



[12] 发明专利申请公开说明书

[21] 申请号 200610057720.7

[43] 公开日 2006 年 11 月 1 日

[11] 公开号 CN 1855026A

[22] 申请日 2006.2.23

[21] 申请号 200610057720.7

[30] 优先权

[32] 2005.4.28 [33] US [31] 11/116,626

[71] 申请人 国际商业机器公司

地址 美国纽约

[72] 发明人 赫尔曼·李·布莱克蒙

菲利普·罗杰斯·希利尔三世

约瑟夫·艾伦·基尔希特

布赖恩·T·范德普尔

[74] 专利代理机构 北京市金杜律师事务所

代理人 王茂华

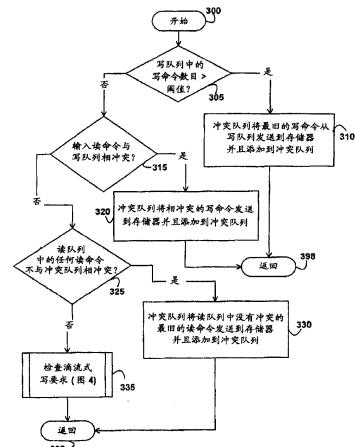
权利要求书 2 页 说明书 12 页 附图 5 页

[54] 发明名称

用于选择命令发送到存储器的方法、存储器控制器和系统

[57] 摘要

一种方法、设备、系统和信号承载介质，在实施例中选择命令以发送到存储器。在实施例中，如果如下情况中的一些或全部为真，则将写队列中不与冲突队列相冲突的最旧的命令发送到存储器和添加到冲突队列：读队列中的所有命令与冲突队列相冲突，来自处理器的任何读命令不与写队列相冲突，写队列中的命令数目大于第一阈值，以及冲突队列中的所有命令已经出现小于第二阈值。在实施例中，如果命令不与队列中的命令访问存储器中的同一高速缓存线，则该命令不与该队列相冲突。以此方式，在实施例中，写命令在减小了对于读命令性能的影响的时刻发送到存储器。



1. 一种方法，包括：

确定写队列中的命令数目是否大于第一阈值；

如果该确定为真，则判断冲突队列中的所有命令在该冲突队列中是否已经出现小于第二阈值周期数目；以及

如果该判断为真，则选择该写队列中不与该冲突队列中的任何命令相冲突的最旧的命令。

2. 权利要求 1 的方法，进一步包括：

将不与该冲突队列中的任何命令相冲突的最旧的命令从该写队列发送到存储器。

3. 权利要求 2 的方法，进一步包括：

将不与该冲突队列中的任何命令相冲突的最旧的命令添加到该冲突队列。

4. 权利要求 1 的方法，其中该选择进一步包括：

确定该写队列中的该最旧的命令与该冲突队列中的任何命令是否要访问存储器中的同一个高速缓存线。

5. 权利要求 1 的方法，其中该冲突队列包括先前发送到存储器的命令。

6. 权利要求 1 的方法，其中该确定进一步包括：

确定读队列中的所有命令是否与该冲突队列相冲突。

7. 权利要求 1 的方法，其中该确定进一步包括：

确定来自处理器的任何读命令是否不与该写队列相冲突。

8. 一种存储器控制器，包括：

写队列；以及

冲突队列，如果该写队列中的命令数目大于第一阈值，并且该冲突队列中的所有命令在该冲突队列中已经出现小于第二阈值周期数目，则该冲突队列将该写队列中不与该冲突队列中的任何命令相冲突的最旧的命令发送到存储器。

9. 权利要求 8 的存储器控制器，其中如果该冲突队列将最旧的命令发送到该存储器，则该冲突队列进一步将该最旧的命令添加到该冲突队列。

10. 权利要求 8 的存储器控制器，其中该冲突队列进一步确定该写队列中的不与该冲突队列中的任何命令相冲突的最旧的命令，不与该冲突队列中的任何命令访问该存储器中的同一个高速缓存线。

11. 权利要求 8 的存储器控制器，其中该冲突队列包括先前发送到该存储器的命令。

12. 权利要求 8 的存储器控制器，其中该冲突队列基于如果读队列中的所有命令与该冲突队列相冲突，进一步将该写队列中不与该冲突队列中的任何命令相冲突的最旧的命令发送到存储器。

13. 权利要求 8 的存储器控制器，其中该冲突队列基于如果来自处理器的任何读命令不与该写队列相冲突，将该写队列中不与该冲突队列中的任何命令相冲突的最旧的命令发送到存储器。

14. 一种计算机系统，包括：

处理器；

存储器；以及

如权利要求 8 至 13 的任一项中限定的从处理器接收命令的存储器控制器。

15. 权利要求 14 的计算机系统，其中该存储器控制器进一步包括第一寄存器和第二寄存器，以及其中该存储器控制器进一步将第一阈值存储于该第一寄存器中和将第二阈值存储于该第二寄存器中。

用于选择命令发送到存储器的方法、存储器控制器和系统

技术领域

本发明一般涉及计算机系统，更具体地涉及发送命令到存储器的存储器控制器。

背景技术

常常将 1948 年的 EDVAC 计算机系统的开发称为计算机时代的开始。从那时起，计算机系统已经发展成在许多不同环境下可以找到的极为复杂的装置。计算机系统典型地包括硬件(例如半导体、电路板等)和软件(例如计算机程序)的组合。随着半导体处理和计算机体系结构的发展将计算机硬件的性能向更高水平推动，更为复杂的计算机软件已经得到了发展，以利用硬件的更高性能，带来了比仅仅数年前强大得多的当今计算机系统。

对计算机系统的性能可以具有巨大影响的计算机系统的一个组件是存储器子系统。计算机系统典型地包括经由处理器总线连接到存储器子系统的一个或多个处理器。存储器子系统典型地包括经由存储器数据总线和相关联的控制信号连接到一个或多个同步动态随机访问存储器(SDRAM)的存储器控制器。存储器控制器负责接受来自处理器的存储器读和写(加载和存储)命令或请求、与 SDRAM 相联系以执行读或写命令、以及将与读操作相关联的任何数据从 SDRAM 返回到处理器。存储器控制器还具有管理存储器存储体(bank)定时、最大化存储器数据总线的利用、以及优化存储器访问，比如在可能时使得读优先于写，的任务。

