行图 1 的数据处理指令的处理设备中的数据路径；

图 3 和 4 概略地示出了另一种 SIMD 型数据处理指令的两种变化形式；

15 图 5 概略地示出了能很好适用于执行图 3 和 4 的数据处理指令的处理设备中的数据路径。

具体实施方式

图 1 示出了第一 SIMD 型数据处理指令的活动，该指令称为 ADD8T016。这一指令应用于有符号和无符号变量，这些变量对应于增加给各输入操作数数据字的选定部分的前面的扩展部的性质，就好像将其长度扩展成所执行的处理的一部分。第一输入操作数数据字存储在数据处理设备的寄存器 Rm 内。数据字是由四个 8 位部分 p0、p1、p2 和 p3 构成的。依照在指令中是否指定了 8 位部分的右向旋转操作，从寄存器 Rm 内的输入数据字中选出多位部分 p0 和 p2 或者多位部分 p1 和 p3。如果需要的话，可选的右向旋转操作还可以是 16 和 24 的量。这就能有效地交换高和低阶部分。图 1 所示的实例示出了在未旋转变量中的选定的非相邻部分 p0 和 p2 以及用虚线所示的其它变量。

在选定了多位部分时，用零或符号扩展部将各部分长度从 8 位升至 16 位。图中所示的提升后的数据字 P 的阴影部分表示上述扩展部分。

30 第二输入数据字存储在寄存器 Rn 内并包括两个 16 位数据值。所示的示例执行单指令多数据加法操作，从而，将扩展的 p0 值加至 Rn 的低 16 位值 a0，同时，将扩展的 p2 值增加至 Rn 值的高 16 位部分 a2。

这种类型的加法可认为是全宽度加法，其中进位链在结果的第 15 和第 16 位之间断开。应该注意，也可以执行诸如例如 SIMD 减法之类的其它 SIMD 运算操作。

5 图 1 的指令所生成的输出结果数据字会在低 16 位上产生 p_0 和 a_0 的和，而高 16 位则包含 p_2 和 a_2 的和。上述指令在确定相应数据值之绝对差的和的操作方面特别有用，从而， a_0 和 a_2 表示累积值，值 p_0 至 p_3 表示诸如像素差值之类的信号差值的各个绝对值。通常在 MPEG 运动估算处理过程中需要这种类型的操作，按高速执行这种类型的操作能力具有很强的优点。

10 图 2 示出了可用于实现图 1 的指令的数据处理系统的示例性数据路径 2。寄存器库 4 含有要加以处理的 32 位数据字。从上述寄存器库中读出存储在 R_m 和 R_n 中的输入操作数数据字，将结果数据字写回寄存器库 4 中的寄存器 R_d 内。数据路径 2 包括位移电路 6 和加法器电路 8。上述系统所提供的多种其它数据处理指令按多种不同的方式使用位移电路 6 和加法器电路 8。这种数据路径 2 被仔细地设计，使得数据值
15 经由位移电路 6 和加法器电路 8 传递所花费的时间能很好地与数据处理周期时间相匹配。在系统中很充分地使用数据路径 2 的硬件资源，其中，这些资源对经由数据路径 2 传递的各数据字的大部分来说都是活动的。与位移电路 6 的低位部分相并行地设置有符号/零扩展和掩蔽
20 电路 10。多路复用 12 选择全位移电路 6 的输出或符号/零扩展和掩蔽电路 10 的输出作为输入给加法器电路 8 的输入之一。向加法器 8 的其它输入是 R_n 的输入操作数数据字。

在执行图 1 的指令时，将 R_m 的输入操作数数据字提供给位移电路 6，在该电路中，根据在指令中是否指定了参数而使 8 位位置的可选右
25 向位移作用于数据字。也可以进行 16 和 24 位位置的可选右向旋转。在以多层次的多路复用器基础的位移器中，可从位移电路 6 的第一部分相对简单地提供这种严格的概率位移（例如就 32 位系统而言，第一层多路复用器可提供 16 位的位移，第二层多路复用器提供 8 位位移）。因此，可从经由位移电路 6 的局部路径中抽出以可选方式位移指定量
30 的值，并将其提供给符号/零扩展和掩蔽电路 10。电路 10 可操作以掩蔽掉 R_m 的可能位移的输入操作数数据字的未选择的多位部分，并且用这些数据字的相应选择的多位部分的零或符号扩展部分来代替上述被

