(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 113366457 B
(45)授权公告日 2024.06.14

(21)申请号 202080011704.2
(22)申请日 2020.01.16
(65)同一申请的已公布的文献号
申请公布号 CN 113366457 A
(43)申请公布日 2021.09.07
(30)优先权数据
19154740.5 2019.01.31 EP
(85)PCT国际申请进入国家阶段日
2021.07.28
(86)PCT国际申请的申请数据
PCT/IB2020/050339 2020.01.16
(87)PCT国际申请的公布数据
WO2020/157594 EN 2020.08.06
(73)专利权人 国际商业机器公司
地址 美国纽约
(72)发明人 C·莱施 M·克雷默 F·莱纳特
M·克莱纳 J·布拉德伯里
C·雅各比 B·贝尔马
P·德里费尔
(74)专利代理机构 北京市中咨律师事务所 11247
专利代理师 于静 刘薇
(51)Int.Cl.
G06F 13/10(2006.01)
(56)对比文件
CN 101126995 A,2008.02.20
CN 104572517 A,2015.04.29
审查员 李妍

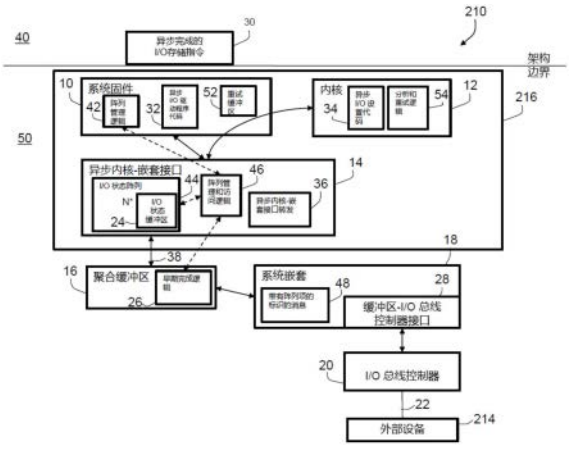
权利要求书5页 说明书16页 附图5页

(54)发明名称

处理输入/输出存储指令

(57)摘要

用于处理输入/输出存储指令(30)的数据处理系统(210)和方法,包括通过输入/输出总线控制器(20)耦合到至少一个输入/输出总线(22)的系统嵌套(18)。数据处理单元(216)经由聚合缓冲区(16)耦合到系统嵌套(18)。系统嵌套(18)被配置以异步地从至少一个外部设备(214)加载数据和/或将数据存储到该至少一个外部设备。数据处理单元(216)被配置以在系统嵌套(18)中输入/输出存储指令(30)的执行完成之前完成输入/输出存储指令(30)。异步内核-嵌套接口(14)包括具有多个输入/输出状态缓冲区(24)的输入/输出状态阵列(44)。系统固件(10)包括重试缓冲区(52),内核(12)包括分析和重试逻辑(54)。



CN 113366457 B

1.一种用于处理输入/输出存储指令(30)的数据处理系统(210),包括:

通过输入/输出总线控制器(20)通信地耦合到至少一个输入/输出总线(22)的系统嵌套(18),

还至少包括

数据处理单元(216),其包括内核(12)、系统固件(10)和异步内核-嵌套接口(14),

其中,所述数据处理单元(216)经由聚合缓冲区(16)通信地耦合到所述系统嵌套(18),

其中,所述系统嵌套(18)被配置以异步地从通信地耦合到所述输入/输出总线(22)的至少一个外部设备(214)加载数据和/或将数据存储到所述至少一个外部设备(214),

其中,所述异步内核-嵌套接口(14)包括具有多个输入/输出状态缓冲区(24)的输入/输出状态阵列(44)和阵列管理和访问逻辑(46),

其中,所述系统固件(10)包括重试缓冲区(52),所述内核(12)包括分析和重试逻辑(54),

并且其中

(i)在所述数据处理系统(210)上运行的操作系统,发布所述输入/输出存储指令(30),其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数;

(ii)所述数据处理单元(216)被配置以通过在所述输入/输出存储指令(30)中指定的所述地址来识别所述输入/输出函数;

(iii)所述数据处理单元(216)被配置以验证在地址空间上和在客户机实例级别上是否允许访问所述输入/输出函数,所述客户机在所述数据处理系统(210)上运行;

(iv)所述数据处理单元(216)被配置以在所述系统嵌套(18)中所述输入/输出存储指令(30)的执行完成之前完成所述输入/输出存储指令(30);(v)所述系统固件(10)被配置得如果在所述输入/输出存储指令(30)的异步执行期间由所述数据处理单元(216)检测到错误,则通过中断来通知所述操作系统;

(vi)所述分析和重试逻辑(54)分别检测错误,硬件确保所述存储指令(30)尚未被转发到输入/输出总线(22);

(vii)所述重试缓冲区(52)保存存储信息,用于在系统硬件/固件(50)中执行存储指令(30)的重试;

(viii)所述分析和重试逻辑(54)分析错误并检查重试可能性;

(ix)所述分析和重试逻辑(54)触发重试。

2.根据权利要求1所述的数据处理系统,其中,所述分析和重试逻辑(54)计算重试错误的数量,检查错误检测的阈值,并向所述操作系统报告失败的重试次数。

3.根据权利要求1或2所述的数据处理系统,所述聚合缓冲区(16)通过异步总线(38)通信地耦合到所述异步内核-嵌套接口(14)。

4.根据权利要求1或2所述的数据处理系统,其中,如果数据的长度超过8字节,数据可以由所述输入/输出存储指令(30)通过具有早期完成消息的异步传输机制在多个数据包中传输到所述聚合缓冲区(16),否则,在一个数据包中传输数据。

5.根据权利要求1或2所述的数据处理系统,所述系统固件(10)包括用于处理所述输入/输出存储指令(30)的异步输入/输出驱动程序代码(32)。

6.根据权利要求5所述的数据处理系统,所述内核(12)包括异步设置代码(34),用于处理对所述异步输入/输出驱动程序代码(32)的状态信息的存储器要求。

7.根据权利要求1或2所述的数据处理系统,所述异步内核-嵌套接口(14)包括异步内核-嵌套接口转发组件(36),用于转发本地完成的数据。

8.根据权利要求1或2所述的数据处理系统,所述聚合缓冲区(16)包括早期完成逻辑(26),用于在发送请求之后传送空闲供重用消息。

9.根据权利要求1或2所述的数据处理系统,其中所述系统固件(10)包括阵列管理逻辑(42),该阵列管理逻辑(42)分配/释放所述输入/输出状态阵列(44)中的输入/输出状态缓冲区(24)和/或发起新的存储指令(30)的开始。

10.根据权利要求5所述的数据处理系统,其中所述异步输入/输出驱动程序代码(32)将每个输入/输出操作的副本存储在重试缓冲区(52)中,并且,如果所述异步输入/输出驱动程序代码(32)检测到错误,则将控制转移到所述分析和重试逻辑(54)。

11.根据权利要求1或2所述的数据处理系统,其中,所述分析和重试逻辑(54)

-确定失败的存储指令(30),

-确定该存储指令(30)的输入/输出函数,

-如果低于阈值,则发起对可重试错误的重试,

-如果在异步内核-嵌套接口(14)中发生错误并且超过阈值,则删除所有对该输入/输出函数的未完成请求;将该输入/输出函数设置为错误状态,并向操作系统发送异步错误信号,

--如果发生严重错误,则确定错误源;如果错误源是系统嵌套(18),则全部删除,将所有涉及的输入/输出函数设置为错误状态,并向操作系统发出异步错误信号;如果错误源是输入/输出总线控制器(20),则删除对连接到该输入/输出总线控制器(20)的输入/输出函数的所有未完成请求;将所有输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。

12.根据权利要求1或2所述的数据处理系统,所述系统固件(10)向所述聚合缓冲区(16)发布系统消息以将数据作为单个嵌套消息异步地转发到所述输入/输出总线控制器(20),同时等待所述聚合缓冲区(16)发送完成消息,其中系统消息包括:

-分层物理目标地址,

-提供SMT线程或聚合缓冲区标识符,

-数据的长度,

-输入/输出总线地址,

-输入/输出状态缓冲区索引。

13.一种用于处理针对数据处理系统(210)的至少一个外部设备(214)的输入/输出存储指令(30)的方法,所述数据处理系统(210)包括:

系统嵌套(18),通过输入/输出总线控制器(20)通信地耦合到至少一个输入/输出总线(22),

并且进一步至少包括数据处理单元(216),所述数据处理单元(216)包括内核(12)、系统固件(10)和异步内核-嵌套接口(14),

其中所述数据处理单元(216)经由聚合缓冲区(16)通信地耦合到所述系统嵌套(18),

其中,所述外部设备(214)通信地耦合到所述输入/输出总线(22),其中,所述异步内核-嵌套接口(14)包括具有多个输入/输出状态缓冲区(24)的输入/输出状态阵列(44)、阵列管理和访问逻辑(46),

其中,所述系统固件(10)包括重试缓冲区(52),所述内核(12)包括分析和重试逻辑(54),

该方法包括:

(i)在所述数据处理系统(210)上运行的操作系统,发布所述输入/输出存储指令(30),其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数;

(ii)所述数据处理单元(216)被配置以通过在所述输入/输出存储指令(30)中指定的所述地址来识别所述输入/输出函数;

(iii)所述数据处理单元(216)被配置以验证在地址空间上和在客户机实例级别上是否允许访问所述输入/输出函数,所述客户机在所述数据处理系统(210)上运行;

(iv)所述数据处理单元(216)被配置以在所述系统嵌套(18)中所述输入/输出存储指令(30)的执行完成之前完成所述输入/输出存储指令(30);

(v)所述系统固件(10)被配置得如果在所述输入/输出存储指令(30)的异步执行期间由所述数据处理单元(216)检测到错误,则通过中断来通知所述操作系统;

(vi)所述分析和重试逻辑(54)分别检测错误,硬件确保所述存储指令(30)尚未被转发到输入/输出总线(22);

(vii)所述重试缓冲区(52)保存存储信息,用于在系统硬件/固件(50)中执行存储指令(30)的重试;

(viii)所述分析和重试逻辑(54)分析错误并检查重试可能性;

(ix)所述分析和重试逻辑(54)触发重试。

14.根据权利要求13所述的方法,其中,所述分析和重试逻辑(54)计算重试错误的数量,检查错误检测的阈值,并向所述操作系统报告失败的重试次数。

15.根据权利要求13或14所述的方法,进一步包括:

(i)所述操作系统发布所述输入/输出存储指令(30);

(ii)系统固件(10)分配空闲输入/输出状态缓冲区索引,如果没有空闲输入/输出状态缓冲区索引可用,则等待空闲输入/输出状态缓冲区索引;

(iii)所述系统固件(10)将所述存储指令(30)注入到异步发送引擎中;

如果被另一个存储指令阻塞,等待直到该存储指令完成;

(iv)取决于数据的长度:如果数据的长度超过八个字节,所述系统固件(10)重复发布系统消息以将数据包发送至所述聚合缓冲区(16),直到存储块的所有数据已经被转发至所述聚合缓冲区(16),同时系统固件(10)等待直到数据已经由系统消息发送;否则

所述系统固件(10)发布系统消息以将所述数据发送到所述聚合缓冲区(16);

