

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.  
G06F 9/38 (2006.01)



## [12] 发明专利说明书

专利号 ZL 200580010660.7

[45] 授权公告日 2009 年 3 月 25 日

[11] 授权公告号 CN 100472433C

[22] 申请日 2005.3.22

审查员 段雪莲

[21] 申请号 200580010660.7

[74] 专利代理机构 中国专利代理(香港)有限公司

[30] 优先权

代理人 王小衡 王忠忠

[32] 2004.3.31 [33] GB [31] 10/813,628

[86] 国际申请 PCT/GB2005/001059 2005.3.22

[87] 国际公布 WO2005/096140 英 2005.10.13

[85] 进入国家阶段日期 2006.9.30

[73] 专利权人 艾色拉公司

地址 英国布里斯托尔

[72] 发明人 S·诺勒斯

权利要求书 3 页 说明书 10 页 附图 3 页

[56] 参考文献

US5600810A 1997.2.4

CN1467622A 2004.1.14

US2003154358A1 2003.8.14

CN1072788A 1993.6.2

时，所述控制指令被提供给第一处理通道以按照程序顺序执行。

[54] 发明名称

用于在双通道处理器中控制处理的装置和方法

[57] 摘要

根据本发明的实施例，公开了一计算机处理器结构；尤其是公开了一种计算机处理器、一种操作所述处理器的方法以及一种计算机程序产品，所述计算机程序产品利用用于所述计算机的指令集合。在根据本发明的一个实施例中，提供了一种计算机处理器，包括：解码单元，用于解码来自于存储器的一串指令包，其中每个指令包包括多条指令；第一处理通道，其包括多个功能单元并可操作用于执行控制处理操作；第二处理通道，其包括多个功能单元并可操作用于执行数据处理操作；其中所述解码单元可操作用于接收指令包，并检测所述指令包是否定义了(i)多个控制指令或(ii)其中的一个或多个指令为数据处理指令的多个指令，以及其中当所述解码单元检测到所述指令包包括多个控制指令

1. 一种计算机处理器，所述处理器包括：

- (a) 解码单元，用于解码来自于一存储器的一串指令包，其中每个指令包包括多条指令；
- (b) 第一处理通道，其包括多个功能单元并可操作用于执行控制处理操作；
- (c) 第二处理通道，其包括多个功能单元并可操作用于执行数据处理操作；其中所述解码单元可操作用于接收指令包，并检测所述指令包是否定义了(i) 多个控制指令或(ii) 其一个或多个指令为数据处理指令的多个指令，以及其中当所述解码单元检测到所述指令包包括多个控制指令时，所述控制指令被提供给所述第一处理通道以按照程序的顺序执行。

2. 根据权利要求1的计算机处理器，其中所述解码单元可操作用于检测包含三个控制指令的指令包，并且可操作用于控制以所述指令在所述指令包中出现的顺序来执行所述三个控制指令中的每一个的控制过程。

3. 根据权利要求1或2的计算机处理器，其中所述解码单元可操作用于检测包含多个长度相等的控制指令的指令包。

4. 根据权利要求3的计算机处理器，其中所述解码单元可操作用于检测在指令包中的位长在18位到24位之间的控制指令。

5. 根据权利要求4的计算机处理器，其中所述解码单元可操作用于检测在指令包中的每个指令均具有21位位长的多个控制指令。

6. 根据权利要求1的计算机处理器，其中所述解码单元可操作用于接收并解码位长为64位的指令包。

7. 根据权利要求1的计算机处理器，其中所述解码单元可操作用于检测何时在所述指令包中存在至少一数据处理指令，并且作为响应，导致相关的数据被提供给所述数据处理通道。

8. 根据权利要求1的计算机处理器，其中所述解码单元可操作用于检测所述指令包包括至少一数据处理指令以及从一个或多个存储器存取指令、控制指令以及数据处理指令中选出的进一步的指令。

9. 根据权利要求8的计算机处理器，其中至少一数据处理指令和所述进一步的指令被同时执行。

10. 根据权利要求 1 的计算机处理器，其中所述第二处理通道专门用于数据处理操作的执行，并且数据处理指令被以汇编语言的形式提供。

11. 根据权利要求 1 的计算机处理器，其中所述控制处理操作基于直到第一预定义位宽的操作数执行，而所述数据处理操作基于直到第二预定义位宽的数据执行，所述第二预定义位宽大于所述第一预定义位宽。