掩蔽掉的部分。符号/零扩展和掩蔽电路 10 的输出经由多路复用器 12 传至加法器电路 8 的第一输入。加法器电路 8 的第二输入是 R_n 的输入操作数数据字。加法器电路 8 对其输入进行 SIMD 加法(即两个并行的 16 位加法, 进位链在位位置 15 和 16 之间有效断开)。将加法器电路 8 5 的输出写回寄存器库 4 的寄存器 R_d 内。

作为另一种形式, 符号/零扩展和掩蔽电路 10 可将 R_m (未旋转) 作为其输入, 然后该电路自己对四个可能的符号位进行 0、8、16 或 24 的旋转, 并形成掩蔽。位移电路 6 以与使 R_m 的整个 32 位位移相并行的方式进行操作。

10 图 3 和 4 说明了半字打包 SIMD 型指令的两种变化形式。图 3 的 PKHTB 指令取存储在寄存器 R_n 内的一个输入操作数数据字的定长的头一半以及存储在寄存器 R_m 中的第二输入操作数数据字的可变位置的一半的位部分并将这两者组合成要存储在寄存器 R_d 内的输出数据字的相应头一半和后半。指令 PKHBT 取 R_n 的输入操作数数据字的后半以及 15 R_m 的第二输入操作数数据字的可变位置的一半字长度的部分并将这两相组合成 R_d 的输出数据字的后半和前半。可以看出, R_n 的输入操作数数据字的选定部分在两种情况下在输出数据字 R_d 内在位置上都未位移。这就能通过简单的掩蔽或选择电路来提供这一部分, 从而体现出非常小的额外硬件开销。在 R_m 的字业已右移 k 个位位置之后, 从 20 该字的位位置 15 至 0 中选择图 3 的指令的可变位置半个字的部分。与此相似, 在 R_m 的字业已左移 k 个位位置之后, 从该字的位位置 31 至 16 中选择根据图 4 的指令选择的 R_m 的半个字长度的可变位置部分。

以与图 3 和图 4 的指令的打包功能相结合方式提供的可变位移对调节固定点计算值的“Q”值的变化特别有用。上述变化会在处理这些 25 固定点计算值过程中出现。

图 5 说明了特别适用于执行图 3 和图 4 指令的数据路径 14。寄存器库 4 再次提供在本例中是 32 位数据字的输入操作数数据字并将输出数据字存储起来。上述数据路径包括位移电路 6、加法器电路 8 以及选择和组合电路 22。

30 在操作中, R_n 的未位移的输入操作数数据字直接从寄存器库 4 中传至选择和组合逻辑电路 22。就图 3 的指令而言, 选出 R_n 的值的最高有效 16 位并形成输出数据字 R_d 中的相应位。就图 4 的指令而言, 选出

Rn 的输入操作数数据字的最低有效 16 位并对其进行传递以形成输出数据字 Rd 的最低有效位。Rm 的输入操作数数据字经过全位移电路 6。就图 3 的指令而言，进行 k 个位位置的算术右移，然后，通过选择和组合电路 22 选择出来自位移电路 6 的输出的最低有效 16 位，以形成
5 Rd 的输出数据字的最低有效 16 位。就图 4 的指令而言，位移电路 6 提供了 k 个位位置的左向逻辑位移并将结果提供给选择和组合电路 22。选择和组合电路 22 选择位移电路 6 的输出的最高有效 16 位并用这些位形成 Rd 的输出数据字的最高有效 16 位。

可以看出，选择和组合电路 22 设置在与加法器电路 8 相并行的位
10 置。因此，假定数据路径 14 被仔细地设计成能在一个处理周期内执行全位移和加法操作，则可在不施加任何处理周期限制的情况下，在通常允许进行加法器电路 8 的操作的时间周期内进行较为直接的选择和组合操作。

应该认识到，根据所获得的结果限定了上述的数据处理指令。应
15 该注意，可用多种不同的处理步骤和步骤顺序获得相同的结果值。本发明包括了所有这些能用单个指令产生同样最终结果值的所有这些变化形式。