(v)所述系统固件(10)向所述聚合缓冲区(16)发布系统消息以将所述数据作为单个嵌套消息异步地转发到所述输入/输出总线控制器(20),同时等待所述聚合缓冲区(16)发送完成消息;

(vi)所述聚合缓冲区(16)将所述嵌套消息注入到所述系统嵌套(18)中,其中,所述聚

合缓冲区(16)刚好在所述步骤(v)的发送操作之后空闲以用于重新使用,发信号回所述系统固件(10);然后聚合缓冲区(16)发送空闲供重用消息;

(vii)所述系统嵌套(18)将所述消息转发到目标位置;

(viii)所述输入/输出总线控制器(20)接收所述消息并且将数据帧中的数据转发到所述输入/输出总线;

(ix)所述输入/输出总线控制器(20)向所述系统嵌套(18)发送完成消息;

(x)所述系统嵌套(18)将所述完成消息转发到所述聚合缓冲区(16);

(xi)所述聚合缓冲区(16)将完成转发至所述异步内核-嵌套接口(14);(xii)所述异步内核-嵌套接口(14)将所述完成状态存储在所述输入/输出状态缓冲区(24)中用于所述输入/输出状态缓冲区索引,并且将操作的完成发信号通知给所述系统固件(10);

(xiii)所述系统固件(10)通过所述输入/输出状态缓冲区索引来更新输入/输出状态缓冲区跟踪;

(xiv)在错误的情况下,所述系统固件(10)将缺陷异步地发信号通知给所述操作系统。

16.根据权利要求13或14所述的方法,进一步,如果数据的长度超过8字节,则由输入/输出存储指令(30)通过具有早期完成消息的异步传输机制在多个数据包中将数据传输到聚合缓冲区(16)。

17.根据权利要求13或14所述的方法,进一步地,所述系统固件(10)使用用于处理所述输入/输出存储指令(30)的异步输入/输出驱动程序代码(32)。

18.根据权利要求17所述的方法,进一步,所述内核(12)使用异步设置代码(34),用于处理对所述异步输入/输出驱动程序代码(32)的状态信息的存储器要求。

19.根据权利要求13或14所述的方法,进一步,所述异步内核-嵌套接口(14)使用异步内核-嵌套接口转发组件(36)来转发本地完成的数据。

20.根据权利要求13或14所述的方法,进一步地,所述聚合缓冲区(16)使用早期完成逻辑(26)用于在发送请求之后传送空闲供重用消息。

21.根据权利要求15所述的方法,其中,系统消息包括以下之一:

-分层物理目标地址,

-提供SMT线程或聚合缓冲区标识符,

-数据的长度,

-输入/输出总线地址,

-输入/输出状态缓冲区索引。

22.根据权利要求17所述的方法,其中所述异步输入/输出驱动程序代码(32)将每个输入/输出操作的副本存储在重试缓冲区(52)中,并且如果所述异步输入/输出驱动程序代码(32)检测到错误,则将控制转移到所述分析和重试逻辑(54)。

23.根据权利要求13或14所述的方法,所述分析和重试逻辑(54)

-确定失败的存储指令(30),

-确定该存储指令(30)的输入/输出函数,

-如果低于阈值,则发起对可重试错误的重试,

-如果在异步内核-嵌套接口(14)中发生错误并且超过阈值,则删除所有对该输入/输出函数的未完成请求;将该输入/输出函数设置为错误状态,并向操作系统发送异步错误信

号，

-如果发生严重错误，则确定错误源；如果错误源是系统嵌套(18)，则全部删除，将所有涉及的输入/输出函数设置为错误状态，并向操作系统发送异步错误信号；如果错误源是输入/输出总线控制器(20)，则删除对连接到该输入/输出总线控制器(20)的输入/输出函数的所有未完成请求；将所有输入/输出函数设置为错误状态，并向操作系统发送异步错误信号。

24.一种计算机程序产品，用于处理针对数据处理系统(210)的至少一个外部设备(214)的输入/输出存储指令(30)，所述数据处理系统(210)包括：系统嵌套(18)，通过输入/输出总线控制器(20)通信地耦合到至少一个输入/输出总线(22)，

并且，还至少包括数据处理单元(216)，所述数据处理单元(216)包括内核(12)、系统固件(10)和异步内核-嵌套接口(14)，

其中所述数据处理单元(216)经由聚合缓冲区(16)通信地耦合到所述系统嵌套(18)，

其中，所述外部设备(214)通信地耦合到所述输入/输出总线(22)，

其中所述异步内核-嵌套接口(14)包括具有多个输入/输出状态缓冲区(24)的输入/输出状态阵列(44)、阵列管理和访问逻辑(46)，

其中，所述系统固件(10)包括重试缓冲区(52)，所述内核(12)包括分析和重试逻辑(54)，

所述计算机程序产品包括具有体现于其中的程序指令的计算机可读存储介质，所述程序指令可由计算机系统(212)执行以致使所述计算机系统(212)执行一种方法，所述方法包括：

(i)在所述数据处理系统(210)上运行的操作系统，发布所述输入/输出存储指令(30)，其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数；

(ii)所述数据处理单元(216)被配置以通过在所述输入/输出存储指令(30)中指定的所述地址来识别所述输入/输出函数；

(iii)所述数据处理单元(216)被配置以验证在地址空间上和在客户机实例级别上是否允许访问所述输入/输出函数，所述客户机在所述数据处理系统(210)上运行；

(iv)所述数据处理单元(216)被配置以在所述系统嵌套(18)中所述输入/输出存储指令(30)的执行完成之前完成所述输入/输出存储指令(30)；(v)所述系统固件(10)被配置得如果在所述输入/输出存储指令(30)的异步执行期间由所述数据处理单元(216)检测到错误，则通过中断来通知所述操作系统；

(vi)所述分析和重试逻辑(54)分别检测错误，硬件确保所述存储指令(30)尚未被转发到输入/输出总线(22)；

(vii)所述重试缓冲区(52)保存存储信息，用于在系统硬件/固件(50)中执行存储指令(30)的重试；

(viii)所述分析和重试逻辑(54)分析错误并检查重试可能性；

(ix)所述分析和重试逻辑(54)触发重试。

25.一种用于执行数据处理程序(240)的数据处理系统(210)，所述数据处理系统包括用于执行根据权利要求13至23中任一项所述的方法的计算机可读程序指令。

处理输入/输出存储指令

技术领域

[0001] 本发明总体涉及数据处理系统,具体涉及用于处理到多个外部设备的输入/输出存储指令的方法以及计算机程序产品和数据处理系统。

背景技术

[0002] 计算环境可包括一个或多个类型的输入/输出设备,包括不同类型的适配器。一种类型的适配器是外围组件互连(PCI)或快速外围组件互连(PCIe)适配器。该适配器包括一个或多个地址空间,用于在适配器与附接到适配器的系统之间传送数据。

[0003] 在一些系统中,耦合到适配器的中央处理单元(CPU)的地址空间的一部分被映射到适配器的地址空间,使得访问存储器的CPU指令能够直接操纵适配器的地址空间中的数据。

[0004] 与适配器(例如PCI或PCIe适配器)的通信可以通过专门设计用于向和从适配器传送数据并用于通信的控制指令来促进。

[0005] 在现有技术中,用于在适配器中存储数据的存储指令包括例如获得用于执行的机器指令,该机器指令是按照计算机架构定义用于计算机执行的,该机器指令包括例如标识存储到适配器(store to adapter)指令的操作码字段。第一字段标识包括将被存储在适配器中的数据的第一位置。第二字段标识第二位置,其内容包括标识适配器的函数句柄(function handle)、适配器内将存储数据的地址空间的名称、以及该地址空间内的偏址(offset)。该机器指令被执行,该执行包括使用函数句柄来获得与适配器相关联的函数表项(function table entry)。利用函数表项中的信息和偏址的至少之一来获取适配器的数据地址。将数据从第一位置存储在由地址空间的名称标识的地址空间中的特定位置中,该特定位置由适配器的数据地址标识。

[0006] 大型多处理器系统中的现有特征是使目标区域内的所有处理器静默(quiesce)的能力。静默功能的作用是临时停止或改变处理器或处理器组的状态,以执行例如系统更新或备份。在一些实例中,静默中断仅适用于系统资源的子集。在这样的情况下,该系统可以被分成不同的区。对于适用于一个区域(目标区域)的静默操作,允许在目标区域之外的处理器继续运行,尽管新的转换可能被阻止。通常,至少一个系统控制器或其他机制向系统中的所有物理处理器广播静默,处理收集静默状态信息,并且在所有处理器已经启动时向请求处理器指示,或者忽略(过滤掉)静默请求。

[0007] 静默控制器可以通信地耦合到多处理器系统中的处理器,以及被配置以接收静默请求的静默状态机。该计算机系统被配置以执行一种方法,该方法包括:在静默控制器处从请求处理器接收静默请求,请求处理器是多处理器系统中的多个处理器之一;以及基于静默状态机的状态确定静默请求不被接受。该方法还包括基于该请求未被接受,生成被配置以指示静默请求已被拒绝的拒绝消息;保持拒绝消息直到静默命令被广播到多处理器系统,该静默命令基于不同的静默请求;以及基于静默控制器检测到该静默命令的广播,向请求处理器发送拒绝消息。

发明内容

[0008] 提出了一种用于处理输入/输出存储指令的数据处理系统,其包括通过输入/输出总线控制器通信地耦合到至少一个输入/输出总线的系统嵌套。数据处理系统还至少包括数据处理单元,其包括内核、系统固件和异步内核-嵌套接口。数据处理单元经由聚合缓冲区通信地耦合到系统嵌套。系统嵌套被配置以异步地从通信地耦合到输入/输出总线的至少一个外部设备加载数据和/或将数据存储到该至少一个外部设备。异步内核-嵌套接口包括具有多个输入/输出状态缓冲区的输入/输出状态阵列、以及阵列管理和访问逻辑。系统固件还包括重试缓冲区,内核包括分析和重试逻辑。

[0009] 该数据处理系统被配置以执行:(i)在数据处理系统上运行的操作系统发布输入/输出存储指令,其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数;(ii)数据处理单元被配置以通过在输入/输出存储指令中指定的地址来识别输入/输出函数;(iii)数据处理单元被配置以验证是否允许在地址空间上和在客户机实例级别上对该输入/输出函数的访问,该客户机在数据处理系统上运行;(iv)数据处理单元被配置以在系统嵌套中输入/输出存储指令的执行完成之前完成输入/输出存储指令;(v)系统固件被配置得如果在输入/输出存储指令的异步执行期间由数据处理单元检测到错误,则通过中断、发送失败的异步执行的数据,来通知操作系统;(vi)分析和重试逻辑分别检测错误,由硬件确保存储指令尚未转发到输入/输出总线;(vii)重试缓冲区保存存储信息,用于在系统硬件/固件中执行存储指令的重试;(viii)分析和重试逻辑分析错误,检查重试可能性;以及(ix)分析和重试逻辑触发重试。

[0010] 有利地,异步——但与指令执行有关的——用于存储指令(store instruction)和屏障指令(barrier instruction)的错误检测和恢复设备,被添加到数据处理系统中。数据处理系统中的所有相关技术都可以同步地存储相关的用于存储指令的错误检测和恢复。其他错误,例如链路丢失(link drop)、由输入/输出设备报告的错误,都是独立于存储指令执行而检测和处理的。