12. 根据权利要求 1 的计算机处理器，其中所述第一处理通道包括从一个或多个控制寄存器文件，控制执行单元，分支执行单元和装载/存储单元中选出的单元。

13. 根据权利要求 1 的计算机处理器，其中所述第二处理通道包括数据执行路径，所述路径包括可配置的数据执行单元。

14. 根据权利要求 1 的计算机处理器，其中所述第二处理通道包括数据执行路径，所述路径包括固定的数据执行单元。

15. 根据权利要求 13 的计算机处理器，其中在使用中，所述可配置的数据执行单元根据单指令多数据的原则进行操作。

16. 根据权利要求 14 的计算机处理器，其中在使用中，所述固定的数据执行单元根据单指令多数据的原则进行操作。

17. 根据权利要求 1 的计算机处理器，其中所述数据处理通道包括一个或多个数据寄存器文件和装载/存储单元。

18. 根据权利要求 1 的计算机处理器，其中单个的装载/存储单元被所述控制处理通道和所述数据处理通道通过各自的端口进行存取。

19. 根据权利要求 1 的计算机处理器，其中所述解码单元可操作用于检测包含至少一数据处理指令的指令包，其中所述至少一数据处理指令的位长在 30 至 38 位之间。

20. 根据权利要求 19 的计算机处理器，其中所述解码单元可操作用于检测包含至少一数据处理指令的指令包，其中所述至少一数据处理指令的位长为 34 位。

21. 根据权利要求 1 的计算机处理器，其中所述解码单元可操作用于检测包含数据处理操作和存储器存取指令的指令包。

22. 根据权利要求 21 的计算机处理器，其中所述存储器存取指令的位长为 28 位。

23. 根据权利要求 1 的计算机处理器，其中所述解码单元可操作用于检测包含数据处理指令和控制处理指令的指令包。

24. 根据权利要求 1 的计算机处理器，其中所述解码单元可操作用于检测形式为 C 代码或其变型的控制处理指令。

25. 根据权利要求 1 的计算机处理器，其中所述解码单元可操作用于检测汇编语言形式的数据处理指令。

26. 一种操作计算机处理器的方法，所述计算机处理器包括第一和第二处理通道，每一个处理通道具有多个功能单元，其中所述第一处理通道能够执行控制处理操作而所述第二处理通道能够执行数据处理操作，所述方法包括：

(a) 从存储器中接收一系列指令包，每一个所述指令包包括多个定义操作的指令；

(b) 依次解码每一个指令包，通过确定所述指令包是否定义了：

(i) 多个控制指令；或

(ii) 至少一个数据处理指令；以及

其中当所述解码单元检测到所述指令包包括多个控制指令时，提供所述多个控制指令给所述第一处理通道以按照序列执行。

---

## 用于在双通道处理器中控制处理的装置和方法

### 技术领域

本发明涉及计算机处理器、操作所述处理器的方法以及包含用于所述计算机的指令集合的计算机程序产品。

### 背景技术

为了提高计算机处理器的速度，现有技术中的结构已经使用了双执行通道来执行指令。双执行通道处理器能依据单指令多数据（SIMD）的原则进行操作，利用并行操作来提高处理器的速度。

然而，尽管使用了双执行通道和 SIMD 处理，现在仍存在提高处理器速度的需求。典型的双执行通道处理器使用两个基本上相同的通道，所以每个通道都处理控制代码和数据路径（data path）代码。虽然已知的处理器支持 32 位标准编码和 16 位“紧凑”编码的组合，但这样的方案存在几个缺点，包括在 16 位格式中可用的少数几位信息中缺乏语义内容。

此外，常规的通用数字信号处理器不能匹配在很多场合下的应用程序专用算法，其中包括执行例如卷积、快速傅立叶变换、Trellis/Viterbi 编码、相关和有限脉冲响应滤波的特殊操作以及其它操作。

### 发明内容

在根据本发明的一个实施例中，提供了一计算机处理器。所述计算机处理器包括：(a) 解码单元，用于解码来自于存储器的一串指令包，其中每个指令包包括多条指令；(b) 第一处理通道，其包括多个功能单元并可操作用于执行控制处理操作；以及(c) 第二处理通道，其包括多个功能单元并可操作用于执行数据处理操作；其中所述解码单元可操作用于接收一指令包，并检测所述指令包是否定义了(i) 多个控制指令或(ii) 其中的一个或多个指令为数据处理指令的多个指令，并且其中当所述解码单元检测到所述指令包包括多个控制指令时，所述控制指令被提供给所述第一处理通道以按照程序的顺序执行。

在相关的实施例中，所述计算机处理器的解码单元可操作用于检测包含三

个控制指令的指令包以及用于控制执行所述三个控制指令中的每一个的控制过程，所述执行以所述指令在所述指令包中出现的顺序进行。所述解码单元也可操作用于检测包含多个长度相等的控制指令的指令包，或用于检测在指令包中的位长（bit length）在 18 位到 24 位之间的控制指令，以及特别用于检测每个均具有 21 位位长的多个控制指令。所述解码单元可操作用于接收并解码位长为 64 位的指令包。

