



(12) 发明专利

(10) 授权公告号 CN 108664596 B

(45) 授权公告日 2021.07.27

(21) 申请号 201810434459.0

G06F 16/2455 (2019.01)

(22) 申请日 2013.06.14

G06F 21/62 (2013.01)

(65) 同一申请的已公布的文献号

G06F 12/0817 (2016.01)

申请公布号 CN 108664596 A

G06F 12/0804 (2016.01)

(43) 申请公布日 2018.10.16

G06F 15/173 (2006.01)

(30) 优先权数据

G06F 16/245 (2019.01)

61/709,142 2012.10.02 US

G06F 9/38 (2006.01)

13/778,009 2013.02.26 US

G06F 16/22 (2019.01)

(62) 分案原申请数据

G06F 16/2453 (2019.01)

201380049301.7 2013.06.14

G06F 16/27 (2019.01)

(73) 专利权人 甲骨文国际公司

G06F 12/14 (2006.01)

地址 美国加利福尼亚

G06F 16/28 (2019.01)

(72) 发明人 K·艾加仑 G·F·斯沃特

G06F 13/40 (2006.01)

S·卡皮尔

G06F 9/30 (2006.01)

(74) 专利代理机构 中国贸促会专利商标事务所

G06F 9/54 (2006.01)

有限公司 11038

G06F 13/10 (2006.01)

代理人 张鑫

G06F 15/78 (2006.01)

(51) Int.Cl.

(56) 对比文件

G06F 12/12 (2016.01)

CN 102014116 A, 2011.04.13

G06F 13/362 (2006.01)

US 2011196833 A1, 2011.08.11

审查员 王艳臣

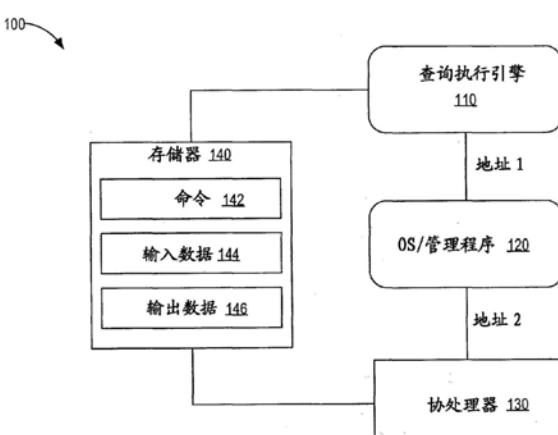
权利要求书3页 说明书14页 附图5页

(54) 发明名称

用于表扫描加速的硬件

(57) 摘要

本公开涉及用于表扫描加速的硬件。提供了用于处理查询的技术。为处理查询所需的一个或多个操作由独立于运行查询处理软件的通用微处理器的协处理器执行。查询处理软件接收查询，确定完全处理查询所需要执行的一个或多个操作，并且发出一个或多个命令给一个或多个协处理器，所述一个或多个协处理器被编程用于执行所述操作之一，比如表扫描操作和/或查找操作。所述查询处理软件从所述协处理器中获得结果，并且在其上执行一个或多个附加操作，以生成所述查询的最终结果。



1. 一种处理芯片,包括:

微处理器,被配置为:

接收包括扫描操作和一个或多个其它操作的查询;

响应于接收到所述查询,生成扫描操作命令,其中所述扫描操作命令包含用于识别输入数据的地址数据以及用于与所述输入数据进行比较的一个或多个值来作为所述扫描操作命令的参数;

协处理器,被配置为:

读取所述扫描操作命令;

促使从由所述地址数据指示的位置处读取所述输入数据;

在所述输入数据与所述一个或多个值之间执行比较;

基于所述比较来生成结果数据;

促使将所述结果数据通知微处理器,

其中,微处理器被进一步配置为基于所述结果数据执行所述一个或多个其它操作。

2. 根据权利要求1所述的处理芯片,其中,微处理器是通用微处理器,所述通用微处理器处理基于以高级编程语言编写的代码的指令。

3. 根据权利要求1所述的处理芯片,其中:

所述扫描操作命令还包含指示要执行的操作的类型的数据;

协处理器还被配置为识别所述操作类型数据以确定操作的类型。

4. 根据权利要求3所述的处理芯片,其中,协处理器还被配置为执行以下步骤:

使用所述操作类型数据来确定用来解释所述输入数据的逻辑;

将所述输入数据从第一数据类型转换为与第一数据类型不同的第二数据类型。

5. 根据权利要求1所述的处理芯片,其中,所述扫描操作命令还包含指示所述结果数据将要存储于何处的目的地数据。

6. 根据权利要求1所述的处理芯片,其中,促使将所述结果数据通知微处理器包括:促使所述结果数据被存储于微处理器的高速缓存内。

7. 根据权利要求1所述的处理芯片,其中:

所述输入数据包含值数据和计数数据;

在所述值数据中的输入值对应于数据元素,所述数据元素的数量由在所述计数数据中的计数值指示;

在所述输入数据与所述一个或多个值之间执行比较包括:将所述输入值与所述一个或多个值进行比较;

生成所述结果数据包括:生成指示所述输入值是否满足所述一个或多个值的结果指示;

生成所述结果数据是在没有将所述输入值与所述一个或多个值进行由所述计数值指示的次数的比较的情况下执行的。

8. 根据权利要求1所述的处理芯片,其中:

协处理器是串联连接的多个协处理器中的第一协处理器;

促使将所述结果数据通知微处理器包括:促使所述结果数据被发送给所述多个协处理器中的第二协处理器的缓冲区。

9. 根据权利要求1所述的处理芯片,其中:

包含于所述扫描操作命令中的所述地址数据包含一个或多个虚地址;

协处理器还被配置为促使所述一个或多个虚地址被协处理器用来读取所述输入数据的一个或多个物理地址所替代。

10. 根据权利要求1所述的处理芯片,其中:

所述一个或多个值包含多个数据元素;

在所述输入数据与所述一个或多个值之间执行比较包括:在所述输入数据与所述多个数据元素之间同时执行比较。

11. 根据权利要求10所述的处理芯片,其中基于所述比较来生成所述结果数据包括生成位向量,其中在所述位向量中的每个位指示比较结果是真还是假。

12. 一种用于处理查询的方法,包括:

由微处理器接收涉及扫描或查找操作和一个或多个其它操作的查询;

响应于接收到所述查询:

由微处理器向与所述微处理器不同的协处理器发送扫描或查找操作命令,其中,所述扫描或查找操作命令对应于所述扫描或查找操作并且与一个或多个值关联;

由所述协处理器通过以下操作来处理所述扫描或查找操作命令:

在被所述查询引用的输入数据和与所述扫描或查找操作命令关联的所述一个或多个值之间进行比较;

基于所述比较来生成结果数据;

促使所述结果数据被存储;以及

由微处理器基于所述结果数据处理所述一个或多个其它操作。

13. 如权利要求12所述的方法,其中:

所述扫描或查找操作命令包含用于识别待由所述协处理器读取的所述输入数据的地址数据来作为参数;

处理所述扫描或查找操作命令还包括:

从存储器读取所述扫描或查找操作命令;

促使从由所述地址数据指示的位置处读取所述输入数据。

14. 如权利要求13所述的方法,其中:

包含于所述扫描或查找操作命令中的所述地址数据包含一个或多个虚地址;

所述方法还包括促使所述一个或多个虚地址被所述协处理器用来读取所述输入数据的一个或多个物理地址所替代。

15. 如权利要求12所述的方法,其中:

所述扫描或查找操作命令还包含指示要执行的操作的类型的操作类型数据;

进一步通过首先识别所述操作类型数据以确定操作的类型来处理所述扫描或查找操作命令。

16. 如权利要求12所述的方法,其中,所述扫描或查找操作命令还包含指示所述结果数据将要存储于何处的目的地数据。

17. 如权利要求12所述的方法,其中,促使所述结果数据被存储包括:促使所述结果数据被存储于所述微处理器的高速缓存内。

18. 如权利要求12所述的方法,其中,促使所述结果数据被存储还包括:促使指示所述扫描或查找操作命令已被执行的完成状态被存储。

19. 如权利要求12所述的方法,还包括:

从存储器中检索所述结果数据;

在从所述存储器中检索了所述结果数据之后,处理要求将所述结果数据作为所述一个或多个其它操作的输入的所述一个或多个其它操作。

20. 如权利要求12所述的方法,其中:

所述协处理器是串联连接的多个协处理器中的第一协处理器;

促使所述结果数据被存储包括:促使所述结果数据被发送给所述多个协处理器中的第二协处理器的缓冲区;

所述方法还包括:

当所述第一协处理器正在执行所述查询的一部分的同时,由所述第二协处理器从所述缓冲区读取所述结果数据,以及

由所述第二协处理器基于所述结果数据来生成第二结果数据。

21. 如权利要求12所述的方法,还包括:

生成所述扫描或查找操作命令,生成所述扫描或查找操作命令包括生成包含所述扫描或查找操作命令的多个扫描或查找操作命令;

对于所述多个扫描或查找操作命令中的每个扫描或查找操作命令,促使所述每个扫描或查找操作命令被存储于存储器内;