[0011] 根据本发明实施例,引入可重试错误(retryable errors),其中在架构边界下的硬件和固件内进行重试。

[0012] 根据本发明的第一实施例的数据处理系统包括经由输入/输出总线从数据处理系统的至少一个外部设备加载和存储到数据处理系统的至少一个外部设备的指令。异步指令在数据已被存储到外部设备之前完成,而同步指令在数据已被存储到外部设备之后完成。在这里描述的实施例内,PCI将可互换地用于任何其他输入/输出技术,因此不将本发明的实施例限制于PCI。

[0013] 本发明的实施例描述了以从架构边界上可观察到的严格有序方式的输入/输出存储指令执行,而在数据处理单元(CPU)的硬件内实际的执行可能是无序的。

[0014] 根据本发明的实施例,可以通过PCIe存储效果的异步执行和异步状态处理来执行PCI存储指令。异步可靠执行是基于本发明数据处理系统的微架构中的可靠转发机制。

[0015] 现有的PCI存储和存储块指令通常是同步的,直到PCI存储数据已经被传送到PCIe接口且完成被返回到处理单元。

[0016] PCI标准只要求PCI信息的异步发送命令,这通常是通过处理器中的存储队列聚合异步发送的数据来实现。

[0017] 有利地,根据本发明的实施例,可以通过用输入/输出存储指令的可靠异步发送处理替换同步PCI指令来实现关于每个指令的周期数的改进。

[0018] 作为要传送的数据的替换或补充,根据本发明的实施例的存储指令还可以指定指向主存储器的指针,该指针应当用于从其获取数据,而不是直接包含该数据。

[0019] 客户机实例级别还可意味着单个客户机或主机可在数据处理系统上运行。

[0020] 输入/输出函数本身的偏址的地址可以是虚拟、物理、逻辑地址。虚拟和逻辑地址通常经由存储器管理单元(MMU)被转换成物理地址,然后可以通过物理地址来识别所指的是哪个函数和偏址。

[0021] 本文中的物理地址是指“可从客户机/操作系统内访问的地址转换层次结构中的最低地址”。

[0022] 有利地,输入/输出状态缓冲区可从系统嵌套和/或从输入/输出总线控制器收集返回状态,特别是从系统嵌套收集完成消息。这些输入/输出状态缓冲区可收集返回状态,由此充当支持异步传送过程的异步系统消息缓冲区。有利地,为了快速响应,可以将输入/输出状态缓冲区直接集成在异步内核-嵌套接口中。

[0023] 根据本发明的数据处理系统的有利实施例,分析和重试逻辑计算重试错误的数量,检查错误检测的阈值,并向操作系统报告失败的重试次数。因此,可以有利地进行对所发生的可重试错误的确定。

[0024] 有利地,输入/输出状态缓冲区可以从系统嵌套和/或从输入/输出总线控制器收集消息状态,特别是从系统嵌套收集完成状态。通过这种方式,可以以有序且高效的方式处理关于不同存储指令的完成状态的信息。

[0025] 有利地,消息状态和/或完成状态可通过输入/输出状态缓冲区索引来编号。编号使得能够有可能以有序且有效的方式处理消息、特别是完成状态,以进一步处理其他存储指令。

[0026] 根据本发明的数据处理系统的有利实施例,聚合缓冲区可经由异步总线通信地耦合到异步内核-嵌套接口。由此,聚合缓冲区可以连续地处理由异步内核-嵌套接口直接发送的数据,直到所有要传送到外部设备的数据都被存储在聚合缓冲区中。通过这种方式,可有利地支持用于从异步内核-嵌套接口传输数据的异步传输机制。

[0027] 根据本发明的数据处理系统的有利实施例,如果源数据的长度超过8字节,数据可以由输入/输出存储指令通过具有早期完成消息的异步传输机制在多个数据包中传输到聚合缓冲区,否则,可以在一个数据包中传输数据。该异步传输机制是有利的,因为发送设备在较早的状态时空闲供重用(free for reuse)。

[0028] 根据本发明的数据处理系统的有利实施例,系统固件可包括用于处理输入/输出存储指令的异步输入/输出驱动程序代码。由此,异步传送机制可被用于将数据从数据处理单元传送到外部设备。

[0029] 根据本发明的数据处理系统的有利实施例,内核可包括用于处理对异步输入/输出驱动程序代码的状态信息的存储器要求的异步设置代码。该异步设置代码可进一步促进通过聚合缓冲区到系统嵌套和输入/输出总线控制器的异步传输机制。

[0030] 根据本发明数据处理系统的一个有利实施例,异步内核-嵌套接口可包括用于转发本地完成的数据的异步内核-嵌套接口转发组件。此组件可在异步内核-嵌套接口中的硬

件中实施。因此,可以支持用于将数据在数据包中发送到聚合缓冲区的有利的异步传输模式。

[0031] 根据本发明的数据处理系统的有利实施例,聚合缓冲区可以包括用于在发送请求之后传送空闲供重用消息的早期完成逻辑。这使得能够早期继续经由聚合缓冲区向系统嵌套和输入/输出总线控制器传输数据的处理。

[0032] 根据本发明的数据处理系统的有利实施例,系统固件可以包括阵列管理逻辑,其分配/释放输入/输出状态阵列中的输入/输出状态缓冲区和/或发起新存储指令的开始。因此,空闲状态缓冲区可归因于进一步的存储指令。可以以高效省时的方式进行对存储指令的有序处理。

[0033] 根据本发明数据处理系统的有利实施例,异步输入/输出驱动程序代码可将每个输入/输出操作的副本存储在重试缓冲区中,并且,如果异步输入/输出驱动程序代码检测到错误,则可以将控制转移到分析和重试逻辑。通过这种方式,可以有利地实现系统恢复,其中可以仅处理实际错误。

[0034] 根据本发明数据处理系统的有利实施例,分析和重试逻辑可以确定失败的存储指令。此外,它可以确定该存储指令的输入/输出函数,如果低于阈值,则发起对可重试错误的重试。如果在异步内核-嵌套接口中发生错误并且超过阈值,则分析和重试逻辑可以删除所有对该输入/输出函数的未完成请求;将该输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。如果发生严重错误,则分析和重试逻辑可以确定错误源。如果错误源是系统嵌套,则分析和重试逻辑可以全部删除,将所有相关的输入/输出函数设置为错误状态,并向操作系统发布异步错误信号。如果错误源是输入/输出总线控制器,则分析和重试逻辑可以删除对连接到该输入/输出总线控制器的输入/输出函数的所有未完成请求;将所有输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。因此,有可能有利地进行系统恢复,其中仅真实错误可以得到处理。

[0035] 根据本发明的数据处理系统的有利实施例,系统消息可以包括以下之一:分层物理目标地址;提供SMT(同时多线程)线程或聚合缓冲区标识符;数据长度;输入/输出总线地址;或输入/输出状态缓冲区索引。由此,可以保证相关信息通过数据处理系统的有利传递。

[0036] 此外,提出了一种用于处理针对数据处理系统的至少一个外部设备的输入/输出存储指令的方法,该数据处理系统包括通过输入/输出总线控制器通信地耦合到至少一个输入/输出总线的系统嵌套。该数据处理系统还包括至少一个数据处理单元,该数据处理单元包括内核、系统固件和异步内核-嵌套接口。数据处理单元经由聚合缓冲区通信地耦合到系统嵌套。外部设备通信地耦合到输入/输出总线。异步内核-嵌套接口包括具有多个输入/输出状态缓冲区的输入/输出状态阵列,以及阵列管理和访问逻辑。系统固件还包括重试缓冲区,内核包括分析和重试逻辑。

[0037] 该方法包括:(i)在数据处理系统上运行的操作系统发布输入/输出存储指令,其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数;(ii)数据处理单元被配置以通过在输入/输出存储指令中指定的地址来识别输入/输出函数;(iii)数据处理单元被配置以验证在地址空间和客户机实例级别上是否允许对输入/输出函数的访问,客户机在数据处理系统上运行;(iv)据处理单元被配置以在系统嵌套中输入/输出存储指令的执行完成之前完成输入/输出存储指令;(v)

系统固件被配置得如果在输入/输出存储指令的异步执行期间由数据处理单元检测到错误,则通过中断、发送失败的异步执行的数据,来通知操作系统;(vi)分析和重试逻辑分别检测错误,由硬件确保存储指令尚未转发到输入/输出总线;(vii)重试缓冲区保存存储信息,用于在系统硬件/固件中执行存储指令的重试;(viii)分析和重试逻辑分析错误,检查重试可能性;以及(ix)分析和重试逻辑触发重试。

[0038] 有利地,可因此同时允许多个未完成的异步存储指令以减少重复的异步存储指令的每个指令的周期。在异步存储指令和同步加载/存储指令之间定义排序。支持多个未完成的异步存储指令基于多个状态消息的簿记和响应与状态项的相关性。

[0039] 根据本发明的另一实施例的方法包括经由输入/输出总线从数据处理系统的外部设备加载和存储到数据处理系统的外部设备的指令。异步指令在数据已被存储到外部设备之前完成,而同步指令在数据已被存储到外部设备之后完成。在这里描述的实施例内,PCI将可互换地用于任何其他输入/输出技术,因此不将本发明的实施例限制于PCI。

[0040] 本发明方法的实施例描述了以从架构边界上方可观察到的严格有序方式的输入/输出存储指令执行,而实际执行在数据处理单元(CPU)的硬件内可能是无序的。

[0041] 根据本发明的方法的实施例,可以通过PCIe存储效果的异步执行和异步状态处理来执行PCI存储指令。异步可靠执行基于本发明数据处理系统的微架构中的可靠转发机制。

[0042] 现有的PCI存储和存储块指令通常是同步的,直到PCI存储数据已经被传送到PCIe接口且完成被返回到处理单元。

[0043] PCI标准只需要PCI信息的异步发送命令,通常通过处理器中的存储队列聚合异步发送的数据来实现。

[0044] 有利地,根据本发明方法的实施例,可以通过输入/输出存储指令的可靠异步发送处理替换同步PCI指令来实现关于每个指令的周期的改进。

[0045] 作为要传送的数据的替换或补充,根据本发明的实施例的存储指令还可指定指向主存储器的指针,该指针应当用于从其获取数据,而不是直接包含该数据。

[0046] 客户机实例级别还可意味着单个客户机或主机可在数据处理系统上运行。

[0047] 输入/输出函数本身的偏址的地址可以是虚拟、物理、逻辑地址。虚拟和逻辑地址通常经由存储器管理单元(MMU)被转换成物理地址,然后可以通过物理地址来识别所指的是哪个函数和偏址。

[0048] 本文中的物理地址是指“可从客户机/操作系统内访问的地址转换层次结构中的最低地址”。

[0049] 根据本发明方法的有利实施例,分析和重试逻辑计算重试错误的数量,检查错误检测的阈值,并向操作系统报告失败的重试次数。因此,可以有利地进行对所发生的可重试错误的确定。