在进一步的相关实施例中，所述解码单元可操作用于检测何时在所述指令包中存在至少一数据处理指令，并且作为响应，使相关的数据被提供给所述数据处理通道。所述解码单元也可操作用于检测所述指令包包括至少一数据处理指令以及从一个或多个存储器存取指令、控制指令以及数据处理指令中选出的一个进一步的指令。所述至少一数据处理指令和所述进一步的指令可被同时执行。所述第二处理通道可专门用于数据处理操作的执行，并且数据处理指令可以汇编语言的形式提供。所述控制处理操作可基于直到第一预定义位宽的操作数执行，而所述数据处理操作可基于直到第二预定义位宽的数据执行，所述第二预定义位宽大于所述第一预定义位宽。

在进一步的相关实施例中，所述第一处理通道可包括从一个或多个控制寄存器文件，控制执行单元，分支执行单元和装载/存储单元中选出的单元。所述第二处理通道可包括数据执行路径，所述路径包括可配置的数据执行单元。所述第二处理通道也可包括数据执行路径，所述路径包括固定的数据执行单元。在使用中，一个或多个所述可配置的和固定的数据执行单元可根据单指令多数据的原则进行操作。所述数据处理通道可包括一个或多个数据寄存器文件和装载/存储单元。一单个的装载/存储单元可被所述控制处理通道和所述数据处理通道通过各自的端口进行存取。

在进一步的相关实施例中，所述解码单元可操作用于检测包含至少一数据处理指令的指令包，其中所述至少一数据处理指令的位长在 30 至 38 位之间，例如，所述位长可为 34 位。所述解码单元也可操作用于检测包含数据处理操作和存储器存取指令的指令包。所述存储器存取指令的位长可为，例如 28 位。所述解码单元也可操作用于检测包含数据处理指令和控制处理指令的指令包。所述控制处理指令可以是 C 代码或其变型。所述解码单元也可操作用于检测汇编语言形式的数据处理指令。

在根据本发明的另一个实施例中，提供了一操作计算机处理器的方法，所述计算机处理器包括第一和第二处理通道，每一个处理通道具有多个功能单元，其中所述第一处理通道可操作用于执行控制处理操作而所述第二处理通道可操作用于执行数据处理操作。所述方法包括：(a) 从存储器中接收一系列指令包，每一个所述指令包包括多个定义操作的指令；(b) 通过确定所述指令包是否定义了(i) 多个控制指令或(ii) 至少一个数据处理指令，来依次解码每一个指令包；并且其中当所述解码单元检测到所述指令包包括多个控制指令时，提供所述多个控制指令给所述第一处理通道以按照序列执行。

在根据本发明的另一个实施例中，提供了一计算机程序产品，其中包括用于使计算机按照前述的方法进行操作的程序代码装置。

在根据本发明的进一步的实施例中，提供了计算机程序代码，其中包括用于使计算机按照前述的方法进行操作的一系列指令。

在根据本发明的另一个实施例中，提供了一计算机指令集合，其中包括第一类指令包和第二类指令包，所述第一类指令包中的每一个指令包包括顺序执行的多个控制指令，所述第二类指令包中的每一个指令包包括同时执行的至少一数据处理指令和一进一步的指令，所述进一步的指令从一个或多个存储器存取指令、控制指令和数据处理指令中选择得出。

本发明的其它优点和新颖的特征部分地将在下面的描述中阐明，并且部分地通过下面的说明和附图的检验，对本领域的技术人员变得显而易见，或者可以通过实践本发明而被了解。

#### 附图说明

为了更好地理解本发明，以及说明可以怎样实现上述内容，将通过举例的方式参考附图进行说明，其中：

图1是根据本发明的一个实施例的非对称双执行通道计算机处理器的结构图；

图2显示了根据本发明的一个实施例的用于图1中的处理器的典型的指令类；以及

图3是显示了根据本发明的一个实施例的可配置的深度(deep)执行单元的部件的示意图。

## 具体实施方式

图 1 是根据本发明的一个实施例的非对称双通道计算机处理器的结构图。

图 1 中的处理器在两个不同的硬件执行通道之间分割一单指令流 100 的处理：专门用于处理控制代码的控制执行通道 102 和专门用于处理数据代码的数据执行通道 103。依照控制代码和数据路径代码的不同特征，所述两个执行通道 102、103 的数据宽度、操作符和其它特征也不同。典型地，控制代码可能使用更少、更窄的寄存器，难以并行，典型地（但非排他地）被编写为 C 代码或另一高级语言的格式，以及其代码密度通常比其速度性能更重要。相反地，数据路径代码典型地可能使用宽的寄存器的大文件，高度并行，被编写为汇编语言的格式，以及其性能比其代码密度更重要。在图 1 中的处理器中，所述两个不同的执行通道 102 和 103 专门用于处理所述两种不同类型的代码，每一侧具有由寄存器的宽度和数目所区分的其固有结构的寄存器文件，例如控制寄存器文件 104 和数据寄存器文件 105；所述控制寄存器按照位数计算，宽度较窄（在一个例子中，32 位），而所述数据寄存器宽度较宽（在一个例子中，64 位）。因此所述处理器是非对称的，由于其两个执行通道执行不同的专用功能的事实，其两个执行通道的位宽不同。