优化存储器访问对于计算机系统性能的提高已经是一个基本的问题。存储器控制器努力解决的一个存储器访问问题是读命令与写命令的选择。一般地，读命令对于系统性能是限制性因素，因为处理器必须等待读取数据，以便继续执行指令，或者处理器预取读取数据、然后试图

隐藏存储器时延。写命令一般不是严重的问题，因为一旦处理器将写数据发送到存储器控制器，处理器认为它们就完成了。然而，存储器控制器具有固定数目的缓冲器(或队列)用以保持写命令和数据。一旦缓冲区近乎充满，由于存储器控制器中的写队列可能溢出，存储器控制器就需要发信号通知处理器不能接受新的命令了，这就拖延了总线。该拖延可能造成所有处理器命令的额外延迟；于是，对于读命令写命令可能成为一个性能问题。

于是，需要一种选择适当的时间以将写命令发送到存储器的更佳方式，以便允许提高性能。

发明内容

提供了一种方法、设备、存储器控制器、系统和信号承载介质，在实施例中，选择命令以发送到存储器。在实施例中，如果如下情况中的一些或全部为真，则将写队列中不与冲突队列相冲突的最旧的命令发送到存储器和添加到冲突队列：读队列中的所有命令与冲突队列相冲突，来自处理器的任何读命令不与写队列相冲突，写队列中的命令数目大于第一阈值，以及冲突队列中的所有命令已经出现小于第二阈值。在实施例中，如果命令不与队列中的命令访问存储器中的同一高速缓存线，则该命令不与该队列相冲突。以此方式，在实施例中，写命令在减小了对于读命令性能的影响的时刻发送到存储器。

附图说明

下文结合附图来描述本发明的各种实施例：

图 1 描绘了用于实现本发明实施例的示例系统的高层次框图。

图 2 描绘了按照本发明实施例的示例存储器控制器的框图。

图 3 描绘了按照本发明实施例的存储器控制器的示例处理流程图。

图 4 描绘了按照本发明实施例的存储器控制器的另一示例处理流程图。

图 5 描绘了按照本发明实施例的存储器控制器的另一示例处理流程

图。

然而，注意的是，附图仅图示了本发明的示例实施例，因此不应认为限制其范围，因为本发明可允许其他同等有效的实施例。

具体实施方式

参见附图，其中相似的编号表示贯穿这几幅附图的相似部分。图 1 描绘了按照本发明实施例的计算机系统 100 的高层次框图表示，该计算机系统经由网络 130 连接到服务器计算机系统 132。仅为了方便而使用了名称“计算机系统”和“服务器”，并且在实施例中，对于一个计算机工作为客户机的计算机可对于另一计算机工作为服务器，反之亦然，而且可使用任何适当的电子装置。在实施例中，计算机系统 100 的硬件组件可由 IBM eServer iSeries 或者 pSeries 计算机系统来实现。然而，本领域的技术人员将理解，本发明实施例的机制和设备同等地应用于任何适当的计算系统。

计算机系统 100 的主要组件包括一个或多个处理器 101、主存储器 102、存储器控制器 106、终端接口 111、存储装置接口 112、I/O(输入/输出)装置接口 113 和通信/网络接口 114，所有组件经由存储器总线 103、I/O 总线 104 和 I/O 总线接口单元 105 相连接用于组件间的通信。

计算机系统 100 包含一个或多个可编程的中央处理单元 (CPU)101A、101B、101C 和 101D，这里整体上称为处理器 101。处理器 101 可包括通用处理器、专用处理器和/或状态机，并且可经由总线 103 与存储器控制器 106 连接，以发出命令比如加载和存储(读和写)命令以便访问存储器 102。在实施例中，计算机系统 100 包含相对大型的系统典型具有的多个处理器；然而，在另一实施例中，作为代替，计算机系统 100 是单个 CPU 系统。每个处理器 101 执行在主存储器 102 中存储的指令，并且可包括一级或多级板上高速缓存。

主存储器 102 是用于存储数据和程序的随机访问半导体存储器。在另一实施例中，主存储器 102 代表计算机系统 100 的整个虚拟存储器，并且还可包括连接到计算机系统 100 或者经由网络 130 连接的其他计算

机系统的虚拟存储器。主存储器 102 在概念上是单个的单片实体，但是在其他实施例中主存储器 102 是更复杂的配置，比如高速缓存和其他存储器装置的分等级体系。例如，存储器可以以多级高速缓存存在，并且这些高速缓存可进一步通过功能来划分，从而一个高速缓存保持指令，而另一个保持非指令数据，其由处理器使用。可将不同的 CPU 或 CPU 集进一步分配给存储器并与之相关联，正如在任何各种所谓的非一致性存储器访问(NUMA)计算机体系结构中所知的那样。在各种实施例中，存储器 102 可以是任何级别的高速缓存或者主存储器。在一种实施例中，将存储器 102 实现为 DRAM(动态随机访问存储器)，但是在其他实施例中可使用任何适当的存储器技术。

存储器控制器 106 与存储器 102 之间的接口一般包括命令信号，比如由存储器控制器 106 驱动的经由具有数据线和数据选通的双向数据总线到存储器的读地址选通(RAS)和列地址选通(CAS)。由存储器控制器 106 驱动的每个命令信号典型地包含多个时延周期。另外，在例如存储器范围的存储体上执行这些命令之后，在该存储体能够进行另一操作之前，该存储体通常需要额外的周期来恢复。例如，为了实现读或加载，存储器控制器 106 打开该存储体，发出读命令，等待用于 CAS 时延的必要周期，并且接收来自存储器的数据突发。在该数据突发到存储器控制器之后，该存储体要求多个周期以预充电关联于该加载的内部总线的行。

