(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 104049909 A
(43)申请公布日 2014.09.17

(21)申请号 201410095476.8

(22)申请日 2014.03.14

(30)优先权数据
13/835,521 2013.03.15 US

(71)申请人 国际商业机器公司
地址 美国纽约阿芒克

(72)发明人 G.A. 范赫本 P.J. 米尼
J.S. 多德森 S.H. 里德
J.C. 格雷格森 E.E. 里特
I.G. 贝萨 G.D. 吉尔达 L.D. 柯利
V.K. 帕帕佐瓦

(74)专利代理机构 北京市柳沈律师事务所 11105
代理人 周少杰

(51)Int.Cl.
G06F 3/06(2006.01)

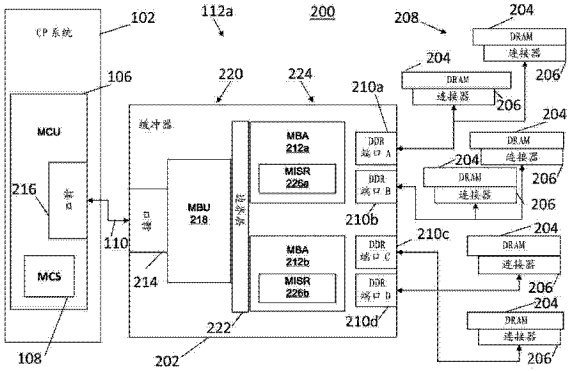
权利要求书3页 说明书15页 附图9页

(54)发明名称

双异步和同步存储器系统

(57)摘要

实施例涉及双异步和同步存储器系统。一个方面是包括存储器控制器和经由同步信道耦接到存储器控制器的存储器缓冲器芯片。存储器缓冲器芯片包括:存储器缓冲器单元,配置为与嵌套域中的存储器控制器同步通信;以及存储器缓冲器适配器,配置为与存储器域中的至少一个存储器接口端口通信。至少一个存储器接口端口可操作为存取至少一个存储器设备。边界层连接到嵌套域和存储器域,其中边界层可配置为在嵌套和存储器域之间以同步传输模式操作,并且在嵌套和存储器域之间以异步传输模式操作。



CN 104049909 A

1. 一种用于双异步和同步存储器操作的系统，所述系统包括：

存储器控制器；以及

经由同步信道耦接到所述存储器控制器的存储器缓冲器芯片，所述存储器缓冲器芯片包括：

存储器缓冲器单元，配置为与嵌套域中的所述存储器控制器同步通信；

存储器缓冲器适配器，配置为与存储器域中的至少一个存储器接口端口通信，所述至少一个存储器接口端口可操作为存取至少一个存储器设备；以及

连接到所述嵌套域和所述存储器域的边界层，所述边界层可配置为在嵌套和存储器域之间以同步传输模式操作，并且在嵌套和存储器域之间以异步传输模式操作。

2. 根据权利要求 1 所述的系统，进一步包括：

经由多个另外的同步信道耦接到所述存储器控制器的多个另外的存储器缓冲器芯片，所述多个另外的存储器缓冲器芯片可配置为与所述存储器缓冲器芯片同步操作，并且在嵌套域中关于彼此同步操作。

3. 根据权利要求 1 所述的系统，进一步包括：

存储器控制器锁相环，配置为向存储器控制器提供主时钟；

嵌套域锁相环，耦接到所述存储器缓冲器单元并且与所述存储器控制器锁相环同步可操作以建立嵌套域；以及

存储器域锁相环，耦接到所述存储器缓冲器适配器和至少一个存储器接口端口，所述存储器域锁相环可操作为建立存储器域。

4. 根据权利要求 3 所述的系统，其中所述嵌套域锁相环和所述存储器域锁相环可配置为输出相对于彼此不同的频率，所述嵌套域锁相环和所述存储器域锁相环在异步操作模式下由单独的参考时钟驱动，并且同步时钟对准逻辑向源自基于在同步操作模式下的嵌套域锁相环的参考的存储器域锁相环提供参考时钟。

5. 根据权利要求 4 所述的系统，其中所述同步时钟对准逻辑可操作为基于对准周期对准存储器域时钟边缘与来自所述嵌套域锁相环的嵌套域时钟边缘，所述对准周期由所述存储器控制器发起。