[0050] 有利地,输入/输出状态缓冲区可从系统嵌套和/或从输入/输出总线控制器收集消息状态,特别是从系统嵌套收集完成状态,其中消息状态和/或完成状态通过输入/输出状态缓冲区索引来编号。通过这种方式,可以以有序且高效的方式处理关于不同存储指令的完成状态的信息。编号使得能够以有序且有效的方式处理消息并且特别是完成状态以进一步处理其他存储指令的可能性。

[0051] 有利地,系统固件可包括阵列管理逻辑、分配/释放输入/输出状态阵列中的输入/

输出状态缓冲区和/或发起新存储指令的开始。因此,空闲状态缓冲区可归因于进一步的存储指令。可以以高效省时的方式进行对存储指令的有序处理。

[0052] 根据有利的实施例,该方法可进一步包括:(i)操作系统发布输入/输出存储指令;(ii)系统固件(10)分配空闲输入/输出状态缓冲区索引;如果不存在可用的空闲输入/输出状态缓冲区索引,则等待空闲输入/输出状态缓冲区索引;(iii)系统固件将存储指令注入到异步发送引擎中;如果被另一存储指令阻挡,则等待直到存储指令完成;(iv)取决于数据的长度:如果数据的长度超过8字节,系统固件软件重复下发系统消息将数据包发送到聚合缓冲区,直到存储块的所有数据都已转发到聚合缓冲区,所述系统固件软件等待直至所述系统消息已发送所述数据;否则,系统固件发布系统消息以将数据发送到聚合缓冲区;进一步独立于所述数据的长度,(v)所述系统固件向所述聚合缓冲区发布系统消息以将所述数据作为单个嵌套消息异步地转发到所述输入/输出总线控制器,同时等待所述聚合缓冲区发送完成消息;(vi)所述聚合缓冲区将所述嵌套消息注入到所述系统嵌套中,其中所述聚合缓冲区空闲以供在所述发送操作之后立即重新使用,从而发信号回所述系统固件;然后聚合缓冲区发送空闲供重用消息;(vii)系统嵌套转发消息到目标位置;(viii)输入/输出总线控制器接收消息并且将数据帧中的数据转发到输入/输出总线;(ix)所述输入/输出总线控制器将完成消息发送到所述系统嵌套;(x)所述系统嵌套将所述完成消息转发到所述发起聚合缓冲区;(xi)聚合缓冲区将完成转发至异步内核-嵌套接口;(xii)异步内核-嵌套接口将完成状态存储在输入/输出状态缓冲区中以用于输入/输出状态缓冲区索引并且将操作的完成发信号通知给系统固件;(xiii)系统固件通过输入/输出状态缓冲区索引更新输入/输出状态缓冲区跟踪;以及(xiv)系统固件在错误的情况下将缺陷异步地发信号通知给操作系统。

[0053] 仅步骤(ii)与数据的长度有关,并且对于超过8个字节的数据长度来说与对于不超过8个字节的数据长度来说是不同的。

[0054] 根据本发明方法的实施例,将数据以分片(slices)形式传输到聚合缓冲区,直到存储块的所有数据被转发到聚合缓冲区,其中,系统固件等待直到数据已由异步内核-嵌套接口发送。

[0055] 由此,如果数据小于8字节,则可以跳过用数据包以分片形式的填充聚合缓冲区的过程,并且可以在单个步骤中完成数据到外部设备的传输过程。

[0056] 根据本发明方法的有利实施方式,如果源数据的长度超过8字节,数据可以由输入/输出存储指令通过具有早期完成消息的异步传输机制在多个数据包中传输到聚合缓冲区。该异步传输机制是有利的,因为发送设备在较早的状态时空闲供重用(free for reuse)。

[0057] 根据本发明方法的有利实施方式,系统固件可使用异步输入/输出驱动程序代码来处理输入/输出存储指令。由此,可以用异步传送机制将数据从数据处理单元传送到外部设备。

[0058] 根据本发明方法的有利实施例,内核可使用异步设置代码来处理对异步输入/输出驱动程序代码的状态信息的存储器要求。该异步设置代码可进一步促进通过聚合缓冲区到系统嵌套和输入/输出总线控制器的异步传输机制。

[0059] 根据本发明方法的有利实施例,所述异步内核-嵌套接口可使用异步内核-嵌套接

口转发组件转发本地完成的数据。因此,可以支持用于将数据在数据包中发送到聚合缓冲区的有利的异步传输模式。

[0060] 根据本发明方法的有利实施例,聚合缓冲区可以使用早期完成逻辑用于在发送请求之后传送空闲供重用消息。这使得能够早期继续经由聚合缓冲区向系统嵌套和输入/输出总线控制器传输数据的处理。

[0061] 有利地,输入/输出状态缓冲区可从系统嵌套和/或从输入/输出总线控制器收集返回状态,特别是从系统嵌套收集完成消息。这些输入/输出状态缓冲区可收集返回状态,由此充当支持异步传送过程的异步系统消息缓冲区。

[0062] 根据本发明方法的有利实施例,系统消息可包括以下之一:分层物理目标地址;提供(sourcing)SMT线程或聚合缓冲区标识符;数据的长度;输入/输出总线地址;或输入/输出状态缓冲区索引。由此,可以保证相关信息通过数据处理系统的有利传递。

[0063] 根据本发明方法的有利实施例,异步输入/输出驱动程序代码可以在重试缓冲区中存储每个输入/输出操作的副本,并且如果异步输入/输出驱动程序代码检测到错误,则可以将控制转移到分析和重试逻辑。通过这种方式,可以有利地实现系统恢复,其中仅可以处理实际错误。

[0064] 根据本发明方法的有利实施例,分析和重试逻辑可以确定失败的存储指令。此外,它可以确定该存储指令的输入/输出函数,如果低于阈值,则发起对可重试错误的重试。如果在异步内核-嵌套接口中发生错误并且超过阈值,则分析和重试逻辑可以删除所有对该输入/输出函数的未完成请求;将该输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。如果发生严重错误,则分析和重试逻辑可以确定错误源。如果错误源是系统嵌套,则分析和重试逻辑可以全部删除,将所有相关的输入/输出函数设置为错误状态,并向操作系统发布异步错误信号。如果错误源是输入/输出总线控制器,则分析和重试逻辑可以删除对连接到该输入/输出总线控制器的输入/输出函数的所有未完成请求;将所有输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。因此,有可能有利地进行系统恢复,其中仅真实错误可以得到处理。

[0065] 进一步,提出了一种有利的计算机程序产品,用于处理到数据处理系统的至少一个外部设备的输入/输出存储指令,该数据处理系统包括通过输入/输出总线控制器通信地耦合到至少一个输入/输出总线的系统嵌套。该数据处理系统还至少包括一个数据处理单元,该数据处理单元包括内核、系统固件和异步内核-嵌套接口。数据处理单元经由聚合缓冲区通信地耦合到系统嵌套。外部设备通信地耦合到输入/输出总线。异步内核-嵌套接口包括具有多个输入/输出状态缓冲区的输入/输出状态阵列,以及阵列管理和访问逻辑。系统固件还包括重试缓冲区,内核包括分析和重试逻辑。

[0066] 所述计算机程序产品包括计算机可读存储介质,所述计算机可读存储介质具有随其体现的程序指令,所述程序指令可由所述计算机系统执行以促使所述计算机系统执行一种方法,所述方法包括:(i)在所述数据处理系统上运行的操作系统,发布所述输入/输出存储指令,其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数;(ii)所述数据处理单元被配置以通过在所述输入/输出存储指令中指定的所述地址来识别所述输入/输出函数;(iii)所述数据处理单元配置以验证在地址空间上和在客户机实例级别上是否允许访问所述输入/输出函数,所述客户机

在所述数据处理系统运行;(iv)所述数据处理单元配置以在所述系统嵌套所述输入/输出存储指令执行完成之前完成所述输入/输出存储指令;(v)所述系统固件被配置得如果在所述输入/输出存储指令的异步执行期间由所述数据处理单元检测到错误,则通过中断、发送失败的异步执行的数据,来通知所述操作系统;(vi)分析和重试逻辑分别检测错误,由硬件确保存储指令尚未转发到输入/输出总线;(vii)重试缓冲区保存存储信息,用于在系统硬件/固件中执行存储指令的重试;(viii)分析和重试逻辑分析错误,检查重试可能性;以及(ix)分析和重试逻辑触发重试。

[0067] 进一步,提出了一种用于执行数据处理程序的数据处理系统,该数据处理系统包括用于执行上述方法的计算机可读程序指令。

附图说明

[0068] 从以下对实施例的详细描述中可以最佳地理解本发明以及上述和其他目的和优点,但是本发明不限于这些实施例。

[0069] 图1示出根据本发明的实施例的用于处理针对外部设备的输入/输出存储指令的数据处理系统的框图。

[0070] 图2示出根据本发明的实施例的用于处理针对外部设备的输入/输出存储指令的方法的消息序列图。

[0071] 图3示出根据本发明的实施例的用于处理针对外部设备的输入/输出存储指令的流程图的第一部分。

[0072] 图4示出根据本发明的实施例的用于处理针对外部设备的输入/输出存储指令的流程图的第二部分。

[0073] 图5示出用于执行根据本发明的方法的数据处理系统的示范性实施例。

具体实施方式

[0074] 在附图中,相同的元件用相同的附图标记表示。附图仅是示意性表示,并非旨在描述本发明的特定参数。此外,附图旨在仅描述本发明的典型实施例,因此不应被视为限制本发明的范围。

[0075] 本文中描述的说明性实施例提供用于处置输入/输出存储指令的数据处理系统和方法,该数据处理系统包括通过输入/输出总线控制器通信地耦合到至少一个输入/输出总线的系统嵌套(system nest)。该数据处理系统还至少包括一个数据处理单元,数据处理单元包括内核、系统固件和异步内核-嵌套接口。数据处理单元经由聚合缓冲区通信地耦合到系统嵌套。系统嵌套被配置以异步地从通信地耦合到输入/输出总线的外部设备加载数据和/或将数据存储到外部设备。异步内核-嵌套接口包括具有多个输入/输出状态缓冲区的输入/输出状态阵列,以及阵列管理和访问逻辑。系统固件还包括重试缓冲区,内核包括分析和重试逻辑。

[0076] 说明性实施例可以用于这样一种方法,该方法包括:(i)在数据处理系统上运行的操作系统至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数;(ii)数据处理单元被配置以通过在输入/输出存储指令中指定的地址来识别输入/输出函数;(iii)数据处理单元被配置以验证是否允许在地址

空间上和在客户机实例级别上对输入/输出函数的访问,客户机在数据处理系统上运行;(iv)数据处理单元被配置以在所述系统嵌套中完成所述输入/输出存储指令的执行之前完成所述输入/输出存储指令;(v)系统固件被配置得如果在输入/输出存储指令的异步执行期间由数据处理单元检测到错误,则通过中断、发送失败的异步执行的数据,来通知操作系统;(vi)分析和重试逻辑分别检测错误,由硬件确保存储指令尚未转发到输入/输出总线;(vii)重试缓冲区保存存储信息,用于在系统硬件/固件中执行存储指令的重试;(viii)分析和重试逻辑分析错误,检查重试可能性;以及(ix)分析和重试逻辑触发重试。