在图 1 中的处理器中，所述指令流 100 由一系列指令包组成。提供的每一个指令包被指令解码单元 101 解码，所述指令解码单元 101 将控制指令与数据指令分开，这一点将在下面被进一步地描述。所述控制执行通道 102 使用分支单元 106、执行单元 107 和装载存储单元 108 来对所述指令流进行控制流操作的处理，并管理所述机器的状态寄存器，所述装载存储单元 108 在该实施例中被所述数据执行通道 103 所共享。对于 C、C++ 或 Java 语言的编译器，或另一高级语言编译器而言，只有所述处理器的控制侧需要是可见的。在所述控制侧中，分支单元 106 和执行单元 107 的操作与本领域普通技术人员所知的常规的处理器设计是一致的。

所述数据执行通道 103 在固定的执行单元 109 和可配置的深度执行单元 110 中均使用 SIMD（单指令多数据）并行操作。如下面将进一步描述的，所述可配置的深度执行单元 110 除了常规的 SIMD 处理器所使用的宽度维，还提供处理的深度维（depth dimension），以增加每条指令的工作量。

如果所述解码的指令定义了一控制指令，其被应用于所述机器的控制执行

通道上的适当的功能单元中（例如，分支单元 106、执行单元 107 和装载/存储单元 108）。如果所述解码的指令定义了具有固定的或者可配置的数据处理操作的指令，其被提供给所述数据处理执行通道。在所述指令包的数据指令部分中，指定的位指示出所述指令是否是固定的或者可配置的数据处理指令，并且在可配置的指令的情况下，指定的位进一步定义配置信息。基于解码的数据处理指令的子类型，数据被提供给所述机器的数据处理通道中的固定的或者可配置的执行子通道。

在这里，“可配置的”表示从多个预定义的（“伪静态的”）操作符配置中选择一操作符配置的能力。操作符的伪静态配置能使操作符（i）执行特定类型的操作，或（ii）以特定的方式与相关联的元素互连，或（iii）上面所述的（i）或（ii）的组合。在实践中，一选定的伪静态配置可同时确定许多操作符元素的行为和互连。也能控制与所述数据通道相关的切换配置。在一更佳的实施例中，至少所述多个伪静态操作符配置中的某些配置由数据处理指令的操作码部分进行选择，这一点将在下面被进一步地描述。依照这里的实施例，“可配置的指令”允许在多位值的级别执行定制的操作，例如在 4 位或更多位的多位值的级别或在字的级别。

应该指出，在所述机器的各自不同端执行的控制和数据处理指令能定义存储器存取（装载/存储）和基本的算术操作。控制操作的输入/操作数可被提供给/提供自控制寄存器文件 104，然而，数据处理操作的数据/操作数可被提供给/提供自寄存器文件 105。

依照本发明的实施例，每个数据处理操作的至少一个输入可以是向量。在这方面，所述可配置操作符和/或可配置的数据包的切换电路能被认为是可配置用于借助所执行的操作和/或其中的互连的特性执行向量操作。例如，数据处理操作的一个 64 位向量输入可包括 4 个 16 位标量操作数。其中，“向量”是标量操作数的组合。向量算术可在多个标量操作数上执行，并可包括标量元素的操纵、移动和置换。不是向量操作的所有操作数都需要是向量，例如向量操作可以同时具有标量和至少一个向量作为输入，并且输出一个不是标量就是向量的结果。

在这里，“控制指令”包括专门用于程序流、分支和地址生成的指令，但不包括用于数据处理的指令。“数据处理指令”包括用于逻辑操作或算术操作

的指令，所述算术操作的至少一个输入是向量。数据处理指令可对多个数据指令进行操作，例如在 SIMD 处理中，或在处理更宽的、短的数据元素的向量过程中。刚刚提到的控制指令和数据指令的基本功能并不重叠，但是一个共同的性质是两种类型的代码都具有逻辑和标量算术的能力。