在实施例中，存储器控制器 106 包括处理器(类似于处理器 101)和能够在其处理器上执行的指令或者能够由在其处理器上执行的指令解释的声明，以执行如以下参照图 3、4 和 5 进一步描述的功能。在另一实施例中，存储器控制器 106 可以以微码或固件实现。在另一实施例中，存储器控制器 106 可以借助于逻辑门和/或其他适当的硬件技术以硬件实现。下面参照图 2 进一步描述了存储器控制器 106。

存储器总线 103 提供了用于在处理器 101、存储器控制器 106 和 I/O 总线接口单元 105 之间传送数据的数据通信路径。I/O 总线接口单元 105 进一步连接到用于传送数据到各种 I/O 单元和传送来自各种 I/O 单元的

数据的系统 I/O 总线 104。I/O 总线接口单元 105 通过系统 I/O 总线 104 与多个 I/O 接口单元 111、112、113 和 114 通信，I/O 接口单元也称为 I/O 处理器(IOP)或 I/O 适配器(IOA)。系统 I/O 总线 104 可以例如是工业标准 PCI 总线或任何其他适当的总线技术。

I/O 接口单元支持与多种存储装置和 I/O 装置的通信。例如，终端接口单元 111 支持一个或多个用户终端 121、122、123 和 124 的配属。存储装置接口单元 112 支持一个或多个直接访问存储装置(DASD)125、126 和 127(它们一般是旋转磁盘驱动器存储装置，不过作为代替，它们可以是其他装置，包括配置为对于主机表现为单个大型存储装置的盘驱动器阵列)的配属。主存储器 102 的内容可存储到直接访问存储装置 125、126 和 127 中以及从其取回。

I/O 和其他装置接口 113 为任何各种其他输入/输出装置或其他类型的装置提供了接口。两个这样的装置，打印机 128 和传真机 129，在图 1 的示例性实施例中示出，但是在其他实施例中可能存在许多其他这样的装置，它们可以是不同类型的。网络接口 114 提供从计算机系统 100 到其他数字装置和计算机系统的一个或多个通信路径；这样的路径可包括例如一个或多个网络 130。

尽管存储器总线 103 在图 1 中表示为相对简单的单个总线结构，其在处理器 101、存储器控制器 106 和 I/O 总线接口 105 之间提供直接的通信路径，但是事实上存储器总线 103 可包括多个不同的总线或通信路径，其可以以各种形式的任一种来配置，比如分等级的、星形或网状配置中的点对点链路、多个分等级的总线、并行和冗余的路径、或者任何其他适当的配置类型。而且，尽管 I/O 总线接口 105 和 I/O 总线 104 表示为单个的各个单元，但是计算机系统 100 事实上可包含多个 I/O 总线接口单元 105 和/或多个 I/O 总线 104。尽管示出了多个 I/O 接口单元，它们将系统 I/O 总线 104 与通向各种 I/O 装置的各种通信路径相隔离，但是在其他实施例中，一些或全部 I/O 装置直接地连接到一个或多个系统 I/O 总线。

图 1 中所示的计算机系统 100 具有多个配属的终端 121、122、123

和 124，比如可以是典型的多用户“大型机”计算机系统。一般地，在这样的情况下，配属的装置的实际数目大于图 1 中所示的，尽管本发明不限于任何特定大小的系统。作为替代，计算机系统 100 可以是单用户的系统，一般仅包含单个用户显示器和键盘输入，或者可以是几乎没有或根本没有直接的用户接口但是从其他计算机系统(客户机)接收请求的服务器或类似装置。在其他实施例中，计算机系统 100 可实现为个人计算机、便携式计算机、膝上或笔记本计算机、PDA(个人数字助理)、平板式计算机(Tablet Computer)、口袋计算机、电话机、寻呼机、汽车、远程会议系统、用具或任何其他适当类型的电子装置。

网络 130 可以是任何适当的网络或网络组合，可支持适于向计算机系统 100 和服务器 132 传送数据/或代码、或者传送来自计算机系统 100 和服务器 132 的数据/或代码的任何适当协议。在各种实施例中，网络 130 可代表存储装置或存储装置的组合，其直接地或间接地连接到计算机系统 100。在实施例中，网络 130 可支持无限带宽 (Infiniband)。在另一实施例中，网络 130 可支持无线通信。在另一实施例中，网络可支持硬有线通信，比如电话线或线缆。在另一实施例中，网络 130 可支持以太网 IEEE(电气和电子工程师协会)802.3x 规范。在另一实施例中，网络 130 可以是互联网并且可支持 IP(互联网协议)。

在另一实施例中，网络 130 可以是局域网(LAN)或广域网(WAN)。在另一实施例中，网络 130 可以是热点服务提供者网络。在另一实施例中，网络 130 可以是内联网。在另一实施例中，网络 130 可以是 GPRS(通用分组无线服务)网络。在另一实施例中，网络 130 可以是 FRS(家庭无线服务)网络。在另一实施例中，网络 130 可以是任何适当的蜂窝数据网络或基于蜂窝的无线网络技术。在另一实施例中，网络 130 可以是 IEEE 802.11B 无线网络。在又一实施例中，网络 130 可以是任何适当的网络或网络组合。尽管示出了一个网络 130，但是在其他实施例中可存在任何数目(包括零)的网络(相同或不同类型)。

图 1 旨于在高层次描绘计算机系统 100 的代表性的主要组件、网络 130 和服务器 132；各个组件可比图 1 中表示的更为复杂；可存在图 1

中所示以外的或附加的组件；这些组件的数目、类型和配置可变化。这里公开了这些附加的复杂性或附加的变化的若干特定示例；应当理解，它们仅仅是作为示例，并且不必仅仅是这样的变化。