[0077] 可替代地或另外地,对于待传输的数据,根据本发明的实施例的存储指令还可以指定指向的指针,该指针应当用于从主存储器提取数据,而不是直接包含该数据。

[0078] 客户机实例级别还可意味着单个客户机或主机可在数据处理系统上运行。

[0079] 输入/输出函数本身的偏址的地址可以是虚拟、物理、逻辑地址。虚拟和逻辑地址通常通过存储器管理单元(MMU)被转换成物理地址,然后可以通过物理地址来识别所指的是哪个函数和偏址。

[0080] 本说明书上下文中的物理地址是指“从客户机/操作系统内可访问的地址转换层次结构中的最低地址”。

[0081] 图1示出根据本发明的实施例的用于处理到至少一个外部设备214的输入/输出存储指令30的数据处理系统210的框图。数据处理系统210包括通过输入/输出总线控制器20通信地耦合到输入/输出总线22的系统嵌套18、包括内核12、系统固件10和异步内核-嵌套接口14的数据处理单元216。输入/输出总线控制器20还可经由多个输入/输出总线22耦合到多个外部设备214。

[0082] 数据处理单元216经由聚合缓冲区16通信地耦合到系统嵌套18。系统嵌套18被配置以通过作为系统嵌套18的一部分的缓冲区输入/输出总线控制器接口28以及输入/输出总线控制器20异步地从通信地耦合到输入/输出总线22的外部设备214加载数据和/或将数据存储到外部设备214。异步内核-嵌套接口14包括具有多个输入/输出状态缓冲器24的输入/输出状态阵列44和阵列管理和访问逻辑46。系统固件10包括重试缓冲区52,内核12包括分析和重试逻辑54。分析和重试逻辑54计算重试错误的数量,检查错误检测的阈值,并向操作系统报告失败重试的次数。

[0083] 聚合缓冲区16通信地耦合到异步内核-嵌套接口14。系统固件10包括用于处理输入/输出存储指令30的异步输入/输出驱动程序代码32。内核12包括异步设置代码34,用于处理对异步输入/输出驱动程序代码32的状态信息的存储器要求。根据多个外部设备214的恢复语义,异步输入/输出驱动程序代码32在重试缓冲区52中存储每个输入/输出操作的副本,并且,如果异步输入/输出驱动程序代码32检测到错误,则将控制转移到分析和重试逻辑54。

[0084] 阵列管理和访问逻辑46提供到内核12的接口,以查询所有输入/输出状态缓冲器24的状态。此外,它还提供到内核12的接口,以重置选择的输入/输出状态缓冲器24的状态。

[0085] 分析和重试逻辑54确定失败的存储指令30。此外,它确定该存储指令30的输入/输出函数,如果低于阈值,则发起对可重试错误的重试。此外,如果在异步内核-嵌套接口14中发生错误并且超过阈值,则分析和重试逻辑54删除所有对该输入/输出函数的未完成请求;将该输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。如果发生严重错

误,则分析和重试逻辑54确定错误源。如果错误源是系统嵌套18,则分析和重试逻辑54删除所有输入/输出函数,将所有涉及的输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。如果错误源是输入/输出总线控制器20,则分析和重试逻辑54删除对连接到该输入/输出总线控制器20的输入/输出函数的所有未完成请求;将所有输入/输出函数设置为错误状态,并向操作系统发送异步错误信号。

[0086] 系统固件10包括阵列管理逻辑42,其分配/释放输入/输出状态阵列44中的输入/输出状态缓冲区24和/或发起新的存储指令30的开始。

[0087] 异步内核-嵌套接口14包括异步内核-嵌套接口转发组件36,用于转发本地完成的数据。聚合缓冲区16包括早期完成(early completion)逻辑26,用于在发送请求之后传送空闲供重用(free for reuse)消息。聚合缓冲区16经由异步总线38耦合到异步内核-嵌套接口14。异步内核-嵌套接口14包括具有多个输入/输出状态缓冲区24的输入/输出状态阵列44,以及阵列管理和访问逻辑46。输入/输出状态缓冲区24收集来自系统嵌套18和/或来自输入/输出总线控制器20的返回状态,特别是来自系统嵌套18的完成消息。输入/输出状态缓冲区24直接集成在异步内核-嵌套接口14中。带有阵列项的标识的消息48(例如发往输入/输出状态缓冲区24其中之一的完成消息)可被系统嵌套18接收。

[0088] 根据本发明方法的实施例,在数据处理系统210上运行的操作系统发布输入/输出存储指令30,其至少指定具有通过地址的偏址、要传送的数据和/或指向要传送的数据的指针以及所述数据的长度的输入/输出函数。数据处理单元216由此被配置以通过在输入/输出存储指令30中指定的地址来识别输入/输出函数。数据处理单元216被配置以验证在地址空间上和客户机实例级别上是否允许对输入/输出函数的访问,客户机在数据处理系统210上运行。数据处理单元216被配置以在输入/输出存储指令30在系统嵌套18中的执行完成之前完成输入/输出存储指令30。系统固件10被配置得如果在输入/输出存储指令30的异步执行期间由数据处理单元216检测到错误,则通过中断、发送失败的异步执行的数据,来通知操作系统。

[0089] 阵列管理和存取逻辑46收集存储指令30的完成,并且基于接收到的完成消息更新输入/输出状态缓冲区24。

[0090] 输入/输出状态缓冲区24从系统嵌套18和/或从输入/输出总线控制器20收集消息状态,特别是从系统嵌套18收集完成状态。最好可以通过输入/输出状态缓冲区索引对消息状态和/或完成状态进行编号。

[0091] 输入/输出存储指令30位于将系统硬件/固件50与用户侧40分开的架构边界的用户接口40一侧的数据处理系统210中。

[0092] 由此,如果源数据的长度超过8字节,数据可以由输入/输出存储指令30通过具有早期完成消息的异步传输机制在多个数据包中传输到聚合缓冲区16,否则,可以在一个数据包中传输数据。

[0093] 根据本发明的数据处理系统的实施例的系统消息包括以下之一:分层物理目标地址;提供SMT线程或聚合缓冲区标识符;数据的长度;输入/输出总线地址;或输入/输出状态缓冲区索引。

[0094] 用于处理到多个外部设备214的存储指令30的排队和排序语义可以有利地如以下描述的那样执行。针对单独的SMT线程与输入/输出函数关系,所有传统输入/输出加载/存

储操作可以相对于处理器单元216的单个线程进行排序。新的输入/输出存储指令彼此之间完全无序。将新的输入/输出存储指令针对传统输入/输出指令进行排序。用于不同输入/输出函数的所有输入/输出指令不彼此相对排序。

[0095] 图2示出根据本发明的实施例的用于处理针对外部设备214的输入/输出存储指令30的方法的消息序列图。

[0096] 如图2所示,该方法始于操作系统发布输入/输出存储指令30。在步骤S101中,系统固件10分配空闲输入输出状态缓冲区索引。如果没有空闲输入/输出状态缓冲区索引可用,则系统固件10等待。在步骤S103中,系统固件10检查是否能够将存储指令注入到异步发送引擎中。如果这是可能的,则该过程继续。如果这是不可能的,则延迟存储指令,直到导致延迟的存储指令完成。

[0097] 接着,如步骤S100、S104所示,如果数据的长度超过8字节,系统固件10重复发布将数据包发送到聚合缓冲区16的系统消息,直到存储块(store block)的所有数据都已经被转发到聚合缓冲区16,与此同时系统固件10等待,直到数据已通过系统消息被发送。在步骤S102和S106中,本地完成(local completion)消息被发送回系统固件10。

[0098] 然后在步骤S108中,系统固件10向聚合缓冲区16发布将数据以单个嵌套消息(nest message)的形式异步地转发到输入/输出总线控制器20的系统消息,与此同时等待聚合缓冲区16发送完成消息。

[0099] 接下来,在步骤S110中,聚合缓冲区16将该嵌套消息注入到系统嵌套18中,其中,聚合缓冲区16在发送操作之后马上变得空闲供重用,向系统固件10发回信号。然后,聚合缓冲区16发送空闲供重用消息。

[0100] 在步骤S112,系统嵌套18将消息转发到目标位置,接着是步骤S114,输入/输出总线控制器20接收消息并且将数据帧中的数据转发到输入/输出总线,接着在步骤S116,输入/输出总线控制器20将完成消息发送到系统嵌套18。

[0101] 接下来在步骤S118,系统嵌套18将完成消息转发到始发聚合缓冲区16,随后在步骤S120,聚合缓冲区16将完成消息转发到异步内核-嵌套接口14。然后在步骤S122,异步内核-嵌套接口14将相应输入/输出状态缓冲区索引的状态存储在输入/输出缓冲区24中,并向系统固件10发送通知操作完成的信号。最后,在步骤S123,系统固件10通过输入/输出状态缓冲区索引来更新输入/输出状态缓冲区24跟踪。输入/输出状态缓冲区24现在再次空闲。

[0102] 在数据传输期间发生错误的情况下,系统固件10向操作系统异步地发送通知缺陷的信号。

[0103] 在要传送的数据小于8字节的情况下,跳过对聚合缓冲区16的重复填充。

[0104] 图3示出根据本发明实施例的用于处理针对外部设备214的输入/输出存储指令30的流程图的第一部分,而图4示出流程图的第二部分。

[0105] 如果步骤S200中发生错误中断,则在步骤S202中,该中断由系统固件10作为消息接收。接下来,在步骤S204中,从包括输入/输出状态缓冲器24的输入/输出状态阵列中提取状态消息。在步骤S206中,确定受影响的输入/输出函数,步骤S208中,检查其是否是可重试错误。如果是这种情况,则在步骤S210中,检查是否达到阈值。如果是这种情况,则过程流在图4中所示的流程图的接续点A处继续。如果不是,则在步骤S212中,递增重试计数器,然后,

在步骤S214中清除该状态阵列项中的错误。然后,在步骤S218中,触发重试。接下来,在步骤S220中,检查是否在状态阵列44中发现错误。如果是这种情况,则过程循环回步骤S206,确定受影响的函数。如果不是,过程将结束。

[0106] 如果在步骤S208中确定没有可重试错误,则在步骤S222中检查该错误是否是严重错误。然后,检查是否连接到输入/输出总线22的函数受未完成存储函数的影响(步骤S226)或者是否所有函数都受未完成存储函数的影响(步骤S228)。在步骤S230中,将受影响的函数设置为错误状态,然后在步骤S232中,异步地向操作系统发送错误信号。

[0107] 如果在步骤S222中未发现严重错误,则在步骤S230中,将受影响的函数设置为错误状态,然后,在步骤S322中,异步地向操作系统发送错误信号,并且并行地,过程流在图4中所示的流程图的接续点A处继续。

[0108] 有关所确定的受影响的输入/输出函数的信息存储在存储器228中的存储数据影子副本62中。有关设置为错误状态的受影响函数的信息,存储在存储器228中的存储访问表64中。

[0109] 图4中示出以连接点A开始的流程图的第二部分。首先在步骤S304,检查是否要传送8个以上的字节。如果是这种情况,则在步骤S306,项。在步骤S308,系统固件等待,直到在步骤S310发送了本地完成的消息,返回到步骤S304。如果在步骤S304的检查中剩余不到8个字节,则流程在步骤S312继续,内核-嵌套接口发送异步输入/输出消息,随后在步骤S314等待步骤S316中的缓冲区响应。然后,在步骤S318,执行结束存储块(finish store block)指令,然后在步骤S320,流程以系统固件的结束而结束。