6. 根据权利要求 5 所述的系统，其中当在对准周期上从存储器控制器接收到同步命令时，在所述存储器缓冲器芯片内生成内部复位。

7. 根据权利要求 5 所述的系统，其中所述边界层配置为允许命令和数据在同步操作模式下时在对准周期上在所述嵌套域和所述存储器域之间跨过。

8. 根据权利要求 1 所述的系统，进一步包括与所述同步信道相关联的训练逻辑，所述训练逻辑配置为初始化和校准所述同步信道的路线，以计算用于同步信道的帧来回行程等待时间，并且基于计算的帧来回行程等待时间建立帧传输协议。

9. 一种用于存储器子系统中的双异步和同步存储器操作的计算机系统实现方法，所述方法包括：

在存储器控制器和存储器缓冲器芯片之间建立同步信道；

通过模式选择器基于所述存储器缓冲器芯片的操作模式，确定用于所述存储器缓冲器芯片的存储器域锁相环的参考时钟源；

基于同步的操作模式将嵌套域锁相环的输出作为参考时钟源提供给所述存储器缓冲

器芯片中的存储器域锁相环，所述嵌套域锁相环与所述存储器控制器的存储器控制器锁相环同步可操作；以及

基于异步的操作模式将独立于所述嵌套域锁相环的单独参考时钟作为参考时钟提供给所述存储器域锁相环。

10. 根据权利要求 9 所述的方法，进一步包括：

基于所述嵌套域锁相环生成嵌套域时钟；

基于所述存储器域锁相环生成存储器域时钟；

将所述嵌套域时钟提供给所述存储器缓冲器芯片的嵌套域中的存储器缓冲器单元；以及

将所述存储器域时钟提供给配置为与存储器域中的至少一个存储器接口端口通信的存储器缓冲器适配器，所述至少一个存储器接口端口可操作为存取至少一个存储器设备，其中边界层提供所述嵌套域和所述存储器域之间的接口。