图 2 显示了用于图 1 中的处理器的三种类型的指令包。每种类型的指令包为 64 位长。指令包 211 是用于紧凑的控制代码的 3 标量类型，并包括三个 21 位的控制指令 (c21)。指令包 212 和 213 是用于并行执行数据路径代码的 LIW (长指令字) 类型。在该实施例中每个指令包 212 和 213 包括两个指令，但如果需要可以包括不同数目的指令。指令包 212 包括一个 34 位的数据指令 (d34) 和一个 28 位的存储器指令 (m28)；并且其被用于和数据端的装载-存储操作 (所述 m28 指令) 一起并行执行数据端的算术操作 (所述 d34 指令)。使用来自于所述控制端的地址，存储器类的指令 (m28) 能被读取自，或被写到，所述处理器的或者控制端或者数据端。指令包 213 包括一个 34 位的数据指令 (d34) 和一个 21 位的控制指令 (c21)；并且其被用于并行执行数据端的算术操作 (所述 d34 指令) 和控制端的操作 (所述 c21 指令)，例如控制端的算术、分支或装载-存储操作。

图 1 中实施例的指令解码单元 101 使用每个指令包的初始标识位、或在预定义的比特位置的一些其它的指定标识位来确定哪一种类型的包正在被解码。例如如图 2 中所示，初始位 “1” 指明指令包是具有三个控制指令的标量控制指令类型；然而初始位 “01” 和 “00” 指明了指令包的类型 212 和 213，其中包 212 中有数据和存储器指令，或者在包 213 中有数据和控制指令。解码每个指令包的初始位后，图 1 中的解码单元 101 根据指令包的类型，将每个包中的指令适当地传送到所述控制执行通道 102 或所述数据执行通道 103。

为了执行图 2 中的指令包，所述图 1 的实施例中的处理器的指令解码单元 101 从存储器顺序地取出程序包，并且所述程序包被顺序执行。在一指令包中，包 211 的指令被顺序执行，其中在所述 64 位字中最低端处的 21 位控制指令被首先执行，然后是下一个 21 位的控制指令，接着是在最高端处的 21 位控制指令。在指令包 212 和 213 中，所述指令被同时执行（尽管在根据本发明的实施例中，这不是必需的）。因而，按照图 1 的实施例中的处理器的程序顺序，所述程序包被顺序执行，但在一个包中的指令可以被顺序执行，如包类型 211，

也可以同时执行，如包类型 212 和 213。下面，212 和 213 类型的指令包分别简写为 MD 和 CD 包（分别为包含一个存储器指令和一个数据指令，以及一个控制指令和一个数据指令）。

使用 21 位的控制指令，图 1 中的实施例克服了在具有其它长度的指令的处理器中发现的许多缺点，所述处理器特别是指支持用于数据指令的 32 位标准编码和用于控制代码的 16 位“紧凑”编码的处理器。在这样的双 16/32 位处理器中，存在由于对每一条指令使用双编码，或者由于使用两个分离的解码器而产生冗余，其中所述的两个分离的解码器具有在分支编码模式和取地址之间进行切换的装置，或其它装置。根据本发明的实施例，该冗余可通过对所有控制指令使用一个单一的 21 位长度来消除。进一步地，21 位控制指令的使用消除了由于在 16 位“紧凑”编码模式中语义内容不足而引起的缺点。由于语义内容不足，使用 16 位模式的处理器典型地需要混合一些设计的折衷，例如：使用两操作数的破坏操作，导致相应的副本的代码膨胀；使用对寄存器文件子集的窗口式访问，导致代码膨胀溢出/装满或窗口指针操作；或频繁地恢复为 32 位格式，因为不是所有操作都能被表示在 16 位格式中的非常少的可用操作代码位中。在根据本发明的实施例中，这些缺点通过使用 21 位的控制指令被缓解了。

根据本发明的实施例，可以使用各种各样的指令。例如，指令的识别标志可以是下面中的任何一种，其中 C 格式、M 格式和 D 格式分别表示控制格式、存储器存取格式和数据格式：

指令的识别标志	参数	使用在
<i>Instr</i>	指令无参数 (argument)	只在 C 格式中
<i>instr dst</i>	指令有一单个的目的地参数	只在 C 格式中
<i>iInstr src0</i>	指令有一单个的源参数	只在 C 格式或 D 格式中
<i>instr dst, src0</i>	指令有一单个的目的地参数和一单个的源参数	D 格式和 M 格式指令
<i>instr dst, src0, src1</i>	指令有一单个的目的地参数和两个源参数	C 格式、D 格式和 M 格式指令