图1中所示的以及实现本发明各种实施例的各种软件组件可以以许多方式实现，包括使用各种计算机软件应用、例行程序、组件、程序、对象、模块、数据结构等，下文称为“计算机程序”或简称为“程序”。计算机程序一般包括一个或多个指令，这些指令在各个时刻驻留于计算机系统100中的各种存储器和存储装置中，并且在由计算机系统100中的一个或多个处理器101读取和执行时，使得计算机100执行用以执行包括本发明实施例各方面的步骤或元素所必需的步骤。

而且，尽管在完全功能计算机系统的环境中已经描述了并且下文将描述本发明的实施例，本发明的各种实施例能够以多种形式作为程序产品来发布，并且本发明可等地加以应用，而不管用以实际地实现该发布的信号承载介质的特定类型如何。定义该实施例的功能的程序可经由各种信号承载媒体交付给计算机系统100，其包括但不限于：

(1)永久地存储于不可改写的存储介质上的信息，该存储介质例如配属于计算机系统或计算机系统内的只读存储器装置，比如 CD-ROM、DVD-R 或 DVD+R；

(2)存储于可改写的存储介质上的可更改信息，该存储介质例如硬盘驱动器(例如 DASD 125、126 或 127)、CD-RW、DVD-RW、DVD+RW、DVD-RAM 或磁盘；或者

(3)由通信介质传送的信息，例如通过计算机或电话网络，比如网络130，包括无线通信。

这样的信号承载媒体在承载用以指示本发明的这些功能的机器可读指令时代表了本发明的实施例。

本发明的实施例还可作为与客户公司、非盈利组织、政府实体、内部组织结构等等的服务承诺的一部分来交付。这些实施例的方面可包括将计算机系统配置为执行这里所述方法的一部分或全部以及部署实现这里所述方法的一部分或全部的 web 服务和软件系统。这些实施例的方

面还可包括分析客户公司、根据该分析来形成推荐、生成用以实现该推荐的各部分的软件、将该软件集成到现有的处理和基础设施中、度量这里所述方法和系统的使用、向用户分配费用以及向用户针对这些方法和系统的使用进行计费。

此外，下面所述的各种程序可根据它们在本发明的具体实施例中实现时所针对的应用来识别。但是，仅为了方便而使用了随后的任何特定程序命名，因此本发明的实施例不应当限制为仅在由这样的命名所识别的和/或暗示的任何具体应用中使用。

图 1 中所示的示例性环境并不旨在限制本发明。实际上，可使用其他替代的硬件和/或软件环境，而不会脱离本发明的范围。

图 2 描绘了按照本发明实施例的示例存储器控制器 106 的框图。存储器控制器 106 包括读队列 205、写队列 210、冲突队列 215、滴流式 (trickle) 写要求寄存器 220、滴流式写周期寄存器 225 和滴流式写周期要求寄存器 230。滴流式写要求寄存器 220、滴流式写周期寄存器 225 和/或滴流式写周期要求寄存器 230 中的值可由存储器控制器 106 的设计者设置，或者存储器控制器 106 可从计算机 100 的用户或系统管理员、从计算机 100 的任何适当硬件和/或软件、或者经由网络 130 来接收它们。

读队列 205 存储从处理器 101 接收的、存储器控制器 106 尚未发送到存储器 102 的读命令。写队列 210 存储从处理器 101 接收的、存储器控制器 106 尚未发送到存储器 102 的写命令。冲突队列 215 存储存储器控制器 106 先前已经发送到存储器 102 的读和写命令二者，但是仅存储那些自从发送该命令时起、阈值时间量尚未届满的命令。冲突队列 215 进一步包括能够在处理器上执行的指令或逻辑，以执行如下面参照图 3、4 和 5 进一步所描述的功能。

滴流式写要求寄存器 220 存储处理器周期、时钟的周期数目或时间量阈值，冲突队列 215 将其与写队列 210 中的命令数目做比较。如果写队列 210 中的命令数目超过滴流式写要求寄存器 220 的值，则冲突队列 215 是在滴流式写要求模式，如下面参照图 4 进一步所描述的那样。

滴流式写周期寄存器 225 存储处理器周期、时钟的周期数目或时间

量阈值，冲突队列 215 将其与命令在冲突队列 215 中已经出现的处理器周期、时钟周期数目或时间量做比较。如果滴流式写周期寄存器 225 中的值超过该命令在冲突队列 215 中已经出现的周期数目或时间量，则冲突队列 215 将写队列 210 中最旧的不与冲突队列 215 相冲突的写命令发送到存储器 102，如下面参照图 4 进一步所描述的那样。如果命令和队列中的至少一个命令访问存储器 102 中的同一高速缓存线，则该命令与该队列相冲突。如果命令所访问的存储器 102 中的高速缓存线与队列中的所有命令所访问的存储器 102 中的高速缓存线均不相同，则该命令不与该队列相冲突。

滴流式写周期要求寄存器 230 存储处理器周期、时钟的周期数目或时间量阈值，冲突队列 215 将其与命令在冲突队列 215 中已经出现的处理器周期、时钟周期数目或时间量做比较。如果滴流式写周期要求寄存器 230 中的值超过该命令在冲突队列 215 中已经出现的周期数目或时间量，则冲突队列 215 将与冲突队列 215 不相冲突的写队列 210 中最旧的写命令发送到存储器 102，如下面参照图 5 进一步所描述的那样。

在各种其他实施例中，滴流式写要求寄存器 220、滴流式写周期寄存器 225 和/或滴流式写周期要求寄存器 230 可实现为存储器控制器 106 中的存储器位置。

图 3 描绘了按照本发明实施例的存储器控制器 106 的示例处理流程图。图 3 中所示的示例逻辑是在存储器控制器 106 的每个处理器周期或每个时钟周期执行的。控制始于方框 300。控制然后继续到方框 305，在其中冲突队列 215 确定写队列 210 中的写数目是否超过写队列阈值。如果方框 305 处的确定为真，则写队列 210 中的写数目超过写队列阈值，因此控制继续到方框 310，在其中冲突队列 215 将最旧的写命令从写队列 210 发送到存储器 102，并且将该写命令添加到冲突队列 215。控制然后继续到方框 398，在其中图 3 的逻辑返回。