其中,多个协处理器中的每个协处理器选择所述多个扫描或查找操作命令中的不同扫描或查找操作命令。

22. 如权利要求12所述的方法,还包括确定是否满足一个或多个准则,其中向所述协处理器发送所述扫描或查找操作命令仅在满足所述一个或多个准则的情况下执行。

23. 根据权利要求22所述的方法,其中所述一个或多个准则是基于需要由所述协处理器读入以执行所述扫描或查找操作的数据量的。

24. 根据权利要求12所述的方法,其中:

所述一个或多个值包含多个数据元素;

在所述输入数据与所述一个或多个值之间执行比较包括:将所述输入数据与所述多个数据元素同时进行比较。

25. 根据权利要求12所述的方法,其中基于所述比较来生成所述结果数据包括生成位向量,其中在所述位向量中的每个位指示比较结果是真还是假。

用于表扫描加速的硬件

[0001] 本申请是国际申请日为2013年6月14日、名称为“用于表扫描加速的硬件”、申请号为201380049301.7 (PCT/US2013/045873) 的发明专利申请的分案申请。

[0002] 优先权和相关申请

[0003] 本申请要求在2013年2月26日提交的美国专利申请No.13/778,009的优先权；该申请No.13/778,009要求在2012年10月2日提交的美国临时申请No.61/709,142的优先权，这两个申请的内容通过引用的方式并入本文，如同完全在此阐述一样。

[0004] 本申请涉及在2013年2月26日提交的美国专利申请No.13/778,013，该申请的内容通过引用的方式并入本文，如同完全在此阐述一样。

技术领域

[0005] 本发明一般地涉及处理查询，并且更具体地，涉及在一个或多个协处理器中使用定制硬件来执行为处理该查询所需的一个或多个操作。

背景技术

[0006] 这部分所描述的方法是能够实行的方法，但不一定是之前已经构想出的或已经实现的方法。因此，除非另有指出，否则不应这样假定：这部分所描述的任何方法仅仅由于它们包含于这部分内而算作现有技术。

[0007] 对数据库发出的查询典型地将诸如关系表的一个或多个数据库对象作为目标。通常，存取按关系表组织的数据涉及扫描该关系表或其至少一部分。常见的SQL查询是针对数据库表需要过滤器的SQL查询，比如下面所示：

[0008] `select EMPLOYEE from T_EMPLOYEES where HIRE_YEAR='2012'`

[0009] 在本例中，在该数据库表T_EMPLOYEES中搜索受雇于2012年的所有雇员。该搜索（或“扫描”）通过在一个或多个微处理器上运行的软件来完成，该一个或多个微处理器执行一系列指令以在该表中搜索该指定值，在本例中该指定值为“2012”。第一步骤在对大的数据库运行分析应用时往往是性能瓶颈，因为该步骤必须在整个表上运行，而该表可以是数太字节大小。后续的步骤将在满足扫描中的准则集（在上例中为受雇于2012年的雇员）的第一扫描步骤的过滤的子集上进行。因此，机器每单位时间能够过滤的行数是该机器重要的性能度量。该度量被称为“扫描速度”。

[0010] 用于处理查询（例如，涉及扫描表的查询）的方法依赖于软件技术，其中该软件在通用微处理器上执行（或“运行”）。

附图说明

[0011] 在附图中：

[0012] 图1是示出根据实施例的用于加速表扫描的示例计算机系统的框图；

[0013] 图2是示出根据实施例的示例协处理器的框图；

[0014] 图3是示出根据实施例的用于处理查询的过程的流程图；

[0015] 图4是示出在实施例中的示例查找向量的一部分的框图;以及

[0016] 图5是示出计算机系统的框图,本发明的实施例可以在该计算机系统上实施。

具体实施方式

[0017] 在下面的描述中,为了解释的目的,阐明了众多具体的细节以便提供对本发明的全面理解。但是,应当很明显的是,本发明可以在没有这些特定的细节的情况下实施。在其它情况下,众所周知的结构和设备以框图的形式来示出以便避免不必要地使本发明变得晦涩难懂。

[0018] 总体概况

[0019] 在实施例中,与查询相关的扫描操作或查找操作以比如与通用微处理器位于同一芯片上的协处理器的硬件来实现。这样,扫描操作由定制硬件来执行,而查询的其它部分由通用微处理器运行软件来处理。让不同的硬件组件执行不同的操作的一个优点是:定制硬件能够更好地处理扫描操作或查找操作所需的大量数据。此外,定制硬件使通用微处理器及其相关联的高速缓存空闲出来以从事于查询的其它部分或者甚至是与处理该查询的软件无关的其它任务。

[0020] 本发明的实施例并不限于任何特定的微处理器或图形处理单元(GPU)。

[0021] 下面的实例将表格作为被扫描的数据对象。但是,并非所有实施例都限定于表格。除了表格之外的数据对象也可以被扫描。

[0022] 系统概况

[0023] 图1是示出根据实施例的用于加速查询的处理的示例计算机系统100的框图。计算机系统100包含查询执行引擎110、OS/管理程序(Hypervisor)120、协处理器130和存储器140。尽管图中仅示出单个协处理器130,系统100可以包含多个协处理器。

[0024] 查询执行引擎110和管理程序120是驻留于存储器(例如,DRAM和/或高速缓冲存储器)内且包含由通用微处理器执行的指令的程序。查询执行引擎110包含一个或多个软件组件,并且可以与不是查询执行引擎110的一部分的一个或多个其它软件组件通信,以便执行查询以生成该查询的结果。查询执行引擎110可以被配置用于重写查询(例如,SQL查询)以生成查询执行引擎110能够执行的重写的查询。作为选择,另一个软件组件接收原始查询,基于原始查询来生成重写的查询,并且将重写的查询传递给查询执行引擎110以进行处理。

[0025] 查询执行引擎110可以被配置用于处理的查询类型的非限制性实例包括SQL查询和XML查询,比如XPath查询和XQuery查询。查询执行引擎110被配置用于处理的至少一种查询类型是要求对对象或其一部分进行扫描的查询。如同前面所述的,被扫描的对象的非限制性实例是逻辑上按照一个或多个列和多个行来组织的关系表。虽然数据在逻辑上可以按照单个表来组织,但是数据也可以非常不同地按照诸如硬盘驱动器或闪存设备的永久存储器来组织。例如,表格的数据可以进行分割,或者表格的不同列可以存储在完全不同的存储位置内。

[0026] 管理程序120充当在查询执行引擎110与协处理器130之间的接口。换言之,由查询执行引擎110对协处理器130发出的命令经由管理程序120来发出。因而,查询执行引擎110通过对OS/管理程序120作出API调用来在管理程序接口上发出命令。

[0027] 管理程序是允许多个操作系统(“客户”)在主机计算机上同时运行的硬件虚拟化

技术。管理程序给客户操作系统提供虚拟操作平台，并且管理客户操作系统的执行。各种操作系统的多个实例可以共享虚拟的硬件资源。管理程序可以安装于服务器硬件上，具有运行客户操作系统的功能，其中客户操作系统自身充当服务器。

[0028] 一种类型的管理程序直接在主机的硬件上运行，用于控制硬件并且管理客户操作系统。从而，客户操作系统在管理程序之上的另一个层面上运行。另一种类型的管理程序在典型的操作系统环境内运行。以该类型的管理程序层作为不同的第二软件层面，客户操作系统在硬件之上的第三层面上运行。换言之，第一类型的管理程序直接在硬件上运行，而第二类型的管理程序在例如FreeBSD、Linux或Windows的另一个操作系统上运行。

[0029] 因而，尽管元件120被标记为“OS/管理程序”，但是操作系统和管理程序是不同的实体。为了该描述的目的，OS和管理程序被同等对待。下面对元件120的引用将为“管理程序120”。

[0030] 尽管管理程序120在一种实施例中被示为计算机系统100的一部分，但是计算机系统100不包含管理程序。在该实施例中，查询执行引擎110在没有首先要求由除了计算机系统100的操作系统(未示出)外的任何其它软件组件进行处理的情况下直接向协处理器130发出命令。

[0031] 协处理器130是被编程来执行与由执行查询执行引擎110和管理程序120的通用处理器执行的任务分隔开的一个或多个任务的硬件元件。虽然协处理器130与执行查询执行引擎110的通用处理器分离，但是协处理器130可以被看作是查询执行引擎110的一部分，因为协处理器130执行原来由查询执行引擎110执行的一个或多个任务。

[0032] 在实施例中，协处理器130至少执行将指定的目标值(或值的目标范围)与表中的一系列输入值进行比较的任务。该任务被称为扫描操作并且在下文更详细地描述。在实施例中，协处理器130能够将多个指定的目标值(或者多个指定的目标范围)与表中的一系列输入值或数据元素进行比较。

[0033] 在另一实施例中，协处理器130至少执行确定在特定的值集中是否存在一个或多个值的任务。该一个或多个值可以被用来索引到该特定的值集之内，使得不需要针对该一个或多个值中的每个值来对该特定的值集进行扫描。在该实施例中，协处理器130被编程以高效地执行查找操作，这将在下文更详细地描述。