同样根据本发明的实施例，C 格式指令都提供 SISD（单指令单数据）操作，然而 M 格式和 D 格式指令提供 SISD 或 SIMD 操作。例如，控制指令可

提供通用算术、比较和逻辑指令，控制流指令，存储器装载和存储指令，以及其它指令。数据指令可提供常规的算术、移位、逻辑和比较指令，混洗、排序、字节扩展和排列指令，线性反馈移位寄存器指令，以及通过可配置的深度执行单元 110（下面进一步描述）的用户定义的指令。存储器指令可以提供存储器装载和存储，复制选定的数据寄存器到控制寄存器，复制广播控制寄存器到数据寄存器，以及紧接着寄存器指令。

根据本发明的一个实施例，图 1 的处理器的特征在于第一固定的数据执行通道和第二可配置的数据执行通道。所述第一数据通道具有固定的 SIMD 执行单元，所述单元被分割为具有与常规的 SIMD 处理设计相似的样式的通道。所述第二数据通道具有可配置的深度执行单元 110。“深度执行”涉及处理器在返回结果给寄存器文件之前对由一个发出的指令提供的数据执行多个连续操作的能力。深度执行的一个例子可在常规的 MAC 操作（乘法和累加）中找到，所述 MAC 操作对来自于一个指令的数据进行两个操作（乘法和加法），因此具有两级的深度。深度执行的特征在于操作数输入的数目等于结果输出的数目，或者同义地，入价（valency-in）等于出价（valency-out）。这样，例如常规的只有一个结果的两操作数加法不是深度执行的例子，因为操作数的数目不等于结果的数目，但是卷积、快速傅立叶变换、Trellis/Viterbi 编码、相关、有限脉冲响应滤波和其它信号处理算法是深度执行的例子。专用数字信号处理（DSP）算法典型地在比特级和存储器映射方式中执行深度执行。然而，常规的寄存器映射通用 DSP 不执行深度执行，改为在 MAC 操作中在至多两次的深度执行指令。作为对比，图 1 的处理器提供能够在大于两次的深度来深度执行动态配置的字级指令的寄存器映射通用处理器。在图 1 的处理器中，所述深度执行指令（将被执行的数学函数的图表）的性质能被指令本身中的配置信息调整/定制。在更佳的实施例中，格式指令包含分配给配置信息的比特位置。为了提供该能力，所述深度执行单元 110 具有可配置的执行资源，其意味着操作符模式、互连和常量能被加载以匹配每个应用程序。深度执行为并行执行增加了一个深度维，所述深度维与早期的 SIMD 和 LIW 处理的概念所提供的宽度维正交，因此这表示了用于增加通用处理器的每条指令的工作量的一个附加维。

图 3 显示了根据本发明实施例的一个可配置的深度执行单元 310 的组件。如图 1 中所示，所述可配置的深度执行单元 110 是数据执行通道 103 的一部分，

并且因此可以被来自于图 2 中的 MD 和 CD 指令包 212 和 213 中的数据端指令所指示。在图 3 中，指令 314 和操作数 315 被从图 1 中的指令解码单元 101 和数据寄存器文件 105 中提供给所述深度执行单元 310。在指令 314 中的多位配置代码被用于访问控制映射 316，所述控制映射 316 扩展所述多位代码到用于配置所述深度执行单元的操作符的一组相对复杂的配置信号中。所述控制映射 316 可以例如被具体表现为查找表，在所述查找表中所述指令的不同的可能存在的多位代码被映射到所述深度执行单元的不同的可能存在的操作符配置中。基于查阅所述控制映射 316 的查找表的结果，交叉开关互连 317 配置了一组操作符 318-321，在所有的方案中，都必须执行由所述多位指令代码所指示的操作符配置。所述操作符可包括，例如乘法操作符 318、算术逻辑部件 (ALU) 操作符 319、状态操作符 320 或交叉一通道序列改变器 (cross-lane permuter) 321。在一个实施例中，所述深度执行单元包含 15 个操作符：一个乘法操作符 318、8 个 ALU 操作符 319、4 个状态操作符 320 和两个交叉一通道序列改变器 321；虽然其它数目的操作符也是可能的。提供给所述深度执行单元的操作数 315 可以是，例如两个 16 位的操作数，这些操作数被提供给第二交叉开关互连 322，所述第二交叉开关互连 322 可提供所述操作数给适当的操作符 318-321。所述第二交叉开关互连 322 也从所述操作符 318-321 接收中间结果的反馈 324，所述反馈 324 也可接着由所述第二交叉开关互连 322 依次提供给适当的操作符 318-321。第三交叉开关互连 323 多路复用来自于所述操作符 318-321 的结果，并输出一最终结果 325。多种控制信号能被用于配置所述操作符，例如，图 3 的实施例中的控制映射 316 不是必须实现为单个查找表的形式，而是可以实现为一系列两个或多个串联的查找表。第一查找表中的条目可以从给定的多位指令代码指向第二查找表，从而减少用于复杂的操作符配置中的每个查找表所需的存储量。例如，第一查找表可以被组织为配置类别的库，因此多个多位指令代码被一起分组在所述第一查找表中，其中每一组指向一随后的查找表，所述随后的查找表为该组中每一个多位代码提供专用配置。