[0110] 在步骤S328中,异步内核-嵌套接口逻辑启动出站(outbound)处理循环,随后,在步骤S322接收聚合缓冲区完成消息,并在步骤S324向聚合缓冲区转发数据消息,随后,在步骤S326将完成消息发送回系统固件。在步骤S330,接收到异步输入/输出发送(input/output send)消息,随后向聚合缓冲区转发该输入/输出发送消息。

[0111] 在步骤S338,聚合缓冲区逻辑开始出站处理循环,随后在步骤S334接收数据,并在步骤S336在聚合缓冲区中聚合数据。在步骤S340,聚合缓冲区也接收输入/输出发送消息,随后在步骤S242用输入/输出发送消息转发聚合缓冲区中的数据。接下来在步骤S344,通过内核-嵌套接口向系统固件发送来自聚合缓冲区的响应消息。

[0112] 现在参照图5,示出了数据处理系统210的示例的示意图。数据处理系统210仅是合适的数据处理系统的一个实例,并且不旨在对本文所述本发明实施例的使用或功能的范围提出任何限制。无论如何,数据处理系统210能够被实现和/或执行以上所述的任何功能。

[0113] 数据处理系统210中有计算机系统/服务器212,其可与许多其他通用或专用计算系统环境或配置一起操作。可以适合于与计算机系统/服务器212一起使用的众所周知的计算系统、环境和/或配置的示例包括但不限于个人计算机系统、服务器计算机系统、瘦客户机、厚客户机,手持式或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、网络PC、小型计算机系统、大型计算机系统以及包括任何上述系统或设备的分布式云计算环境,等等。

[0114] 计算机系统/服务器212可以在由计算机系统执行的计算机系统可执行指令(诸如程序模块)的一般上下文中描述。一般而言,程序模块可包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、逻辑、数据结构等。计算机系统/服务器212可以在分布

式云计算环境中实践,其中任务由通过通信网络链接的远程处理设备来执行。在分布式云计算环境中,程序模块可位于包括存储器存储设备的本地和远程计算机系统存储介质两者中。

[0115] 如图5所示,数据处理系统210中的计算机系统/服务器212以通用计算设备的形式示出。计算机系统/服务器212的组件可以包括但不限于一个或多个处理器或处理单元216、系统存储器228和将包括系统存储器228的不同系统组件耦合到处理器216的总线218。

[0116] 总线218表示若干类型的总线结构中的任一种总线结构中的一种或多种,包括存储器总线或存储器控制器、外围总线、加速图形端口、以及使用各种总线架构中的任一种的处理器或局部总线。作为示例而非限制,此类架构包括工业标准架构(ISA)总线、微通道架构(MCA)总线、增强型ISA(EISA)总线、视频电子标准协会(VESA)局部总线和外围组件互连(PCI)总线。

[0117] 计算机系统/服务器212通常包括各种计算机系统可读介质。这样的介质可以是可由计算机系统/服务器212访问的任何可用介质,并且包括易失性和非易失性介质、可移动和不可移动介质。

[0118] 系统存储器228可包括易失性存储器形式的计算机系统可读介质,诸如随机存取存储器(RAM)230和/或高速缓冲存储器232。计算机系统/服务器212还可以包括其他可移除/不可移除、易失性/非易失性计算机系统存储介质。仅通过举例,存储系统234可以被提供用于从不可移除、非易失性磁介质(未予示出,通常被称为“硬盘驱动器”)读取和向其写入。尽管未示出,可以提供用于读写可移动非易失性磁盘(例如“软盘”)的磁盘驱动器以及用于读写诸如CD-ROM、DVD-ROM或其他光学介质之类的可移除非易失性光盘的光盘驱动器。在这样的实例中,每一个都可以通过一个或多个数据介质接口连接到总线218。如下面将进一步描绘和描述的,存储器228可以包括具有被配置以执行本发明的实施例的功能的一组(例如至少一个)程序模块的至少一个程序产品。

[0119] 具有一组(至少一个)程序模块242的程序/实用工具240以及操作系统、一个或多个应用程序、其他程序模块和程序数据可以通过示例而非限制的方式存储在存储器228中。操作系统、一个或多个应用程序、其他程序模块和程序数据中的每一者或其某一组合可包含联网环境的实施例。程序模块242通常执行本文所述的本发明实施例的功能和/或方法。

[0120] 计算机系统/服务器212还可以与以下各项通信:一个或多个诸如键盘、定点设备、显示器224之类的外部设备214;使得用户能够与计算机系统/服务器212交互的一个或多个设备;和/或使计算机系统/服务器212能够与一个或多个其他计算设备通信的任何设备(例如网卡、调制解调器等)。这样的通信可以通过输入/输出(I/O)接口222进行。此外,计算机系统/服务器212可以通过网络适配器220与诸如局域网(LAN)、通用广域网(WAN)和/或公共网络(例如因特网)之类的一个或多个网络通信。如图所示,网络适配器220通过总线218与计算机系统/服务器212的其他组件通信。应当理解,虽然未示出,但是其他硬件和/或软件组件可以与计算机系统/服务器212结合使用。示例包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动器阵列、RAID系统、磁带驱动器和数据归档存储系统等。

[0121] 本发明可以是系统、方法和/或计算机程序产品。所述计算机程序产品可包含上面具有计算机可读程序指令的计算机可读存储介质(或介质),所述计算机可读程序指令用于致使处理器执行本发明的方面。

[0122] 计算机可读存储介质可以是可以保留和存储指令以供指令执行设备使用的有形设备。计算机可读存储介质可以是例如但不限于电子存储设备、磁存储设备、光存储设备、电磁存储设备、半导体存储设备或前述各项的任何合适的组合。计算机可读存储介质的更具体例子的非穷举列表包括以下:便携式计算机盘,硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦除可编程只读存储器(EPROM或闪存),静态随机存取存储器(SRAM)、便携式致密盘只读存储器(CD-ROM),数字通用盘(DVD)、记忆棒、软盘、机械编码设备(诸如穿孔卡片或具有记录在其上的指令的凹槽中的凸起结构),以及上述的任意合适的组合。如本文中所使用的计算机可读存储介质不应被解释为瞬态信号本身,诸如无线电波或其他自由传播的电磁波、通过波导或其他传输介质传播的电磁波(例如通过光纤电缆的光脉冲)、或通过导线传输的电信号。

[0123] 本文所述的计算机可读程序指令可从计算机可读存储介质下载到相应的计算/处理设备,或经由网络(例如因特网、局域网、广域网和/或无线网络)下载到外部计算机或外部存储设备。网络可以包括铜传输电缆、光传输光纤、无线传输、路由器、防火墙、交换机、网关计算机和/或边缘服务器。每个计算/处理设备中的网络适配器卡或网络接口从网络接收计算机可读程序指令,并转发计算机可读程序指令以存储在相应计算/处理设备内的计算机可读存储介质中。

[0124] 用于执行本发明的操作的计算机可读程序指令可以是汇编指令,指令集架构(ISA)指令、机器指令、机器相关指令、微代码、固件指令、状态设置数据,或者以一种或多种编程语言的任意组合编写的源代码或目标代码,所述编程语言包括诸如Smalltalk、C++等面向对象的编程语言,以及诸如“C”编程语言或类似的编程语言的常规过程式编程语言。计算机可读程序指令可完全在用户的计算机上执行、部分在用户”的计算机上执行、作为独立软件包执行、部分在用户的计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在后一种情形中,远程计算机可以通过任何类型的网络(包括局域网(LAN)或广域网(WAN))连接到用户的计算机,或者可以连接到外部计算机(例如通过使用因特网服务提供商的因特网)。在一些实施例中,电子电路(包括例如可编程逻辑电路、现场可编程门阵列(FPGA)或可编程逻辑阵列(PLA))可以通过利用计算机可读程序指令的状态信息来执行计算机可读程序指令以使电子电路个性化,以便执行本发明的各方面。

[0125] 本文中参照根据本发明的实施例的方法、设备(系统)和计算机程序产品的流程图说明和/或框图描述了本发明的各方面。应当理解,流程图和/或框图的每个方框以及流程图和/或框图中各方框的组合,都可以由计算机可读程序指令来实现。

[0126] 这些计算机可读程序指令可以被提供给通用计算机的处理器、专用计算机或其他可编程数据处理装置以产生机器,使得指令通过计算机的处理器或其他可编程数据处理装置的执行,创建用于实现在流程图和/或方框图的一个或多个方框中指定的功能/动作的装置。这些计算机可读程序指令还可存储在计算机可读存储介质中,后者可指令计算机、可编程数据处理装置和/或其他装置以特定方式起作用,使得具有存储在其中的指令的计算机可读存储介质包括制品,该制品包括实现流程图和/或框图中的一个或多个方框中规定的功能/动作的各方面的指令。

[0127] 计算机可读程序指令还可以加载到计算机、其他可编程数据处理装置上或其他装置,使得在计算机、其他可编程装置或其他设备上执行一系列操作步骤,以产生计算机实现

的过程,使得在计算机、其他可编程装置或其他设备上执行的指令实现流程图和/或框图中的一个或多个方框中规定的功能/动作。

[0128] 附图中的流程图和框图图示了根据本发明的不同实施例的系统、方法和计算机程序产品的可能实现的架构、功能和操作。就此,流程图或框图中的每个方框可以代表指令的模块、段或部分,其包括用于实现规定的逻辑功能的一个或多个可执行指令。在一些替代实现方式中,框中所标注的功能可以不以图中所标注的顺序发生。例如,取决于所涉及的功能,连续示出的两个框实际上可以基本上同时执行,或者这些框有时可以以相反的顺序执行。还将注意的是,框图和/或流程图中的每个框、以及框图和/或流程图中的框的组合可以由执行指定功能或动作或执行专用硬件与计算机指令的组合的基于专用硬件的系统来实现。

[0129] 已经出于说明的目的给出了本发明的不同实施例的描述,但以上描述并不旨在是穷尽性的或局限于所披露的实施例。在不背离所描述的实施例的范围和精神的情况下,许多修改和变化对本领域的普通技术人员而言将是显而易见的。选择在此使用的术语以最佳地解释实施例的原理、实际应用或对市场上的技术的技术改进,或使得本领域普通技术人员能够理解本文中披露的实施例。