[0034] 除了由查询执行引擎110确立或指示的初始参数外，协处理器130在没有来自查询执行引擎110或任何其它软件的干预的情况下执行扫描操作和/或查找操作，直到扫描操作或查找操作(由启动命令指出)完成。在那时，协处理器130向查询执行引擎110发出信号：操作的结果可用。该信号可以是设定标记的形式。管理程序120可以使用该信号在协处理器130的命令队列中插入新的命令。

[0035] 在实施例中，协处理器130被编程来处理不同的数据类型/格式和元素尺寸。例如，协处理器130可以处理字符串格式的、日期格式的或数字(例如，整数或浮点数)格式的数据。此外，协处理器130处理的数据元素的尺寸可以是特定数量的位(例如，7位)或者特定数量的字节(例如，2字节)。而且，来自特定源(比如表)的数据元素可以为可变长度或固定长度。在一种实施例中，协处理器130接收自对象(比如表)的数据元素可以是一个尺寸的，并且协处理器130执行操作以在执行例如该数据元素与另一个数据元素的比较之前缩小或增大该数据元素的尺寸，比如从该数据元素中去除一个字节，给该数据元素增添9位或者解压

该数据元素。

[0036] 在实施例中，协处理器130驻留于芯片上，也就是，与执行查询执行引擎110的通用微处理器相同的芯片上。协处理器130包含(a)使表数据(或其它数据)从片上或片外存储器流到协处理器130的存储器接口以及(b)执行扫描操作和/或查找操作的计算块。例如，在扫描操作的情形中，计算块作用于表数据上以确定在表数据中是否出现了指定的值或值范围。因而，一组比较器被用来确定传入流的每个元素是否等于搜索值或者位于所搜索的值范围之内。在该组比较器中的每个比较器可以同时执行比较操作。因而，协处理器130可以同时执行多个比较操作。在实施例中，协处理器130被配置用于执行多种类型的比较，例如，一个4字节比较、两个2字节比较、四个1字节比较，和/或一个2字节与一个1字节的比较。协处理器130通过存储器接口将搜索结果发送给片上存储器或片外存储器。

[0037] 如图1所示，存储器140存储命令142、输入数据144和输出数据146。反映于命令142中的命令指的是(1)用于存储至少一部分输入数据144的存储器140中的位置以及(2)与命令对应的操作(或多个操作)的结果(由协处理器130生成)将在存储器140中被存储的位置。

[0038] 在实施例中，协处理器130包含用于存储一个或多个命令的一个或多个地址的命令队列。在不忙时，协处理器130从命令队列中选择一个或多个地址(由管理程序120插入)，以便在存储器(例如，存储器140)中检索该一个或多个命令(例如，反映于命令142中)。

[0039] 协处理器控制块

[0040] 在实施例中，查询执行引擎110包含在由通用微处理器(未示出)执行时促使生成协处理器控制块(CCB)的指令。CCB是表示由查询执行引擎110发出的命令且包含协处理器被配置用于读取并处理的数据的数据结构。在实施例中，CCB包含指示协处理器130将要执行的操作的类型的命令类型数据以及与由该命令类型数据指示的操作对应的一个或多个操作数。如果协处理器130仅执行一个操作，则命令类型数据可以不是在CCB中的可用操作数。作为选择，如果协处理器被配置用于仅执行一个操作，则协处理器130可以忽略命令类型数据。

[0041] 命令类型数据指示协处理器130将使用哪个逻辑来处理命令。因而，不同的命令类型对应于由协处理器130实现的不同逻辑。例如，扫描操作要求协处理器130执行第一逻辑，而查找操作要求协处理器130执行与第一逻辑不同的第二逻辑。

[0042] 在CCB中指出的操作数可以是以下两种类型之一：立即操作数或间接操作数。立即操作数是在协处理器于没有首先要求翻译操作数的情况下执行比如存储器查找的操作时能够由协处理器立即使用的操作数。在扫描操作的情况下，立即操作数的一个实例是用于执行与表数据中的数据元素的比较的4字节整数。间接操作数是在协处理器能够执行指定的操作之前必须首先被翻译或被查找的操作数。间接操作数的一个实例是物理地址，该物理地址指示表数据存储于何处(例如，在存储器140中)以由协处理器执行操作，无论是扫描操作还是查找操作。

[0043] 在扫描操作的情况下，在CCB中指出的(扫描操作的)操作数包括：(a)指示被用来执行与表数据中的数据进行比较的数据的比较数据，以及(b)指示表数据位于何处的位置数据(例如，在存储器140中的输入数据144)。

[0044] 比较数据可以是任意类型的数据，比如数字、日期、字符或字符串。比较数据可以是单个值和/或值范围。另外，比较数据可以指示多个值和/或多个值范围。例如，查询可以

请求查看收入低于\$30,000的雇员的以及收入在\$100,000到\$130,000之间的雇员的姓名。在本例中,比较数据指示范围0~30,000以及范围100,000~130,000。

[0045] 位置数据可以是单个地址或多个地址,比如起始地址和结束地址或者起始地址和相对起始地址的偏移。在位置数据中指出的每个地址都可以是虚地址、实地址或物理地址。在实施例中,管理程序120以第二位置数据来替代在CCB中指出的位置数据。例如,(例如客户)操作系统识别在CCB中指出的位置数据,其中该位置数据是虚地址,并且以实地址来替代该虚地址。操作系统然后将CCB发送给管理程序120。管理程序120在映射表中查找被映射到实地址的物理地址,并且在CCB中以物理地址来替代实地址。

[0046] 在实施例中,CCB还包含指示协处理器130要将由CCB指示的操作的执行结果发送到何处的输出位置数据。在图1中,输出位置数据将指向存储器140内的输出数据146。如果多个微处理器被集成于同一芯片上并且每个微处理器都具有它自己的私有高速缓存,这会是有益的。因而,如果用于执行查询执行引擎110的特定的通用微处理器是同一芯片上的多个通用微处理器之一,并且每个微处理器与不同的(例如,L3)高速缓存关联,则查询执行引擎110可以指定与该特定的通用微处理器邻接的或在其附近的高速缓存,作为在CCB中的参数。因而,作为协处理器130将操作的结果发送给RAM的替换,协处理器130可以(基于在CCB中指出的输出位置数据)将结果不仅发送给高速缓存,而且还发送给与查询执行引擎110“最近”的特定的高速缓存。这样,查询执行引擎110没有被要求来请求(a)来自RAM的、(b)来自另一个微处理器的(或内核的)高速缓存的或者(c)来自为多个内核共享的共享高速缓存的结果,上述三种方式中的每一个都会比从微处理器的自己的(私有的)高速缓存中存取数据慢得多。相反,查询执行引擎110被允许指示由一个或多个硬件元件(即,在该实施例中为协处理器)执行的操作的结果将被存储于何处。

[0047] 协处理器

[0048] 一旦协处理器130经由管理程序120的接口接收到(或者直接来自查询执行引擎110的)命令(例如,以CCB的形式),协处理器130就执行与查询执行引擎110发出原始命令的线程异步的命令。如果协处理器130接收到多个命令,则协处理器130可以按照循环调度(roundrobin)的方式调度该多个命令来执行。某些命令可以并行地执行。

[0049] 在实施例中,用于命令的输入数据(例如,关系数据)经由接口来提取(到查询执行引擎110),并且命令的结果(即,协处理器130基于输入数据来生成的结果)经由该接口来写出。

[0050] 在实施例中,协处理器130促使完成状态在每个命令结束时经由接口写出到接口中的完成数据结构内。查询执行引擎110可以使用完成数据结构来与查询执行引擎110的一个或多个线程重新同步。

[0051] 图2是示出根据实施例的示例协处理器200的框图。协处理器200可以是在图1中的协处理器130。协处理器200包含存储器接口210、命令调度器220、解压器230、查询管道240和消息管道250。消息管道250处理存储器拷贝和消息传递。在解压器230解压了压缩的输入数据(例如,来自关系表的数据)之后,查询管道240处理一个或多个查询命令,比如扫描命令或查找命令。

[0052] 解压器230可以被配置为仅解压以单一格式压缩的数据。作为选择,解压器230可以被配置为解压以一种格式压缩的数据以及以另一种格式压缩的其它数据。在实施例中,

协处理器200不包含解压器230。在协处理器200接收数据时,如果协处理器200接收到的数据不是压缩的(例如,是已经解压的),则解压可以不是必需的。此外,如果协处理器200被配置用于在压缩数据上直接操作,不必首先解压该压缩数据,则即使对于压缩数据,解压也可以是非必需的。

[0053] 管道240和管道250的每个都与不同的命令队列集关联,并且可任选地与不同的命令格式集关联。管理程序120被配置用于确保:命令(反映于CCB中)被定向到正确管道的命令队列。在CCB中的标记位可以指出CCB是消息命令还是查询命令。

[0054] 管道240和管道250的每个都可以是多线程的,并且能够一次执行多个命令。多线程的程度对于软件是不公开的。命令调度器220可以对可用线程调度命令,假定该命令是可并行的。如果给定的命令需要串行化于另一个命令之后,则这两个命令可以被放置于同一命令队列内,并且在两个命令中可以设置适当的串行化标记。