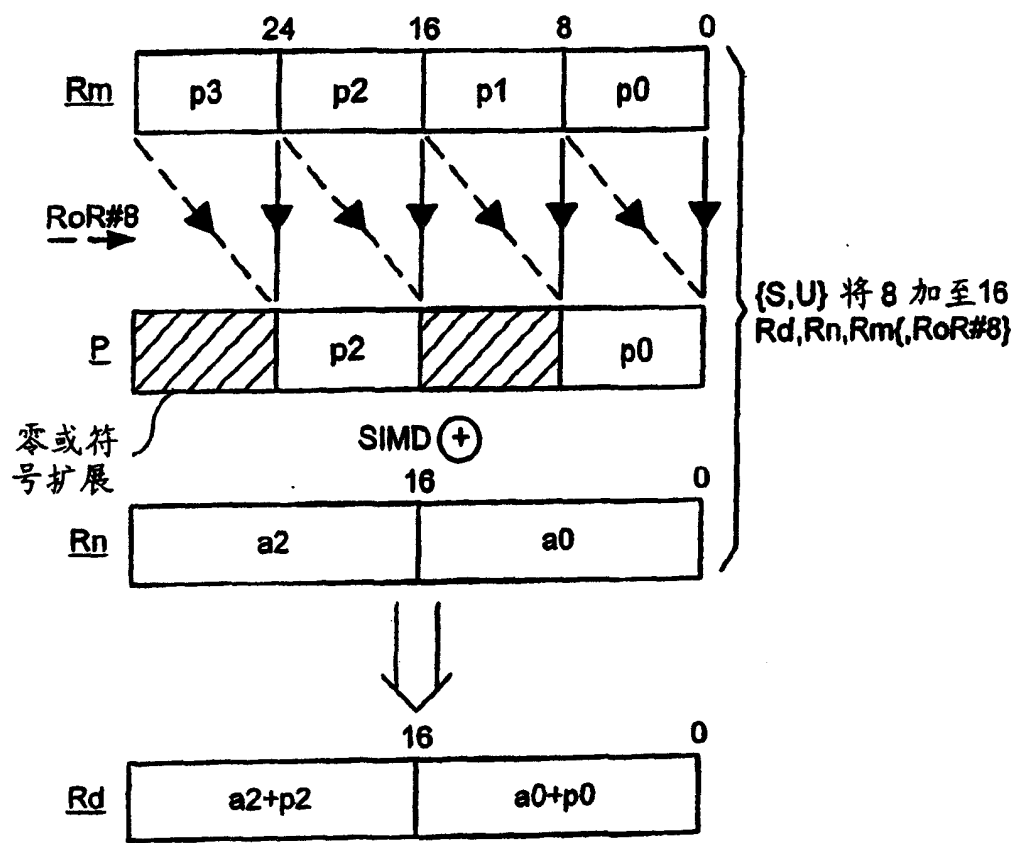


图 1

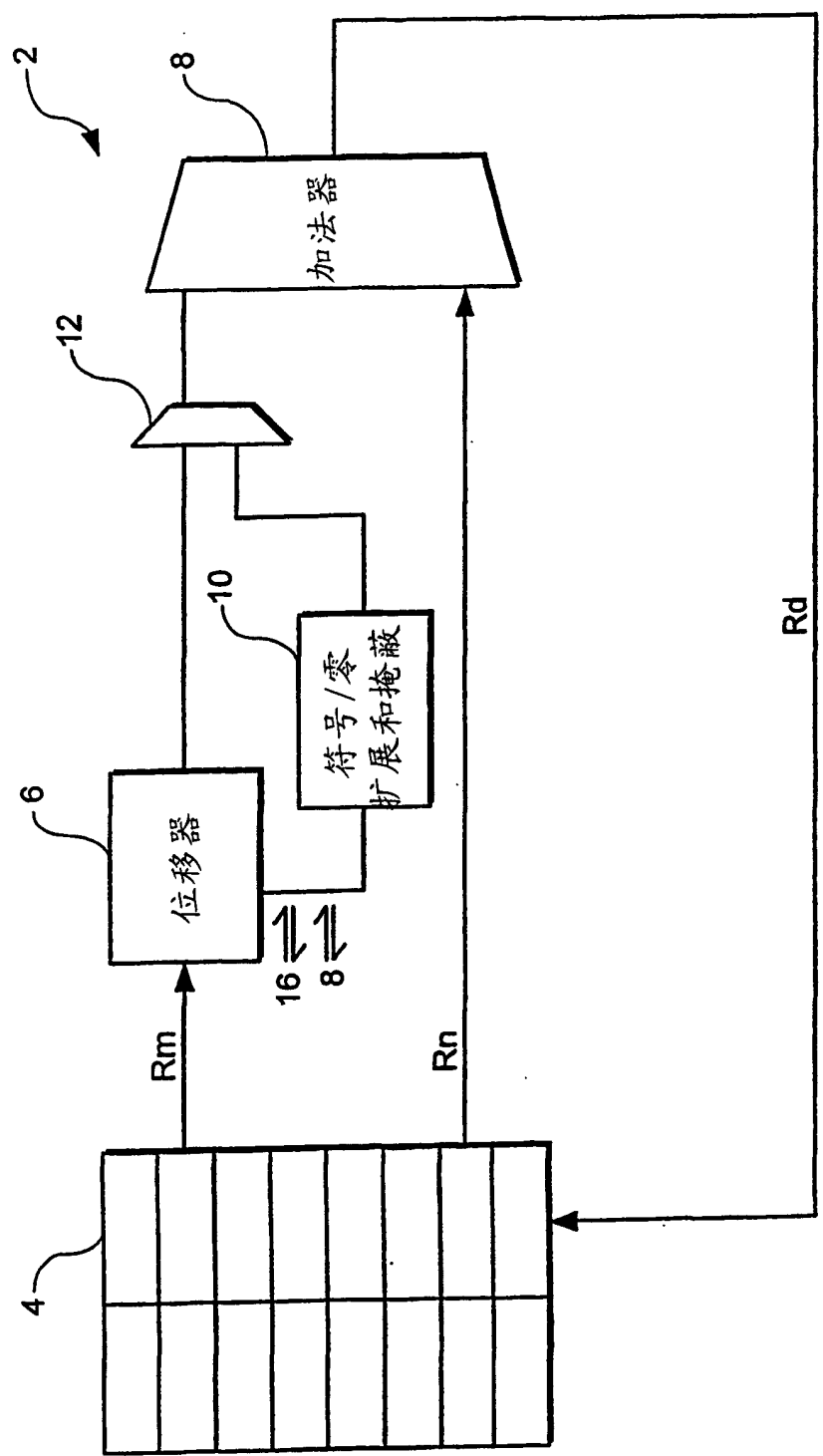