[0130] 附图标记

[0131] 10 系统FW

[0132] 12 内核

[0133] 14 异步内核-嵌套IF

[0134] 16 聚合缓冲区

[0135] 18 系统嵌套

[0136] 20 I/O总线控制器

[0137] 22 I/O总线

[0138] 24 I/O状态缓冲区

[0139] 26 早期完成逻辑

[0140] 28 缓冲区-I/O总线控制器IF

[0141] 30 I/O存储指令

[0142] 32 异步I/O驱动代码

[0143] 34 I/O设置代码

[0144] 36 异步转发

[0145] 38 异步总线

[0146] 40 用户IF

[0147] 42 阵列管理逻辑

[0148] 44 I/O状态阵列

[0149] 46 阵列管理与访问逻辑

[0150] 48 带有阵列项的标识的消息

[0151] 50 系统HW/FW

[0152] 52 重试缓冲区

[0153] 54 分析和重试逻辑

[0154] 60

[0155] 62 存储数据影子副本

[0156] 64 存储访问表

[0157] 210 数据处理系统

[0158] 212 计算机系统/服务器

[0159] 214 外部设备

[0160] 216 CPU/数据处理单元

[0161] 218 输入输出总线

[0162] 220 网络适配器

[0163] 222 输入输出接口

[0164] 224 显示器

[0165] 228 存储器

[0166] 230 RAM

[0167] 232 高速缓冲存储器

[0168] 234 存储系统

[0169] 240 程序/实用程序

[0170] 242 程序模块

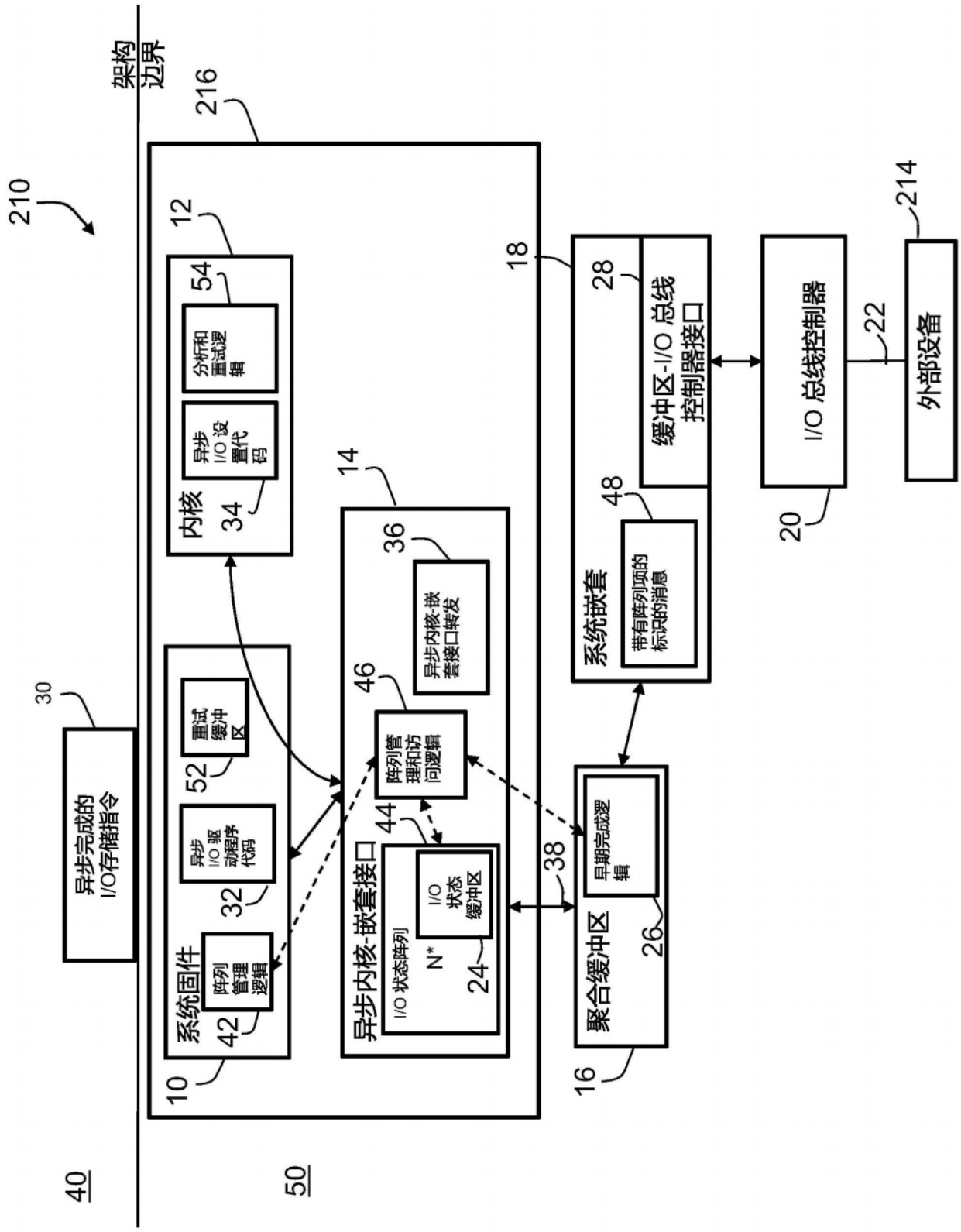

架构边界
210
216
12
54
分析和重试逻辑
异步I/O设置代码
内核
34
14
36
异步内核-嵌套接口转发
46
阵列管理和访问逻辑
44
I/O状态阵列
N*
I/O状态缓冲区
24
异步内核-嵌套接口
38
18
28
缓冲区-I/O总线控制器接口
系统嵌套
48
带有阵列项的标识的消息
20
I/O总线控制器
22
214
外部设备
30
异步完成的I/O存储指令
系统固件
42
阵列管理逻辑
32
异步I/O驱动程序代码
52
重试缓冲区
10
聚合缓冲区
早期完成逻辑
26
16
40
50

图1

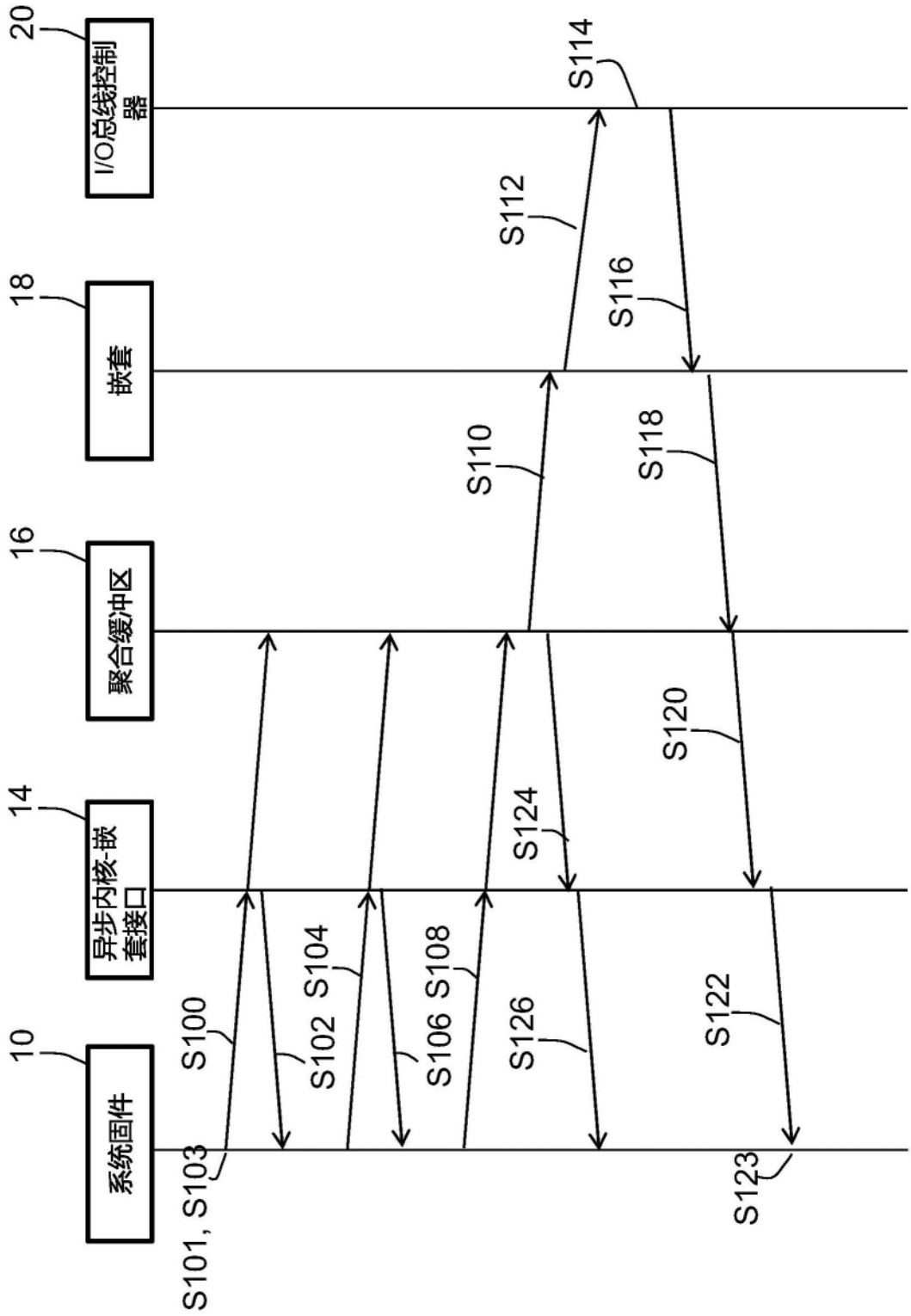

20
I/O总线控制器
18
嵌套
16
聚合缓冲区
14
异步内核-嵌套接口
10
系统固件
S100
S101, S103
S102
S104
S106
S108
S110
S112
S114
S116
S118
S120
S122
S123
S124
S126