[0055] 尽管未示出,协处理器200包含一定数量的存储器,用于在数据经由存储器接口210流入时存储数据,或者用于存储在查找操作中使用的数据,比如查找向量,其实例之一为布隆过滤器(Bloom Filter)。由于目前的芯片在很大程度上由用于主要内核或通用微处理器的一个或多个高速缓存构成,协处理器200的存储器的大小可以是很小的(例如,4KB)。

[0056] 在查询执行引擎110“查看”表数据关系并对其执行操作时,协处理器200仅“看见”数据的向量或单维数组或者在其上操作。换言之,协处理器200不“查看”多个列或行标识符。更确切地,对于数据在逻辑上被如何组织或存储,协处理器200是不知道的。因此,在给协处理器200提供指令时,查询执行引擎110确保由协处理器200执行的任何操作的输出按照特定的顺序来存储。如若不是,查询执行引擎110将不清楚输出与表的哪一部分对应。用以保持排序的一种方式是让查询执行引擎110保持对哪个表数据集与哪个CCB对应的跟踪,其中每个CCB都包含唯一的CCB标识符。然后,由协处理器200基于特定的CCB来生成的输出包含该特定CCB的标识符,以允许查询执行引擎110来确定输出与逻辑表的哪一部分对应。例如,查询执行引擎110可以存储使雇员(Employee)表的行1001-2000与CCB标识符432899关联起来的关联数据。协处理器200接收并处理具有标识符432899的CCB,以生成存储于特定位置的输出。

[0057] 作为选择,代替保持对CCB标识符的跟踪,查询执行引擎110存储使指示表的一部分(例如,雇员表的行1001-2000)的表数据与指示由协处理器200生成的输出将要存储于何处的输出位置数据(例如,物理地址1298737+4KB)关联的关联数据。后来,当查询执行引擎110检查存储于该存储位置的输出时,查询执行引擎110使用关联数据来确定表的那个部分对应于该输出。因而,即使协处理器200在不同的时间在表的不同部分上操作,并且即使查询执行引擎110可能指导多个协处理器在表的不同部分上操作,该操作可能同时执行,查询执行引擎110都能够保持对输出顺序的跟踪。

[0058] 处理查询

[0059] 图3是示出在实施例中的用于处理要求扫描操作的查询的过程300的流程图。在块310,查询执行引擎110接收将一个或多个数据对象(比如表)作为目标以及要求对该一个或多个数据对象中的至少一个进行扫描操作的查询。例如,查询执行引擎110可以处理SQL查询,以生成包含一个或多个数据库操作(包括扫描操作)的、查询执行引擎110被配置用于执行或指导一个或多个其它软件组件来执行的重写查询。作为选择,另一个软件组件接收原

始查询并且生成查询执行引擎110被配置用于处理的重写查询。

[0060] 在块320,查询执行引擎110基于一个或多个准则来确定在处理查询时是否涉及协处理器130。该一个或多个准则可以指示之前是否已生成并缓存结果,对表的索引是否存在并且是否可用于回答该查询来代替扫描表、表的尺寸等。例如,如果需要扫描的表的尺寸是相对小的,则相比在没有涉及协处理器130的情况下执行查询,涉及协处理器130可能要求更多的工作(例如,以执行查询执行引擎110的指令的通用微处理器的使用的形式)或者比在花费更多的时间。

[0061] 除此以外或作为选择,该一个或多个准则可以指示按照不同的方式来处理查询(或重写查询)的相对成本。例如,查询执行引擎110可以包含用于估计在不同的执行方案下执行查询的成本的成本估计器组件,不同的执行方案比如使用索引、在不使用协处理器130的情况下扫描表以及使用协处理器130来扫描表。然后,查询执行引擎110根据成本来选择最廉价的执行方案。“成本”可以基于一个或多个因素,例如,CPU使用情况、存储器使用情况、I/O使用情况和网络I/O使用情况。

[0062] 如果查询执行引擎110确定在执行查询时涉及协处理器130,则过程300进入块330。

[0063] 在块330,查询执行引擎110给管理程序120发送一个或多个指令的地址、输入数据的地址,以及输出数据将要被存储的地址。该一个或多个指令可以是查询执行引擎110生成的CCB的形式。管理程序120将地址由虚地址翻译成物理地址,并且将该物理地址放置于协处理器130的命令队列或缓冲区内。

[0064] 在查询执行引擎110促使该一个或多个指令被存储于存储器内(并且因而可供协处理器130读取)之后,查询执行引擎110可以执行与查询的执行相关的或者还与另一个查询相关的其它任务。这样,由协处理器130执行的操作与由查询执行引擎110执行的任务异步地执行,该查询执行引擎110由通用微处理器执行。

[0065] 如上所述,计算机系统100可以包含多个协处理器。因而,查询执行引擎110可以将指令(例如,CCB)发送给多个协处理器中的每个协处理器。这样,扫描操作或查找操作可以被划分成多个“微”操作,允许每个协处理器执行不同的“微”操作。例如,特定的表可以包含10,000行,并且可以有10个协处理器。然后,查询执行引擎110可以生成10个不同的CCB,这10个CCB每个都类似于其它的CCB,只是每个CCB指示用以存取该特定表的不同组的1,000行的不同地址。这样,这10个协处理器在该特定表的不同部分上并行操作。

[0066] 除此以外或作为选择,块330包含查询执行引擎110基于一个或多个准则来选择多个协处理器的子集来发送CCB。例如,查询执行引擎110可以只需要总共10个协处理器中的3个协处理器来各自执行扫描操作(但是在与每个其它协处理器相关不同的组的表数据上)。查询执行引擎110用来选择一个或多个协处理器的一个或多个准则可以是每个协处理器的当前负荷、每个协处理器的延迟和/或每个协处理器的处理历史。例如,查询执行引擎110选择当前最小“负荷”的或最不繁忙的3个协处理器。协处理器的负荷可以反映于处于协处理器的一个或多个命令队列中的命令数中。因而,等待由特定的协处理器处理的命令越多,该特定的协处理器就变得负荷越大。

[0067] 在块340,协处理器130接收一个或多个指令,并且执行反映于该一个或多个指令中的一个或多个操作。例如,协处理器130接收CCB,确定反映于该CCB中的操作(或多个操

作)的类型,读入完成该操作(或多个操作)必需的任何数据,执行该操作(或多个操作),并且(在块350中)促使操作(或多个操作)的结果被发送给查询执行引擎110。由协处理器130进行的命令执行可以通过由查询执行引擎110(或其代理之一)对协处理器130的一个或多个内部寄存器的写入来触发。

[0068] 扫描操作

[0069] 在实施例中,该一个或多个指令指示扫描操作以及存储有表数据的一个或多个地址。协处理器130检索表数据,并且在值或值范围(在该一个或多个指令中指定)与表数据之间执行比较。协处理器130通过存储器(例如,存储器140)来从查询执行引擎110中请求表数据,该存储器可以是系统中的动态RAM或者芯片上的高速缓冲存储器。表数据可以按块存储,所述块可以相对较大,比如64KB或更大。协处理器130可以将这些块中的每个块作为单维数组来存取。在列式数据库中,数据处于单维数组中,并且可以由协处理器130容易地读取。在行优先的数据库中,数据可以在由协处理器130处理之前首先被变换为列优先的格式。

[0070] 如果表数据跨越在存储器中不连续的块,则协处理器130单独请求每个块(作为单独的作业)。在此情形中,查询执行引擎将每个作业的结果(由协处理器130生成的)“拼接(stitch)”在一起。例如,在行优先的数据库中,数据将为跨步的,并且协处理器130将会每隔N个数据就选择一个,其中N在命令中指定。

[0071] 协处理器130可以“即时”,也就是,在表数据流入协处理器130时执行比较。一旦在表数据中的数据元素与在该一个或多个指令中指定的目标值或者值的目标范围进行比较,协处理器130就可以(立即或最终)使用来自表数据的新的数据元素来覆写用于存储该数据元素的存储器。

[0072] 可以配置协处理器以使其执行的比较操作的类型的实例包括大于(>)、小于(<)、等于(==)、不等于(!=)、大于或等于(>=)以及小于或等于(<=)。

[0073] 语义感知式压缩(SEMANTIC-AWARE COMPRESSION)

[0074] 在实施例中,协处理器130被配置用于对压缩数据进行操作。使用一种或多种非语义感知式的压缩技术来压缩某些数据,而可以使用一种或多种语义感知式压缩技术来压缩其它数据。使用非语义感知式的压缩技术压缩的数据要求在可以对已解压数据进行操作之前首先解压。使用语义感知式压缩技术压缩的数据可以不必在操作(例如,数字或字符串比较)执行之前进行解压。语义感知式压缩技术的一个实例是行程长度编码(RLE)。

[0075] RLE是一种数据压缩的形式,其中将数据行程(runs)(也就是,相同的数据值出现于许多连续的数据元素中的序列)存储为单个数据值和计数,而不是存储为原始行程的数据压缩的形式。这对于含有许多此类行程的数据最有用。例如,表的列可能含有下列序列的部门标识符:

[0076] AAAAABBBCCCCCCCCDDDDAAAA

[0077] 将RLE数据压缩算法应用于上述序列可以得出以下输出:6A3B8C5D4A。该行程长度编码仅以10个字符来表示原始的26个字符。在RLE中,单个数据值在输入序列中的行程越长,压缩程度就越大。

[0078] 返回至块340,表数据可以是行程长度编码的。因而,需要读入协处理器130内的表数据的数量以及协处理器130需要针对行程长度编码的表数据执行的比较次数可以大大小

于表数据不是行程长度编码的情形。考虑上述实例,代替执行26次比较(即,26个字符中的每个字符一次),协处理器130将仅需执行5次比较。

[0079] 针对行程长度编码的数据来执行扫描操作的结果自身可以是行程长度编码的,该结果可以最终由查询执行引擎110来处理。考虑上述实例,确定特定表的行是否包含部门标识符“A”的结果可以是6Y16N4Y,其中“Y”指示确定的肯定结果,而“N”指示确定的否定结果。

[0080] 作为选择,针对行程长度编码的数据来执行扫描操作的结果可以不是行程长度编码的。而是,结果可以被“解压”。考虑上述实例,确定特定表的行是否包含部门标识符“A”的结果可以是YYYYYYNNNNNNNNNNNNNNNNYYYY。在该实施例中,尽管在扫描操作期间针对字符“C”来执行单个比较,但是协处理器130对于该行程长度编码的数据元素生成了8个否定指示(例如,“N”或“0”)。

[0081] 查找操作

[0082] SQL查询经常需要交叉引用数据库中的多个表。处理这样的查询典型地包括交集操作。当前,交集操作由运行于通用微处理器上的软件执行,其中当被连接的表列的基小时,该软件使用向量查找(例如,布隆过滤器查找)。根据实施例,向量查找(或查找操作)以硬件来实现,这会比软件实现方式快得多。

[0083] 因而,在实施例中,(块330的)一个或多个指令指示查找操作、存储有表数据的一个或多个地址,以及存储有查找向量或数组的一个或多个地址。存储有表数据(或查找向量)的一个或多个地址可以是两个地址(例如,起始地址和结束地址)或者具有偏移的单个地址。协处理器130促使查找向量和表数据被发送给协处理器130,并且对于表数据中的每个数据元素,在查找向量中执行数据元素的查找。换言之,协处理器130使用数据元素(或者数据元素的散列)来在查找向量中识别位置,并且在查找向量的该位置处检索数据。

[0084] 要求查找操作的示例查询

[0085] 可能要求查找操作的查询的一个实例是请求有关居住于“富裕”的邮政编码区(zip code)内的“贫困”人口的信息的查询。“贫困”人员可以被看作是每年收入低于\$30,000的人,而“富裕”的邮政编码区可以被看作是其中中间薪酬超过\$100,000的邮政编码区。在本例中,协处理器130要求来自至少两个数据对象的数据:查找向量和人员(Person)表。查找向量(例如,以单个位)指示邮政编码区是否“富裕”。查找向量可以被预先计算出(即,在接收到查询之前),或者可以响应于接收到查询而计算出。

[0086] 人员表含有有关众多人员的信息,其中该表包含至少三列:一列为每个人的姓名,一列为每个人的薪酬,以及一列为每个人的邮政编码区。在一个方面中,指示邮政编码区是富裕还是贫困的数据是维度表,而含有有关每个人的信息的人员表是事实表。维度表典型地远小于事实表。在本例中,查找向量可以基于维度表来生成。与此相关地,事实表可以由多个数据对象(例如,表)表示:一个数据对象可以含有有关每个人的薪酬的信息,而另一个数据对象可以含有有关每个人的邮政编码区的信息。

[0087] 查找向量

[0088] 查找向量的一个实例是布隆过滤器,该过滤器是被用来检验元素是否为集合的成员的概率数据结构。虽然在使用布隆过滤器时可能误报,但是不会漏报。布隆过滤器与一个或多个散列函数关联,每个散列函数将元素映射到在布隆过滤器中的一个数组位置。

[0089] 但是,查找向量不必是概率性的。例如,如果只有10,000个可能的邮政编码区,并

且每个邮政编码区都与指示该邮政编码区是否为“富裕的”单个位关联，则查找向量的尺寸（例如，1.25KB）可以小到足以使整个查找向量装入协处理器130的存储器（例如，SRAM）。因此，为了减小其尺寸，概率查找向量并非是必要的。

[0090] 图4是示出示例查找向量400的一部分的框图。在查找向量400中的每个位置与不同的邮政编码区关联。换言之，邮政编码区被用来索引到查找向量400中。在查找向量400中的每个位置含有单个位，指示相应的邮政编码区是“富裕”的邮政编码区（“1”）还是“贫困”的邮政编码区（“0”）。

[0091] 如果查找向量不能够完全装入协处理器130的存储器，则（除了生成概率查找向量外）要么（a）查找向量可以被分开（或被划分），使得协处理器130每次为查找向量的每个部分读入表数据，要么（b）在有多个协处理器的实施例中，不同的协处理器可以一次读入表数据，但是仅针对它存储的查找向量的一部分。

[0092] 在前一种情形中，如果协处理器130仅能够使例如查找向量的1/4部分装入其存储器，则协处理器130四次（即，协处理器130所读入的查找向量的每个部分各一次）读取指示人员的邮政编码区的表数据。如果人员的邮政编码区在表数据的任一次传递中都没有被识别出，则该人员的结果可以指示否定的判定。表数据的每次传递的结果可以是一个长的位数组，在人员表中指出的每个人用一个位。

[0093] 在后一种情形中，四个不同的协处理器可以存储查找向量的不同的1/4部分，并且一次性读入指示人员的邮政编码区的表数据，并然后为反映于表数据中的每个邮政编码区执行对查找向量的查找。此外，来自每个协处理器的查找操作的结果可以是一个长的位数组，在人员表中指出的每个人用一个位。

[0094] 生成扫描或查找操作的结果

[0095] 在实施例中，协处理器130生成具体输出格式，作为执行反映于一个或多个指令中的一个或多个操作的结果。具体输出格式的一个实例是位向量，其中在该位向量中的每个位置要么指示真要么指示假。例如，如果“10”为目标值，并且比较是确定该目标值是否大于表中的数据值或数据元素，则比较的结果（a）在数据值大于10时为真，并且（b）在数据值小于或等于10时为假。

[0096] 在向量中的每个位置对应于接收自输入（例如，表）数据的数据值或数据元素。例如，在上述查找操作的实例中，协处理器130生成反映居住于“富裕”的邮政编码区的“贫困”人员的位向量。为了生成该位向量，协处理器130从人员表的邮政编码列中读入数据，并且为在读入数据中指出的每个人且基于查找向量来确定该人是否居住于“富裕”的邮政编码区内。在位向量中的每个位指出在人员表中的不同人是否居住于“富裕”的邮政编码区内。协处理器130随后将该位向量传递给通用微处理器，该通用微处理器使用位向量来识别还被认为是“贫困”的人员。例如，对于居住于富裕的邮政编码区内的每个人（如位向量所示），通用微处理器查找人员表中的相应行，以确定该人是否是“贫困”的。

[0097] 因为位向量是尺寸相对小的，所以由通用微处理器处理该位向量是相对较快的；比通用处理器直接处理输入数据快得多。而且，存储该位向量所需的高速缓冲存储器空间远小于存储输入数据（例如，一大列数据）所要求的高速缓冲存储器空间。

[0098] 生成扫描或查找操作的结果

[0099] 返回至过程300，在块350，协处理器130促使扫描操作（或查找操作）的结果将对查

询执行引擎110可用。块350可以包括：协处理器130将操作的结果发送给在来自启动该操作的查询执行引擎110的一个或多个指令中指定的存储器。例如，查询执行引擎110生成乐CCB，并且在该CCB中指出：相应操作的结果将被发送给例如DRAM、共享的L3高速缓存或者具体微处理器（例如，运行查询执行引擎的微处理器或者与生成原始命令的微处理器不同的微处理器）的高速缓存。

[0100] 块350还可以包括：协处理器130设定标记，该标记在被置位时指示操作已完成。该标记设定充当这样的信号：(1) 管理程序120将新的请求插入协处理器130的命令队列中，以及(2) 查询执行引擎110检索结果。

[0101] 作为选择，协处理器130被编程为总是将操作的结果发送给可由查询执行引擎110访问的具体存储器组件（无论是RAM还是高速缓存）。

[0102] 作为选择，代替使扫描操作的结果对查询执行引擎110可用，协处理器130将结果发送给另一个协处理器。如同前面所述的，计算机系统100可以包含多个协处理器。该多个协处理器可以串联连接。在两个协处理器之间的每个连接都可以包含FIFO缓冲区，使得与“消费”协处理器能够处理结果相比，“生产”协处理器可以更快地生成结果。最终，在“链”上的最后一个协处理器将结果发送给可由查询执行引擎110访问的指定目的地，比如在RAM或高速缓存内。