如果方框 305 处的确定为假，则写队列 210 中的写数目未超过写队列阈值，因此控制从方框 305 继续到方框 315，在其中冲突队列 215 确定来自处理器 101 的输入读命令是否与写队列 210 中的任一命令相冲

突。如果输入读命令与写队列 210 中的命令想要访问存储器 102 中的同一高速缓存线，则两个命令相冲突。

如果方框 315 处的确定为真，则输入读命令与写队列 210 相冲突，因此控制继续到方框 320，在其中冲突队列 215 将与输入读命令相冲突的写命令从写队列 210 发送到存储器 102，并且将该写命令添加到冲突队列 215。控制然后继续到方框 398，在其中图 3 的逻辑返回。

如果方框 315 处的确定为假，则输入读命令不与写队列 210 中的写命令相冲突，因此控制从方框 315 继续到方框 325，其中冲突队列 215 确定读队列 205 中是否存在不与冲突队列 215 中的命令相冲突的读命令。

如果方框 325 处的确定为真，则读队列 205 中存在不与冲突队列 215 相冲突的读命令，因此控制继续到方框 330，在其中冲突队列 215 将读队列 205 中与写队列 210 中的所有写命令不相冲突的最旧的读命令发送到存储器 102，并且将该读命令添加到冲突队列 215。控制然后继续到方框 399，在其中图 3 的逻辑返回。

如果方框 325 处的确定为假，则读队列 205 中的所有读命令与冲突队列 215 中的至少一个命令相冲突，因此控制继续到方框 335，在其中执行图 4 的逻辑，其如下面进一步所述。控制然后继续到方框 399，在其中图 3 的逻辑返回。

图 4 描绘了按照本发明实施例的存储器控制器 106 的又一示例处理流程图。控制始于方框 400。控制然后继续到方框 405，在其中冲突队列 215 确定写队列 210 中的命令数目是否大于滴流式写要求寄存器 220 中的值。如果方框 405 处的确定为真，则写队列 210 中的命令数目大于滴流式写要求寄存器 220 中的值，因此控制继续到方框 410，在其中执行图 5 的逻辑，其如下面进一步所述。控制然后继续到方框 499，在其中图 4 的逻辑返回。

如果方框 405 处的确定为假，则写队列 210 中的命令数目不大于滴流式写要求寄存器 220 中的值，因此控制从方框 405 继续到方框 415，在其中冲突队列 215 对于冲突队列 215 中的所有命令来确定滴流式写周

期寄存器 225 中的值是否大于所有命令在冲突队列 215 中已经出现的处理器周期、时钟周期数目或时间量的定时器值。在实施例中，冲突队列 215 中存储的所有命令具有不同的关联定时器，该定时器对该命令在冲突队列 215 中已经出现的时间量或周期数目保持跟踪。

如果方框 415 处的确定为真，则滴流式写周期寄存器 225 中的值大于所有命令在冲突队列 215 中已经出现的周期数目或时间量，因此控制继续到方框 420，在其中冲突队列 215 将写队列 210 中不与冲突队列 215 相冲突的最旧的写命令(已经出现最久的写命令)发送到存储器 102，并且将该写命令添加到冲突队列 215。控制然后继续到方框 499，在其中图 4 的逻辑返回。

如果方框 415 处的确定为假，则滴流式写周期寄存器 225 中的值不大于所有命令在冲突队列 215 中已经出现的周期数目或时间量，因此控制从方框 415 继续到方框 499，在其中图 4 的逻辑返回。

图 5 描绘了按照本发明实施例的存储器控制器 106 的又一示例处理流程图。控制始于方框 500。控制然后继续到方框 505，在其中冲突队列 215 对于冲突队列 215 中的所有命令来确定滴流式写周期要求寄存器 230 中的值是否超过该命令在冲突队列 215 中已经出现的处理器周期、时钟周期数目或时间量的定时器值。

如果方框 505 处的确定为真，则滴流式写周期要求寄存器 230 中的值超过所有命令在冲突队列 215 中已经出现的周期数目，因此控制继续到方框 510，在其中冲突队列 215 将写队列 210 中不与冲突队列 215 相冲突的最旧的写命令发送到存储器 102，并且将该写命令添加到冲突队列 215。控制然后继续到方框 599，在其中图 5 的逻辑返回。

如果方框 505 处的确定为假，则滴流式写周期要求寄存器 230 中的值不超过所有命令在冲突队列 215 中已经出现的周期数目，因此控制继续到方框 599，在其中图 5 的逻辑返回。

以此方式，写命令对于读命令的性能的影响得以减小，在实施例中，如果写队列 210 几乎充满，写命令才从写队列 210 发送到存储器 102，并且如果不采取动作，则可能发生总线 103 的拖延。

在本发明示范实施例的先前详细描述中，参考了形成本说明书一部分的附图(其中相似的编号代表相似的元素)，在附图中作为例子示出了可实施本发明的具体示例性实施例。充分详细地描述了这些实施例，以使得本领域的技术人员能够实施本发明，但是不脱离本发明的范围，可利用其他实施例，以及可进行逻辑上的、机械上的、电气上的和其他方面的变化。在本说明书中使用的措辞“实施例”的不同实例不必指代同一实施例，但是它们可以指代同一个实施例。因此，先前的详细描述不能以限制性的意义来理解，本发明的范围仅由所附权利要求书来限定。

在先前的描述中，给出了大量具体的细节，以提供本发明实施例的透彻理解。但是本发明可在没有这些具体细节时实施。在其他实例中，没有具体地示出公知的电路、结构和技术，以免使得本发明难以理解。