图 2

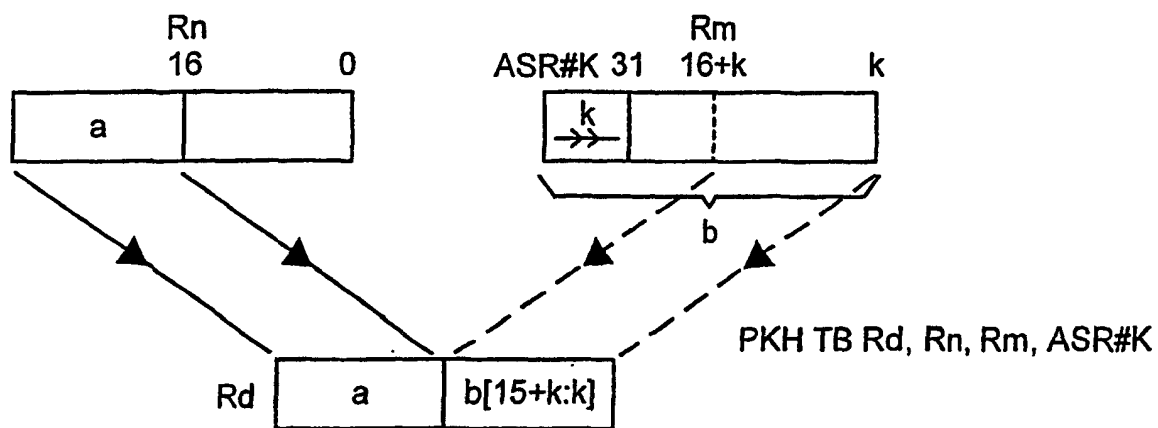


图 3