[0103] 在块360，查询执行引擎110处理结果，并且执行一个或多个其它操作，以便生成原始查询的最终结果。在协处理器130执行扫描或查找操作时，运行查询执行引擎110的通用微处理器可以是空闲的，或者可以由查询执行引擎110或连同另一个进程一起来使用。例如，查询执行引擎110可以执行一个或多个其它操作，该一个或多个其它操作是查询所要求的或者按照任何方式不与该查询相关而是与另一个查询相关的。

[0104] 作为扫描操作的一个实例，查询可以请求在具体的日期范围内发起的采购订单的ID和价格。在本例中，协处理器130执行包括读入多个采购订单的日期信息的扫描操作，其中日期信息反映于采购订单表中。扫描操作的结果可以是一系列位（例如，位向量），这些位中的每一个都反映出相应的采购订单是否是在指定的日期范围内发起的。协处理器130将结果发送给可由查询执行引擎110访问的存储器，并且可以通过对查询执行引擎110周期性检查的标记置位来通知查询执行引擎110扫描操作的完成。然后，查询执行引擎110使用这些位在采购订单表中识别出与在指定的日期范围内发起的那些采购订单对应的条目。然后，查询执行引擎110在所识别的条目中识别出ID和价格，并且返回（例如，显示）该信息作为查询的结果。查询还可以指定：查询的结果将要按照价格降序的方式来排序。因而，查询执行引擎110在接收到由协处理器110执行的扫描操作的结果之后执行一个或多个操作。

[0105] 考虑其中查询将要识别出居住于“富裕”的邮政编码区内的“贫困”人员的查找操作实例，除了指导协处理器130来执行查找操作之外，查询执行引擎110还可以让所指导的协处理器130（或另一个协处理器）对人员表执行扫描操作，以识别出年薪低于\$30,000的所有人。扫描操作的结果（像查找操作的结果一样）可以是一系列位（例如，位向量）的形式，其中每个位对应于在人员表中指出的不同的人。在一种实施例中，查询执行引擎110对作为输入的扫描操作的结果和查找操作的结果执行与（AND）操作。作为选择，协处理器130（或另一个协处理器）可以被编程用于执行与（AND）操作。在该实施例中，查询执行引擎110可以创建另一个CCB，在该CCB中，操作数包括查找操作的结果的地址（例如，虚地址）以及扫描操作的

结果的地址。

[0106] 如前所述,查找向量的尺寸可以不是一次完全装入协处理器130的存储器。在前面描述的两种情形之一中,查找向量被划分成四个“微”向量,并且协处理器130单独对每个微向量进行操作,从而要求协处理器130为每个人(从人员表中)读入邮政编码区数据四次。由执行该查找操作的协处理器130产生的总的结果可以包含四个单独的位数组,它们最终在一起求或(OR),以得到单个位数组(同样,在人员表中指出的每个人用一个位)。该或(OR)步骤(该步骤可以包括三次或(OR)操作)可以由查询执行引擎110执行。作为选择,协处理器130可以被编程用于执行或(OR)操作。

[0107] 在上述两种情形的另一种情形中,查询执行引擎110促使四个不同的协处理器使用查找向量的不同部分来执行查找操作。然后,来自一个协处理器的结果与来自其它协处理器中的每个协处理器的结果求或(OR),以得出单个位数组(在人员表中指出的每个人用一个位)。同样,该或(OR)步骤可以由查询执行引擎110或者由协处理器之一执行。

[0108] 一旦查询执行引擎110确定了哪些人居住于“富裕的”邮政编码区,查询执行引擎110就使用该信息来确定也被认为是“贫困的”那些人,如同上文所指出的。

[0109] 虽然以上的描述指的是执行扫描操作或查找操作,但是实施例可以包括:一个协处理器为特定的查询执行扫描操作,而另一个协处理器为特定的查询执行查找操作。因而,多个协处理器可以同时为同一查询来执行,但是执行不同的操作。

[0110] 本文所描述的实施例的优点在于:通用微处理器可以将数据密集型的操作卸下给与微处理器分开的一个或多个协处理器,以便为其它任务空出微处理器供使用。因而,协处理器可以相对促使协处理器执行操作的查询处理软件异步地进行操作。另外,与通用微处理器运行查询处理软件相比,该一个或多个协处理器可以更加快速地执行那些操作。

[0111] 硬件概况

[0112] 根据一种实施例,本文所描述的技术由一个或多个专用计算设备来实现。专用计算设备可以被硬连线以执行这些技术,或者可以包含被永久编程来执行这些技术的、比如一个或多个专用集成电路(ASIC)或现场可编程门阵列(FPGA)的数字电子设备,或者可以包含被编程来依照在固件、存储器、其它存储器或者它们的组合中的程序指令而执行这些技术的一个或多个通用硬件处理器。这样的专用计算设备还可以将定制的硬连线逻辑、ASIC或FPGA与定制编程结合,以完成这些技术。专用计算设备可以是台式计算机系统、便携式计算机系统、手持式设备、网络设备,或者用于合并硬连线和/或程序逻辑来实现这些技术的任何其它设备。

[0113] 例如,图5是示出本发明的实施例可以实施于其上的计算机系统500的框图。计算机系统500包含总线502或者用于传输信息的其它通信机构,以及与总线502耦接用于处理信息的硬件处理器504。硬件处理器504可以是例如通用微处理器。

[0114] 计算机系统500还包含与总线502耦接用于存储信息以及待由处理器504执行的指令的主存储器506,例如,随机存取存储器(RAM)或者其它动态存储设备。主存储器506在待由处理器504执行的指令的执行期间还可以用于存储临时变量或者其它中间信息。这样的指令,在存储于可由处理器504访问的非瞬时存储介质时,使计算机系统500成为被定制用于执行在指令中指定的操作的专用机器。

[0115] 计算机系统500还包含与总线502耦接用于为处理器504存储静态信息和指令的只

读存储器(ROM)508或其它静态存储设备。存储设备510,例如,磁盘、光盘或固态硬盘驱动器,被提供并与总线502耦接,用于存储信息和指令。

[0116] 计算机系统500可以经由总线502耦接至用于给计算机用户显示信息的显示器512,例如,阴极射线管(CRT)。包含字母数字键和其它键的输入设备514与总线502耦接,用于将信息和命令选择传输给处理器504。另一个类型的用户输入设备是用于将方向信息和命令选择传输给处理器504的并且用于在显示器512上控制光标运动的光标控制516,例如,鼠标、轨迹球或光标方向键。该输入设备典型地在两个轴(第一轴(例如,x)和第二轴(例如,y))上具有两个自由度,这两个自由度允许件指定平面内的位置。

[0117] 计算机系统500可以使用下列设备来实现本文所描述的技术:定制的硬连线逻辑、一个或多个ASIC或FPGA、固件和/或结合计算机系统来促使计算机系统500或对其编程使其成为专用机器的程序逻辑。根据一种实施例,此处的技术由计算机系统500响应于处理器504执行包含于主存储器506内的一个或多个指令的一个或多个序列而执行。这样的指令可以从比如存储设备510的另一个存储介质读入主存储器506。包含于主存储器506内的指令序列的执行促使处理器504执行此处所描述的过程步骤。在可替换的实施例中,硬连线电路可以代替软件指令或者与其结合而被使用。

[0118] 此处所使用的术语“存储介质”指的是存储促使机器按照具体的方式来操作的数据和/或指令的任何非瞬时介质。这样的存储介质可以包含非易失性的介质和/或易失性的介质。非易失性的介质包括光盘、磁盘或固态硬盘驱动器,比如存储设备510。易失性的介质包括动态存储器,比如主存储器506。存储介质的常见形式包括例如软盘、软磁盘、硬盘、固态硬盘驱动器、磁带,或者任何其它磁性数据存储介质、CD-ROM、任何其它光学的数据存储介质、任何具有孔图形的物理介质、RAM、PROM,以及EPROM、FLASH-EPROM、NVRAM、任何其它存储芯片或存储盒。

[0119] 存储介质不同于传输介质,但可以与传输介质结合起来使用。传输介质参与在存储介质之间传输信息。例如,传输介质包括同轴电缆、铜导线和光纤,包括包含总线502的导线。传输介质还能够采取声波或光波的形式,例如,在无线电波和红外数据通信期间生成的那些波。

[0120] 各种形式的介质都会参与将一个或多个指令的一个或多个序列输送给处理器504以用于执行。例如,指令最初可以载存于远程计算机的磁盘或固态硬盘驱动器上。远程计算机能够将指令加载到其动态存储器内,并且使用调制解调器经由电话线来发送该指令。计算机系统500本地的调制解调器能够接收电话线上的数据,并且使用红外发射器将数据转换为红外信号。红外检测器能够接收红外信号所携带的数据,并且适当的电路能够将数据放置于总线502上。总线502将数据输送给主存储器506,处理器504从该主存储器506中检索并执行指令。由主存储器506接收的指令可以要么在由处理器504执行之前,要么在其之后可选地存储于存储设备510上。

[0121] 计算机系统500还包含与总线502耦接的通信接口518。通信接口518提供耦接至与本地网络522连接的网络链路520的双向数据通信。例如,通信接口518可以是综合业务数字网络(ISDN)卡、电缆调制解调器、卫星调制解调器,或者用于提供到相应类型的电话线的数据通信连接的调制解调器。作为另一个实例,通信接口518可以是局域网(LAN)网卡,用于提供数据通信连接到可兼容的LAN。还可以实现无线链路。在任何这样的实施方式中,通信接