根据图 3 的实施例，所述操作符被有利地预先设置进不同的操作符类中。实际上，这可通过战略层次的硬连线而实现。这种方法的一个优点是其意味着更少的预定义配置需要被存储，并且所述控制电路能更简单。例如，操作符 318 被预配置在乘法操作符类中，操作符 319 被预配置为 ALU 操作符，操作符 320

---

被预配置为状态操作符，以及操作符 321 被预配置为交叉一通道序列改变器，而其它预配置操作符类也是可能的。然而，即使预先配置了操作符的类，仍存在运行时灵活性使指令为了实现一给定算法的专用配置的最终布置，能够至少布置：(i) 每一类中操作符的连接性；(ii) 与其它类中的操作符的连接性；(iii) 任一相关切换装置间的连接性。

有技术的读者将认识到，虽然前述内容描述了被认为是最佳方式的内容以及执行本发明的其它适当的方式，本发明不应当被限制为在所述较佳的实施例的说明中公开的特定装置配置或方法步骤。本领域的技术人员也将认识到本发明具有宽阔的应用范围，并且所述实施例容许不偏离于本发明概念的不同实现和修改的极大的变化范围。特别地，其中提到的示例的位宽不是用来限制，涉及如半字、字、长整型等的位宽的任意选择也不是用来限制。