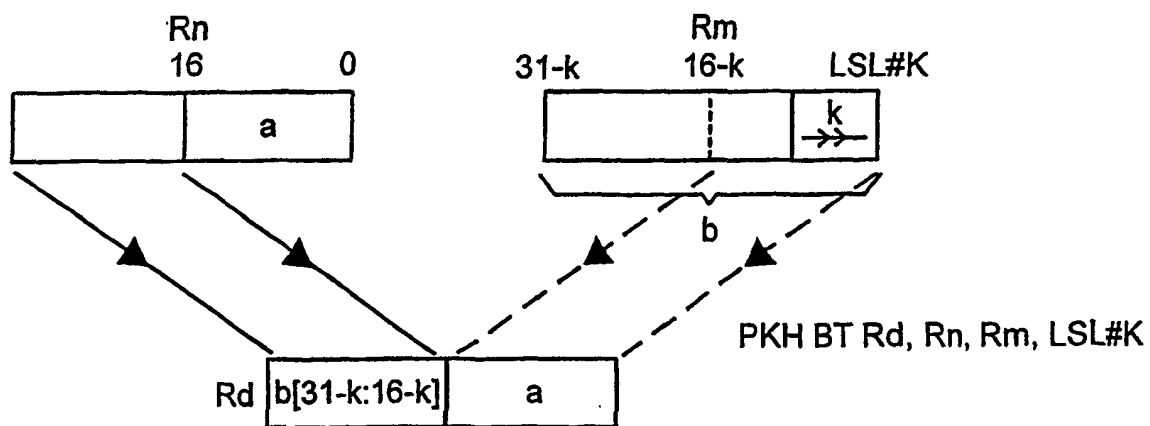


图 4

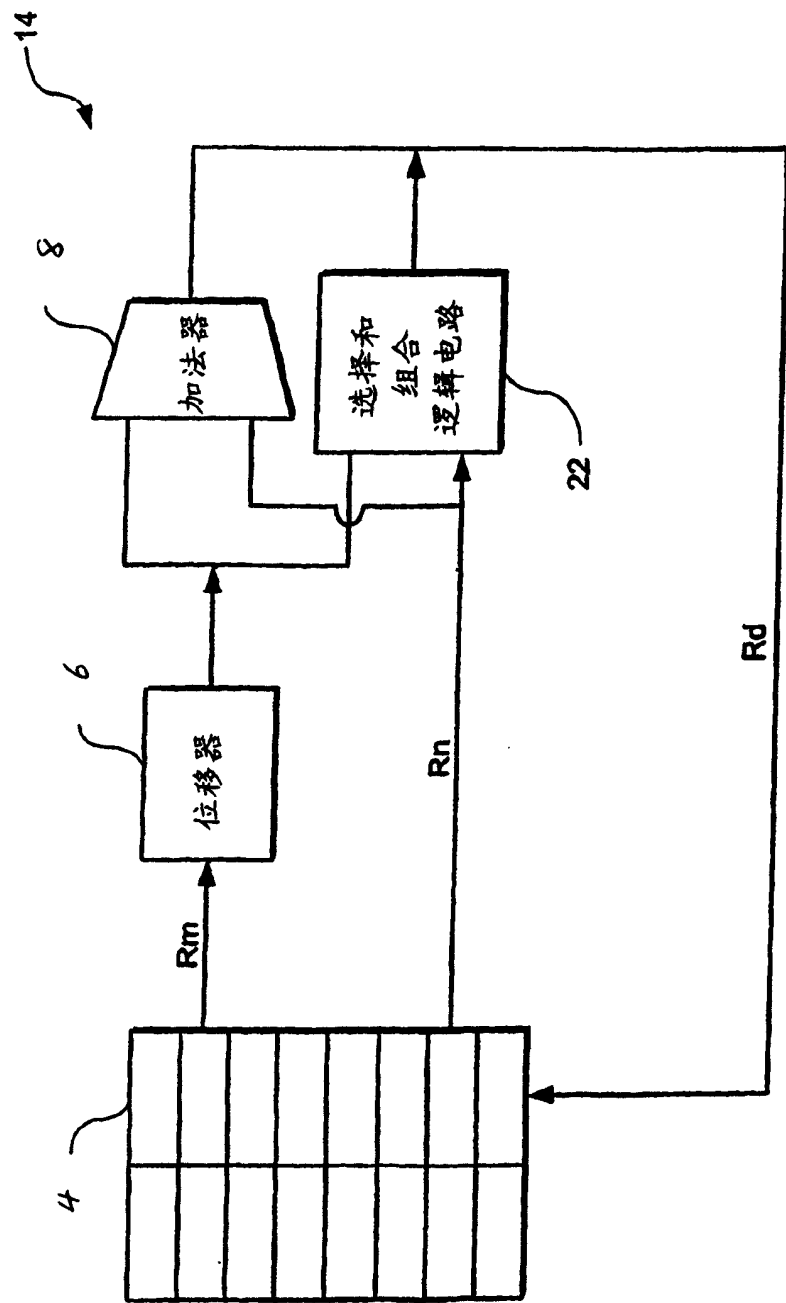


图 5