口518发送及接收用于承载表示各种类型的信息的数字数据流的电信号、电磁信号或光信号。

[0122] 网络链路520典型地通过一个或多个网络来提供到其它数据设备的数据通信。例如,网络链路520可以通过本地网络522提供到主机计算机524的或者到由网络服务提供商(ISP)526运营的数据设备的连接。ISP 526进而通过全球分组数据通信网络(现在常称为“因特网(Internet)”528)来提供数据通信服务。本地网络522和因特网528两者都使用承载数字数据流的电信号、电磁信号或光信号。通过各种网络的信号以及在网络链路520上的且经过通信接口518的信号是传输介质的实例形式,这些信号将数字数据送入和送出计算机系统500。

[0123] 计算机系统500能够通过网络、网络链路520和通信接口518来发送消息和接收数据,包括程序代码。在因特网的实例中,服务器530可以通过因特网528、ISP 526、本地网络522和通信接口518来发送应用程序的请求代码。

[0124] 所接收的代码可以在其被接收到时由处理器504执行,和/或被存储于存储设备510或其它非易失性的存储器内,待以后执行。

[0125] 在前面的详细说明中,本发明的实施例已经参考众多具体的细节进行了描述,这些细节在不同的实施方式之间可以是不同的。因此,本说明书和附图应当被看作是说明性的,而不是限制性的。本发明的范围的独有指标以及本申请人想要其成为本发明的范围的内容,是由本申请按照此类权利要求书发布的具体形式来发布的权利要求书的字面及等效范围,包括任何后续校正。