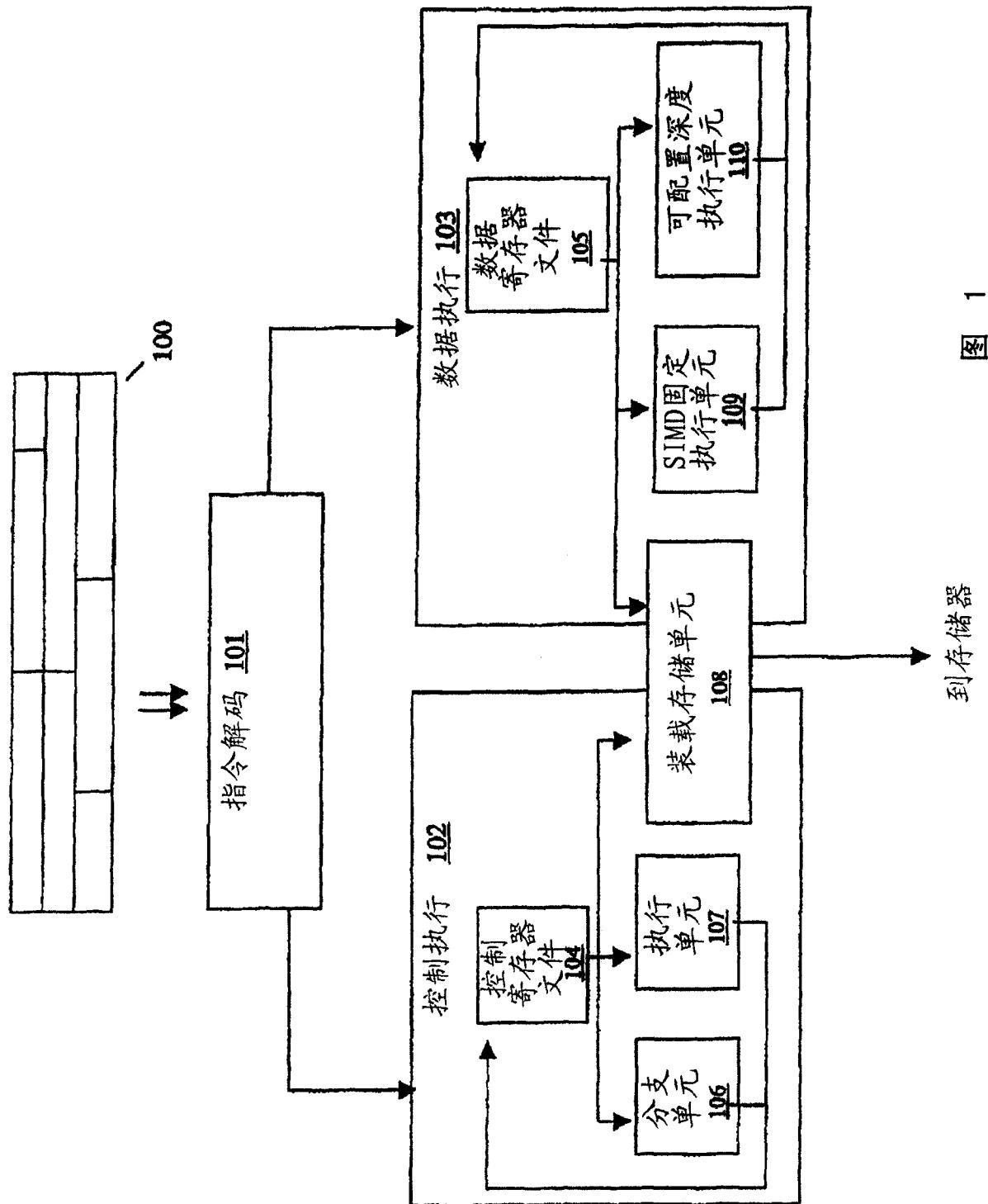


图 1

1	c21(第三次发布)	c21(第二次发布)	c21(第一次发布)
0	1	d34	m28
0	0	d34	保留
0	0		c21

图 2

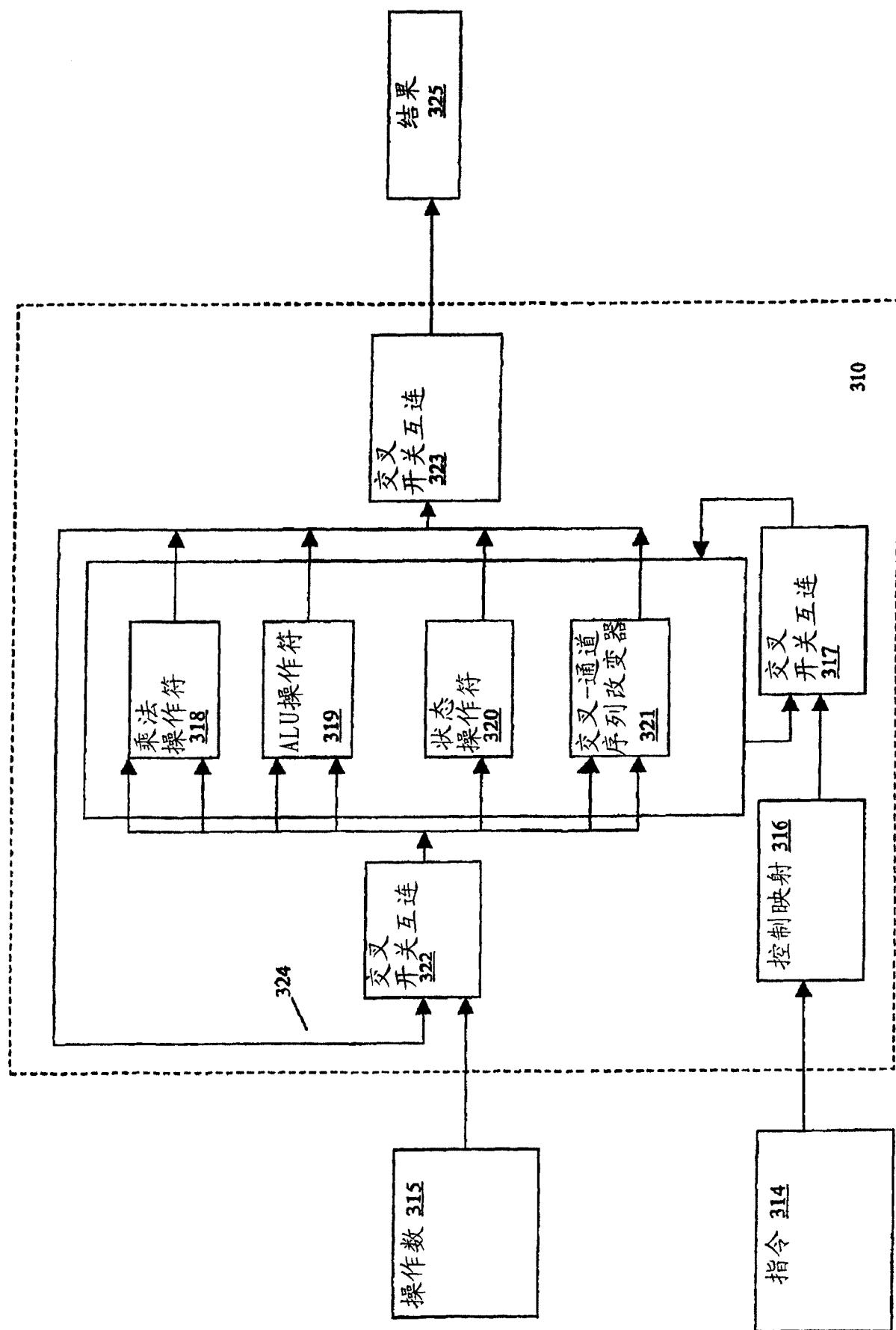


图 3