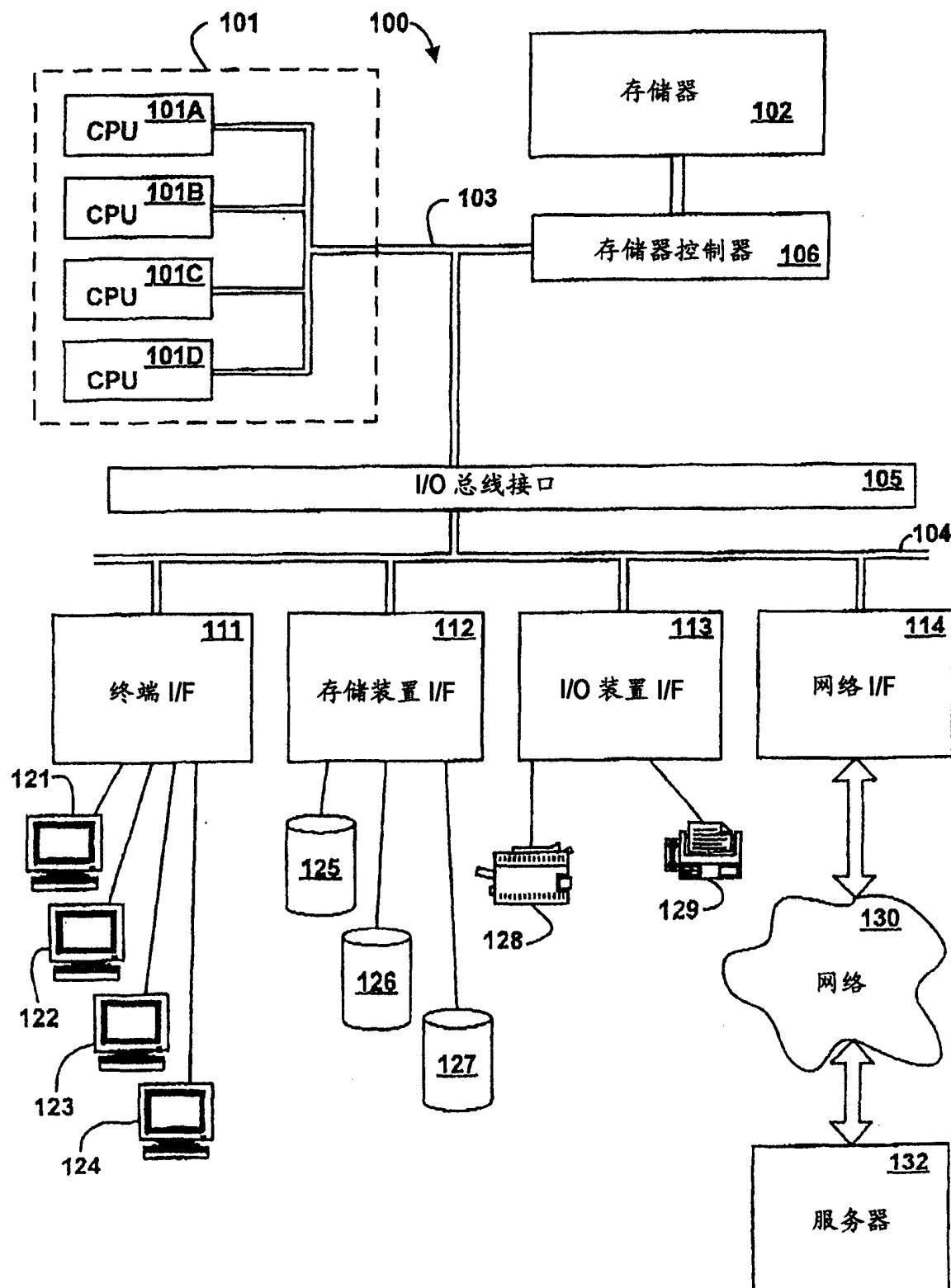


图 1

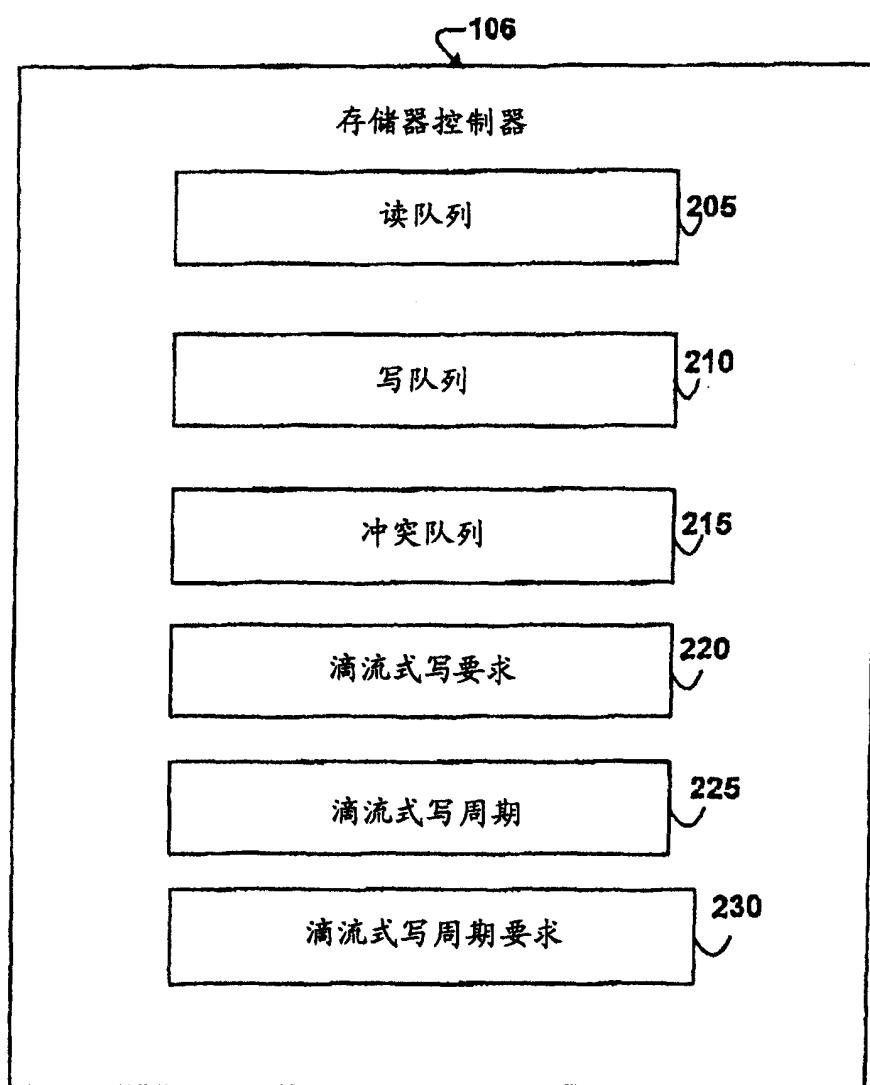