图2

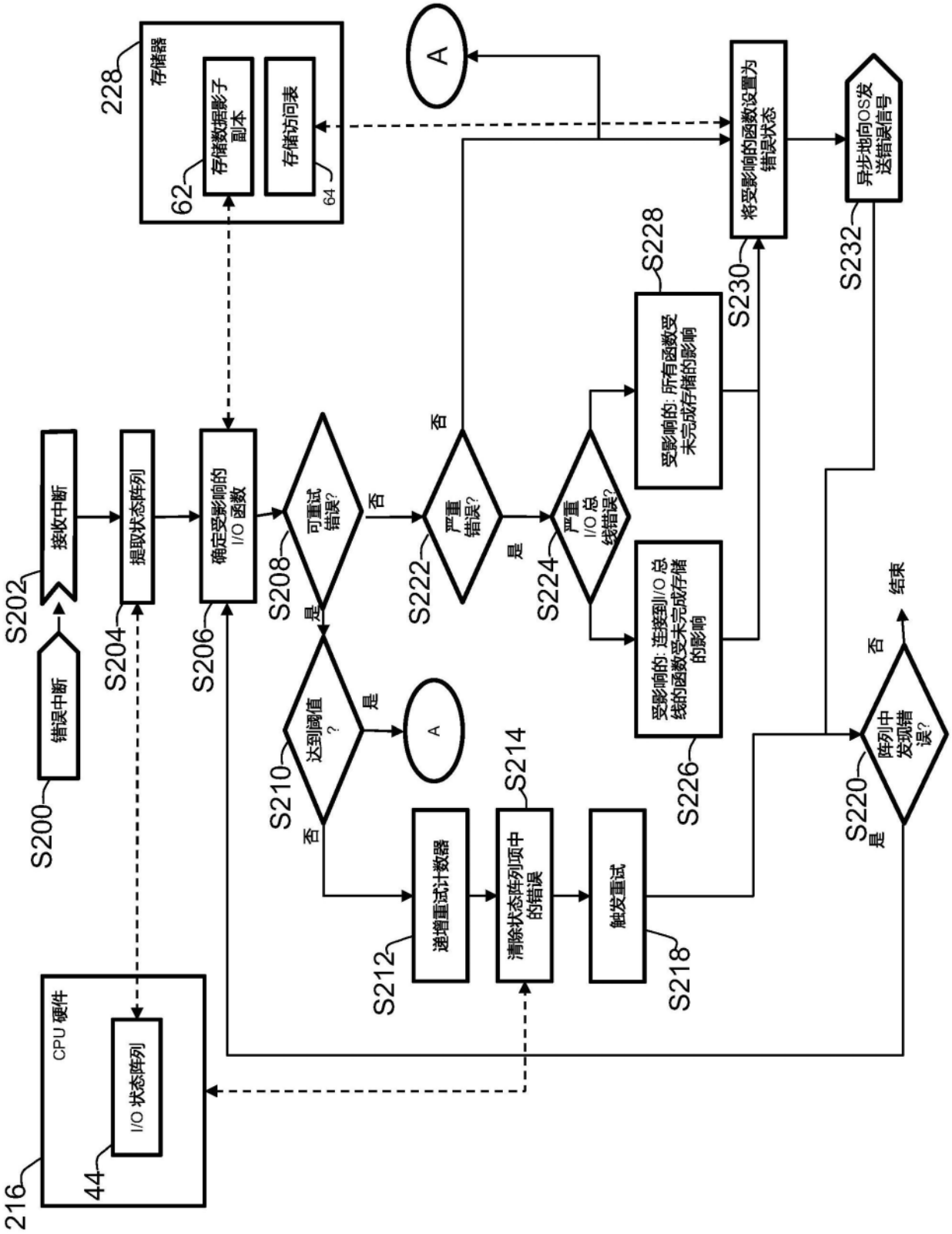

存储器
存储数据影子副本
存储访问表
62
64
228
A
接收中断
错误中断
S202
S200
提取状态阵列
S204
确定受影响的I/O 函数
S206
可重试错误?
S208
否
是
严重错误?
S222
否
是
严重I/O 总线错误?
S224
受影响的: 所有函数受未完成存储的影响
S228
受影响的: 连接到I/O 总线的函数受未完成存储的影响
S226
将受影响的函数设置为错误状态
S230
异步地向OS发送错误信号
S232
达到阈值?
S210
是
否
A
递增重试计数器
S212
清除状态阵列项中的错误
S214
触发重试
S218
阵列中发现错误?
S220
否
是
结束
CPU 硬件
I/O 状态阵列
44
216

图3

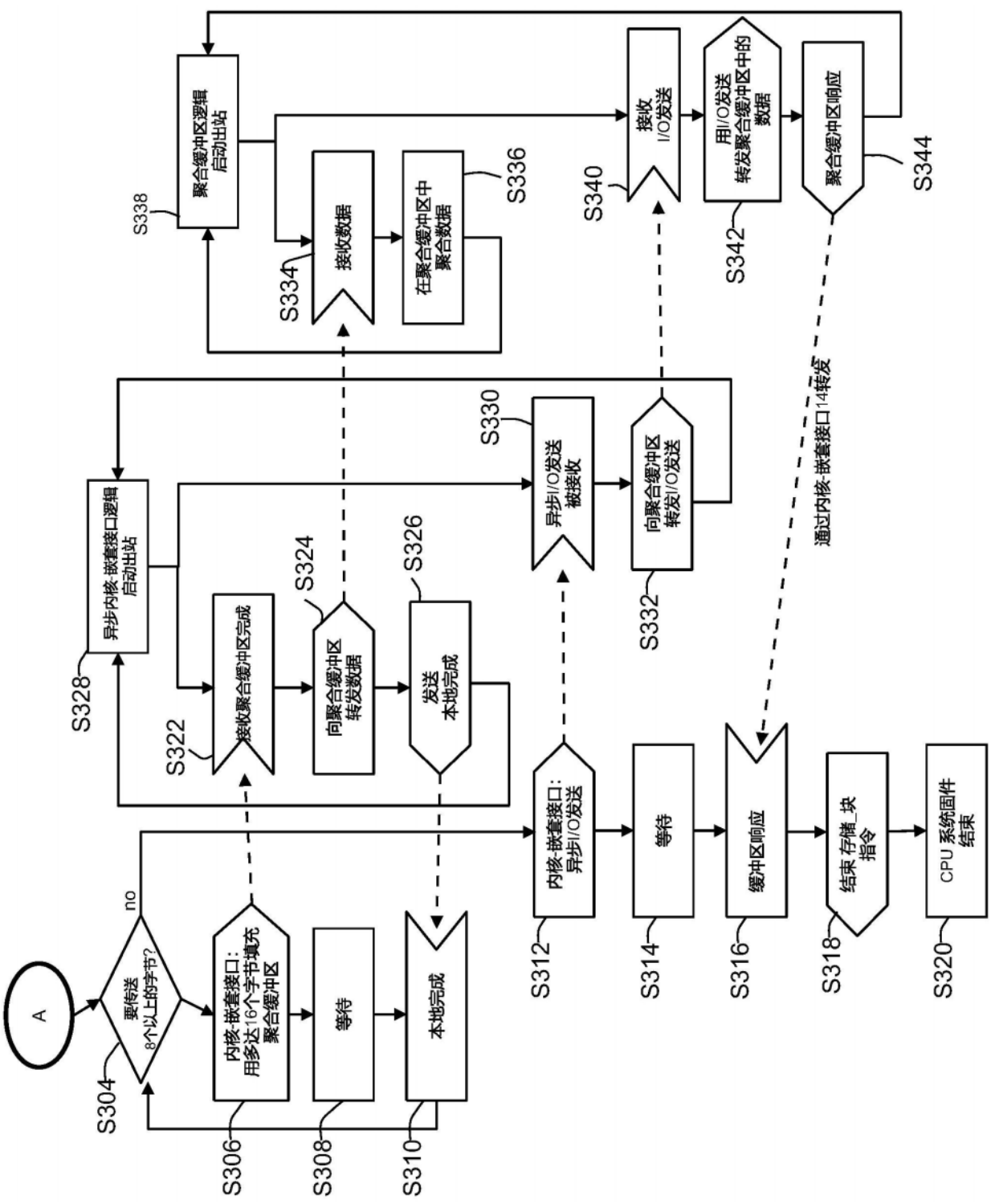

A
要传送8个以上的字节?
no
内核:嵌套接口:用多达16个字节填充聚合缓冲区
等待
本地完成
内核:嵌套接口:异步I/O发送
等待
缓冲区响应
结束 存储_块指令
CPU 系统固件结束
异步内核:嵌套接口逻辑启动出站
接收聚合缓冲区完成
向聚合缓冲区转发数据
发送本地完成
异步I/O发送被接收
向聚合缓冲区转发I/O发送
聚合缓冲区逻辑启动出站
接收数据
在聚合缓冲区中聚合数据
接收I/O发送
用I/O发送转发聚合缓冲区中的数据
聚合缓冲区响应
通过内核:嵌套接口14转发
S304
S306
S308
S310
S312
S314
S316
S318
S320
S322
S324
S326
S328
S330
S332
S334
S336
S338
S340
S342
S344

图4

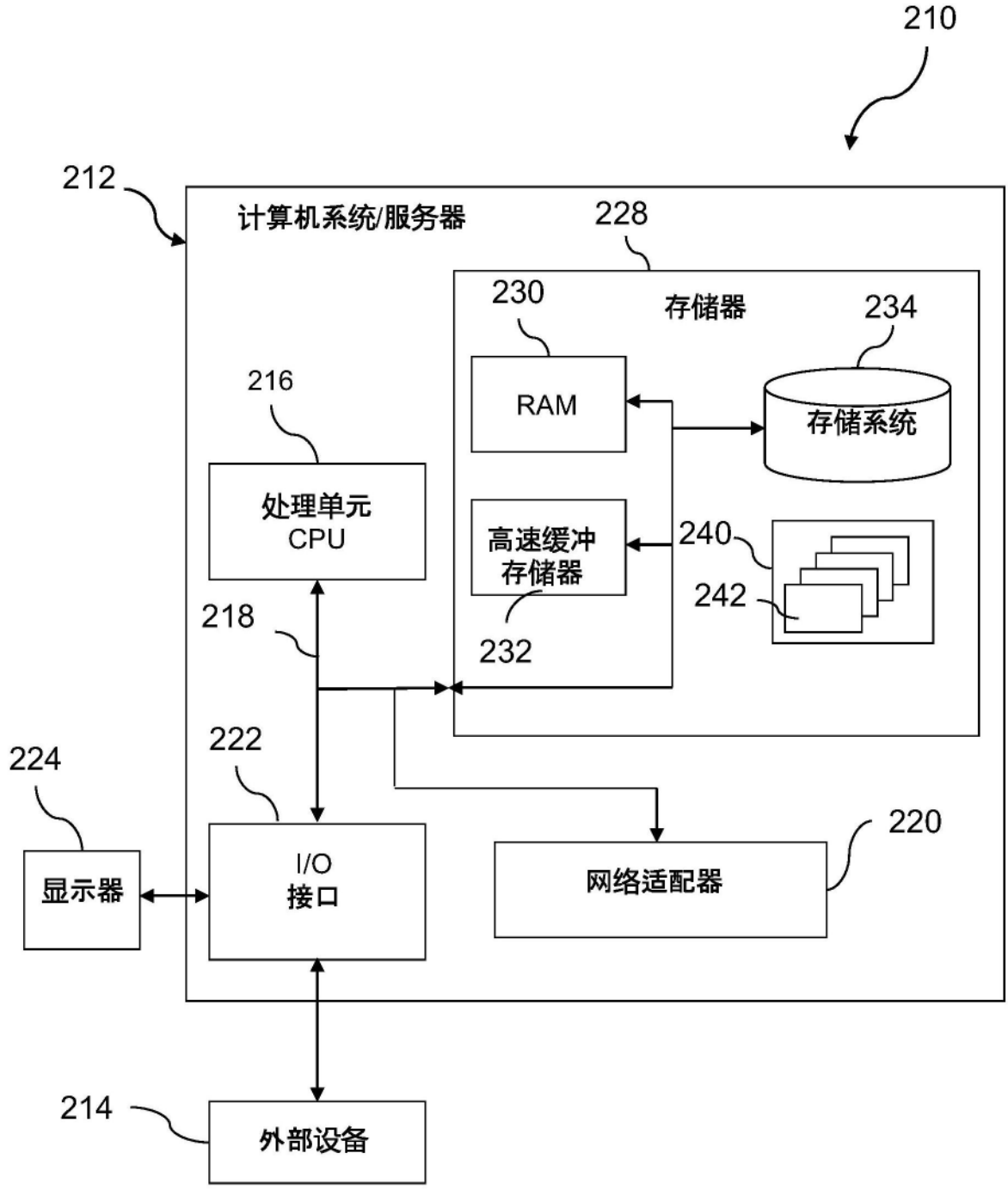

210
212
计算机系统/服务器
228
存储器
230
234
RAM
存储系统
216
处理单元
CPU
高速缓冲
存储器
240
242
232
218
222
224
显示器
I/O
接口
网络适配器
220
214
外部设备

图5