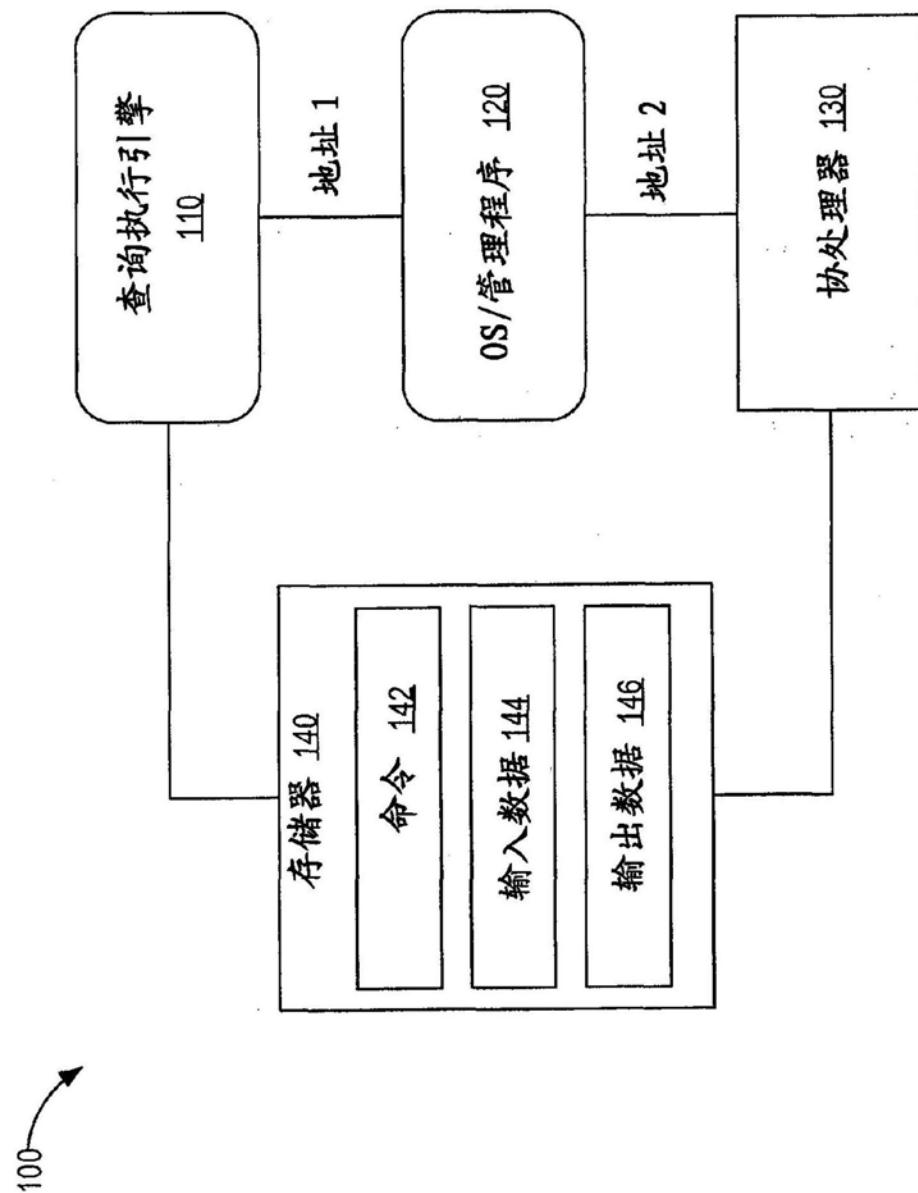


图1

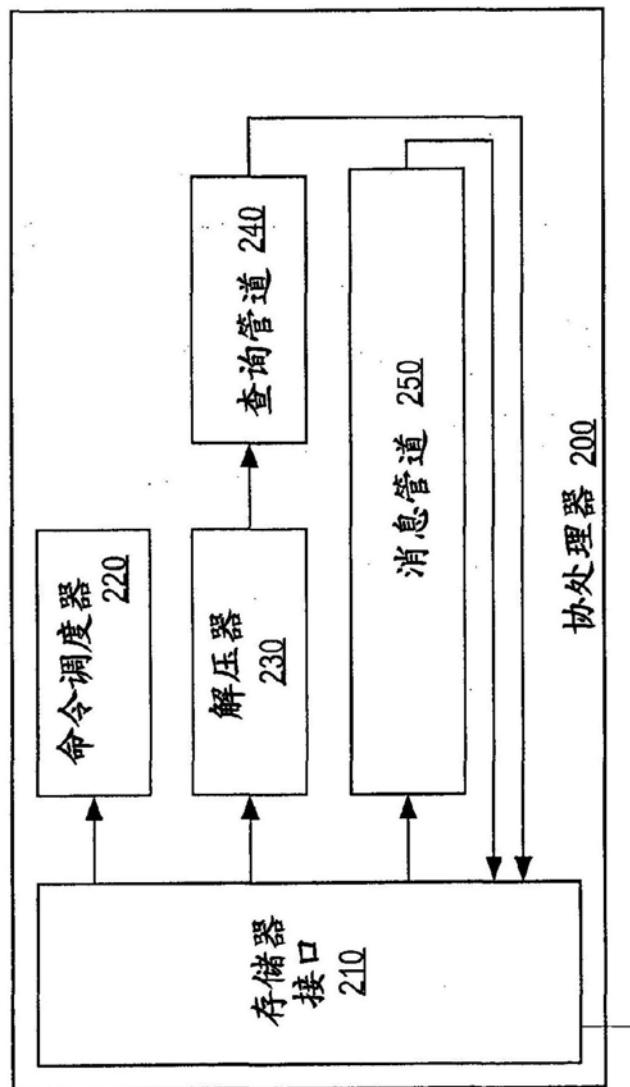


图2

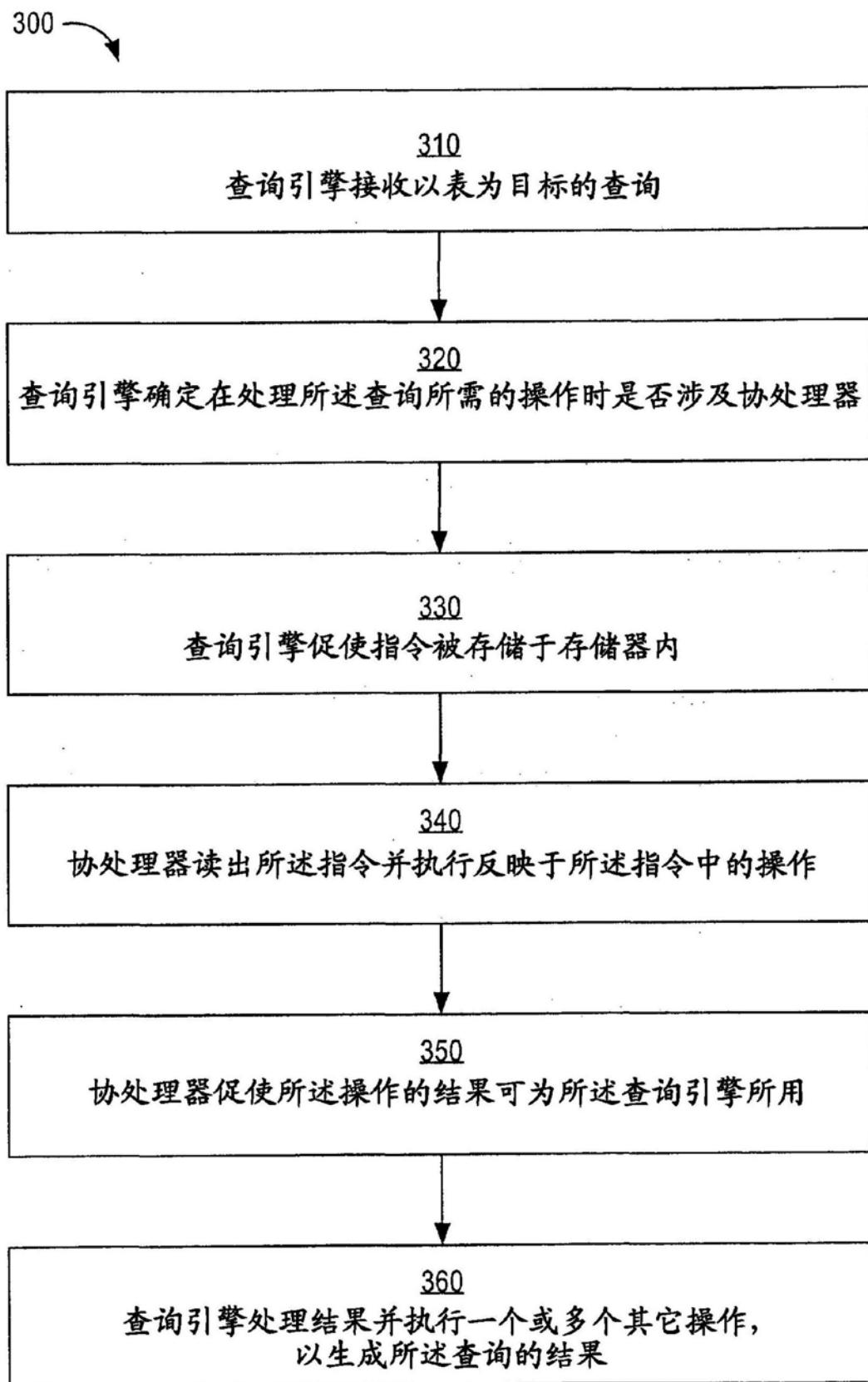


图3

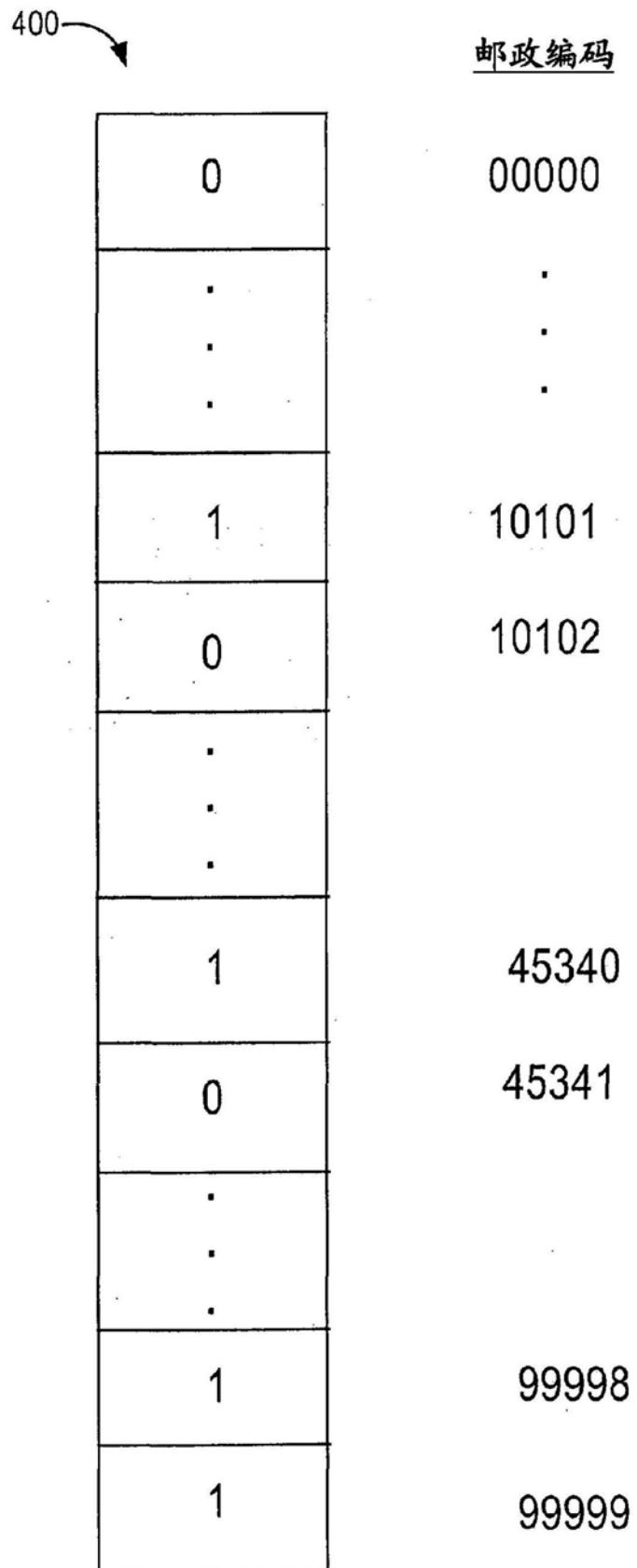


图4

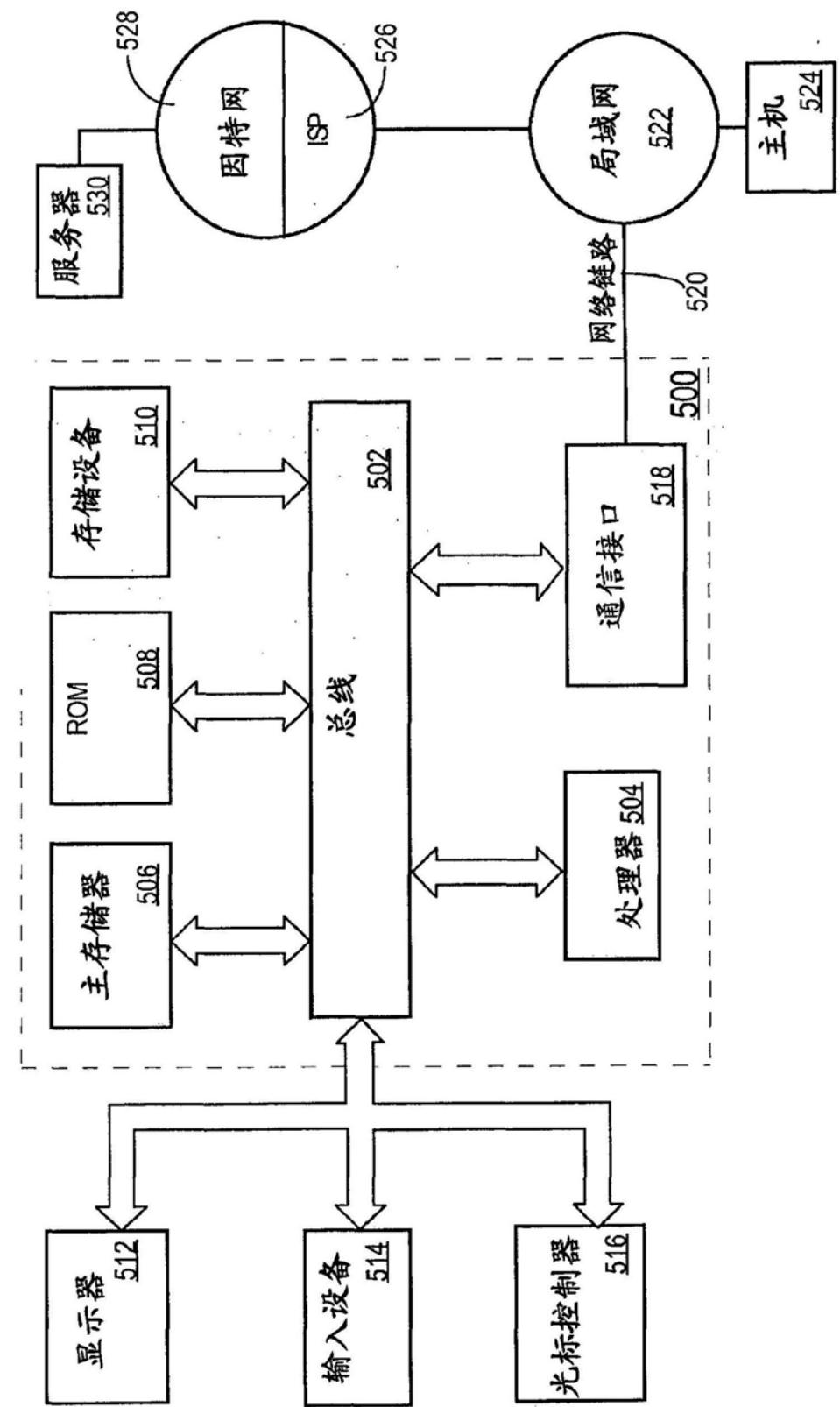


图5