图 2

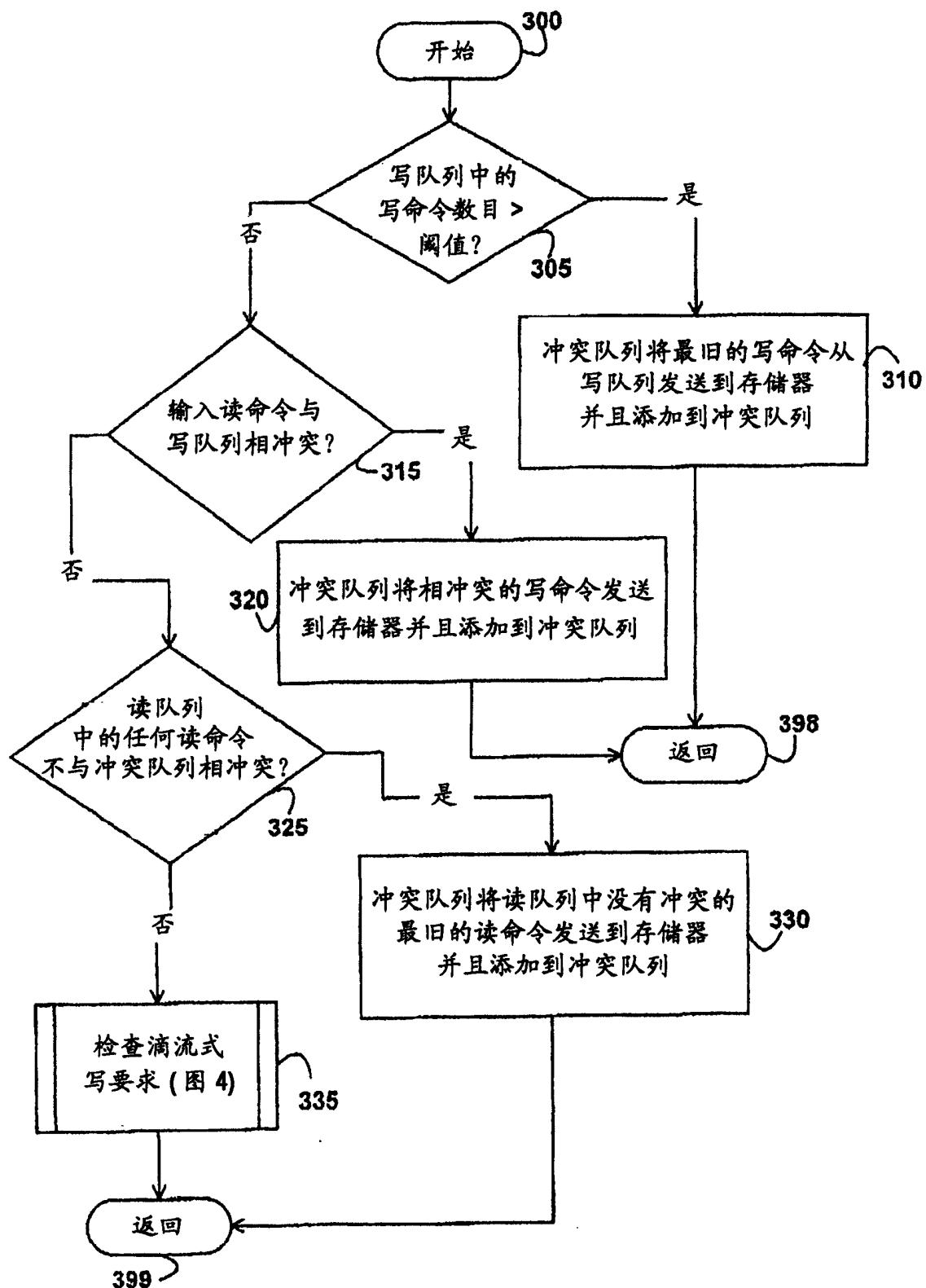


图 3

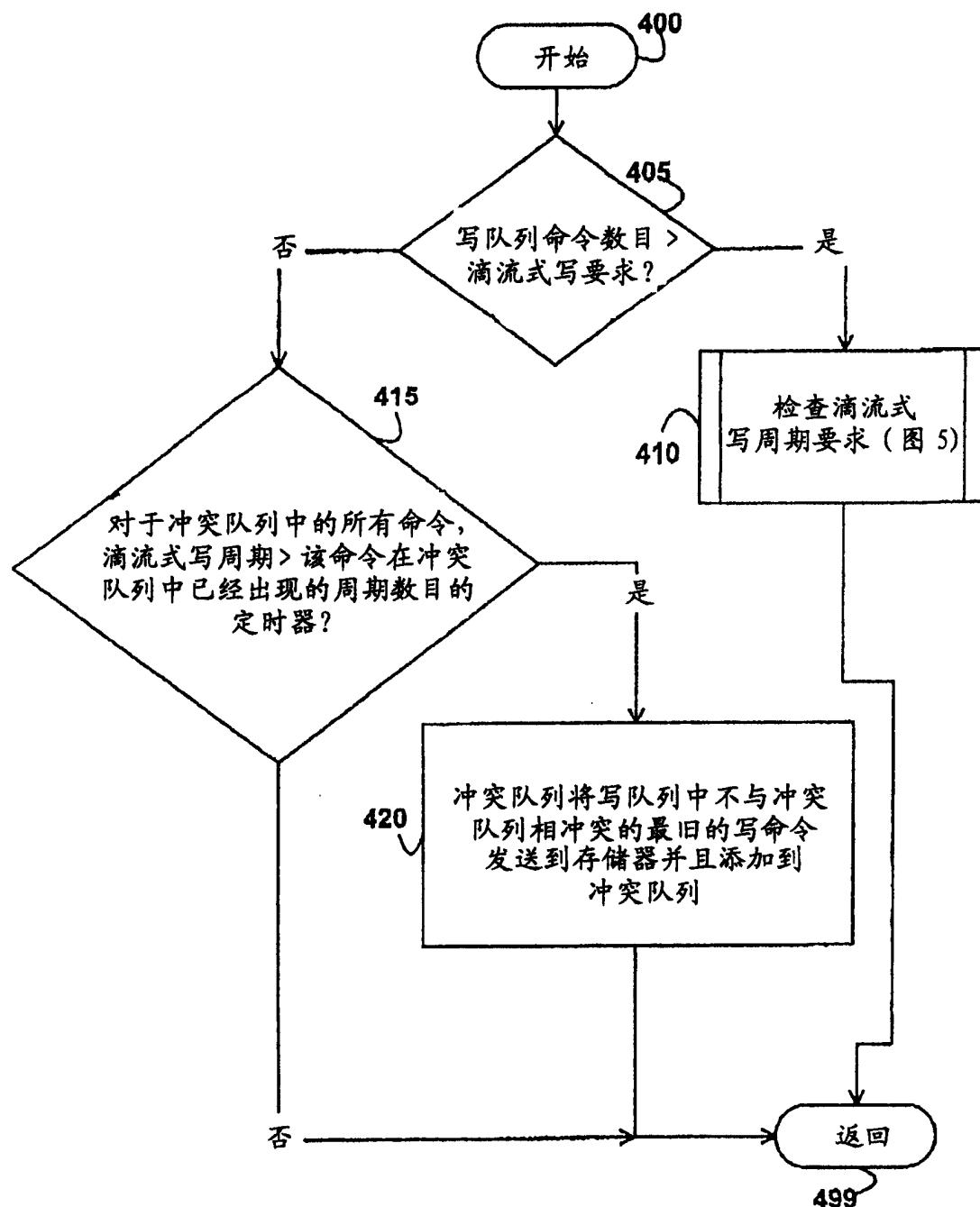


图 4

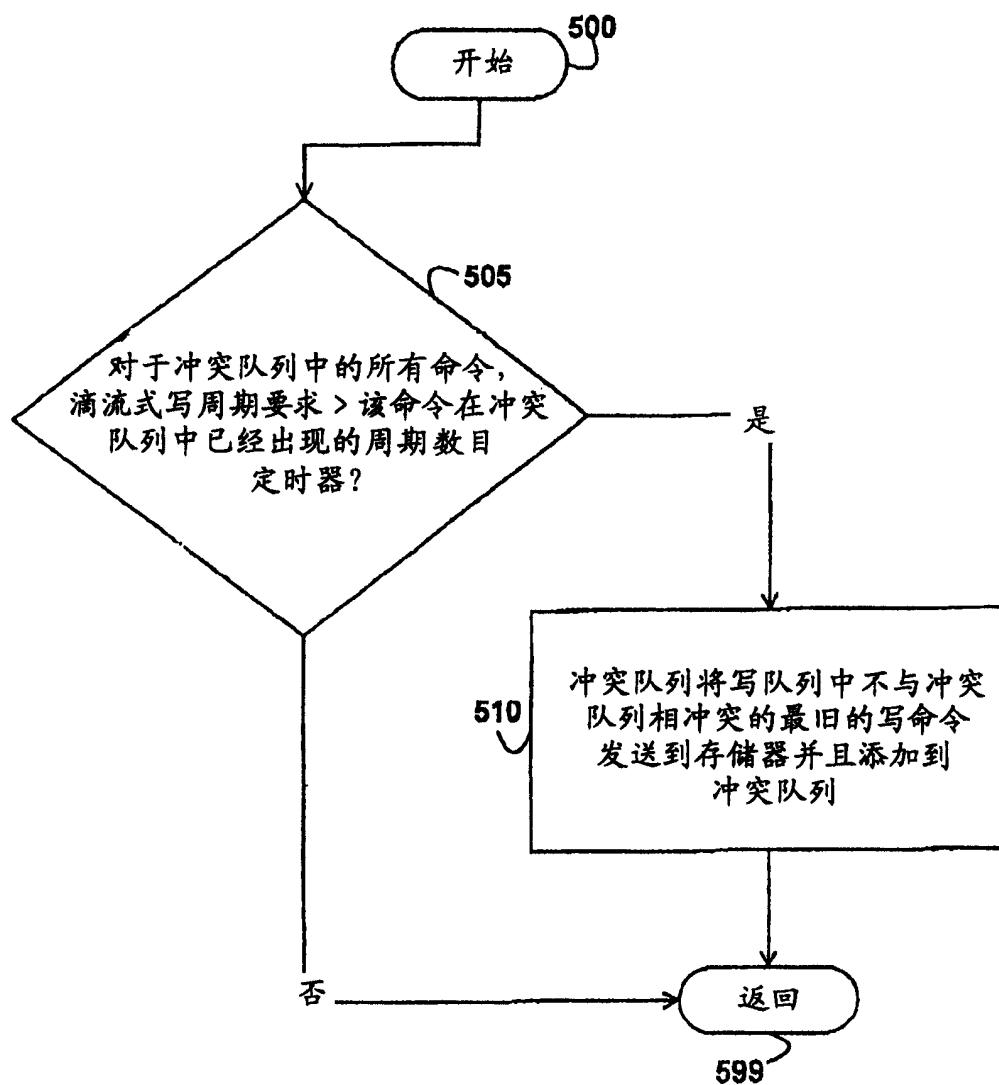


图 5