11. 根据权利要求 9 所述的方法，其中多个另外的存储器缓冲器芯片经由多个另外的同步信道耦接到所述存储器控制器，所述方法进一步包括：

将所述多个另外的存储器缓冲器芯片配置为基于公共同步参考，与所述存储器缓冲器芯片同步操作，并且关于彼此同步操作。

12. 根据权利要求 9 所述的方法，进一步包括：

基于对准周期对准所述存储器域锁相环的存储器域的时钟边缘与所述嵌套域锁相环的嵌套域时钟边缘，所述对准周期由所述存储器控制器发起。

13. 根据权利要求 12 所述的方法，进一步包括：

当在对准周期上从所述存储器控制器接收到同步命令时，在所述存储器缓冲器芯片内生成内部复位。

14. 根据权利要求 12 所述的方法，其中所述嵌套域锁相环建立嵌套域，并且所述存储器域锁相环建立存储器域，并且所述方法进一步包括：

允许命令和数据在操作模式是同步时在对准周期上跨过所述嵌套域和所述存储器域之间的存储器缓冲器芯片的边界层。

15. 根据权利要求 9 所述的方法，进一步包括：

初始化和校准所述同步信道的路线；

计算用于所述同步信道的帧来回行程等待时间；以及

基于计算的帧来回行程等待时间建立帧传输协议。

16. 一种用于同步双异步和同步存储器子系统的系统，包括：

用于通过经由同步信道耦接到存储器缓冲器芯片的存储器控制器初始化和校准所述同步信道的路线的部件；

用于通过所述存储器控制器计算用于所述同步信道的帧来回行程等待时间的部件；

用于基于计算的帧来回行程等待时间建立帧传输协议的部件；以及

用于通过所述存储器控制器建立用于所述存储器缓冲器芯片的同步参考的部件，所述存储器缓冲器芯片包括相对于所述存储器控制器的同步操作模式和异步操作模式。

17. 根据权利要求 16 所述的系统，其中用于初始化和校准所述同步信道的路线的部件进一步包括：

用于对所述同步信道的驱动器缓冲器执行阻抗校准的部件；以及

用于执行所述同步信道的路线的导线测试的部件。

18. 根据权利要求 16 所述的系统，其中所述同步信道包括上行总线和下行总线，并且进一步包括：

用于在所述下行总线上发送包括固定模式的帧到所述存储器缓冲器芯片的部件；

用于基于所述存储器缓冲器芯片锁定到所述下行总线上的固定模式，从所述存储器缓冲器芯片接收所述上行总线上的固定模式的部件；以及

用于锁定到所述上行总线上的固定模式的部件。

19. 根据权利要求 16 所述的系统，其中用于计算用于所述同步信道的帧来回行程等待时间的部件进一步包括：

用于从所述存储器控制器发出空分组到存储器缓冲器设备的部件；

用于开始下行定时器的部件；

用于基于所述存储器缓冲器设备接收所述空分组从所述存储器缓冲器设备接收上行确收帧的部件；以及

用于基于接收所述上行确收帧设置下行来回行程等待时间值的部件。

20. 根据权利要求 16 所述的系统，其中多个另外的存储器缓冲器芯片经由多个另外的同步信道耦接到所述存储器控制器，并且进一步包括：

用于将同步命令发出到所述存储器缓冲器芯片和所述另外的存储器缓冲器芯片以建立用于所述存储器缓冲器芯片和所述另外的存储器缓冲器芯片的对准周期，从而对准各个存储器域与嵌套域的部件；

用于在对准周期上发出随后的同步命令以在所述存储器缓冲器芯片和所述另外的存储器缓冲器芯片中生成内部复位的部件；以及

用于监视对于错误情况、丧失同步情况和丧失顺序情况从所述存储器缓冲器芯片和所述另外的存储器缓冲器芯片返回的数据的部件。

双异步和同步存储器系统

技术领域

[0001] 本发明一般涉及计算机存储器，更具体地涉及双异步和同步存储器系统。

背景技术

[0002] 当前的高性能计算主存储器系统通常由一个或多个存储器设备组成，一个或多个存储器设备经由诸如缓冲器、集线器、总线到总线转换器之类的一个或多个存储器接口元件连接到一个或多个存储器控制器和/或处理器。存储器设备一般位于存储器子系统(诸如，存储卡或存储器模块)上，并且经常经由可插拔互连系统(例如，一个或多个连接器)连接到系统板(例如，PC母板)。

[0003] 总体计算机系统性能受到计算机结构的每个关键元件的影响，包括(多个)处理器、任何(多个)存储器高速缓存器(memory cache)、(多个)输入/输出(I/O)子系统的性能/结构；(多个)存储器控制功能的效率；(多个)主存储器设备和任何相关联存储器接口元件的性能；以及(多个)存储器互连接口的类型和结构。

[0004] 行业在发展的基础上投入广泛的研究和开发努力，以通过改进存储器系统/子系统设计和/或结构，创建改进和/或创新的方案来最大化总体系统性能和密度。由于消费者期望除了提供附加的功能、提高的性能、增大的存储、较低的操作成本等，新计算机系统还将关于平均故障间隔时间(MTBF,mean-time-between-failure)显著超过现有系统，高可用性系统进一步提出关于总体系统可靠性的进一步的挑战。其他频繁的消费者要求进一步加剧存储器系统设计挑战，并且包括如升级的容易和减小的系统环境冲击(诸如空间、电源和冷却)这样的项目。此外，消费者正在要求以越来越快的存取速度存取增加数量的更高密度的存储器设备(例如，DDR3和DDR4SDRAM)。

[0005] 鉴于变化的成本、容量和可扩缩性要求，可以需要考虑大量的存储器系统选项。经常需要在使用处理器和存储器缓冲器芯片之间的异步边界或设计完全同步系统之间进行选择。异步设计允许以固定的频率运行处理的灵活性，同时存储器缓冲器芯片可以编程为变化频率以匹配期望的存储器组件。例如，如果成本是最重要的，则可以使用较慢的更广泛可用的双列直插存储器模块(DIMM,dual in-line memory module)。相反，如果性能极为重要，则可以使用前沿技术DIMM。这种类型的存储器系统架构可以在每个存储器信道(memory channel)独立运行的系统中良好工作。然而，该方法在高可用性系统中不足。

[0006] 已经开发独立存储器冗余阵列(RAIM)系统来提高性能和/或增加存储系统的可用性。RAIM跨若干独立存储器模块分布数据，其中每个存储器模块包含一个或多个存储器设备。存在许多不同的已经开发的RAIM方案，每个具有不同的特征以及与其相关联的不同的利和弊。性能、可用性以及效用/效率(例如，实际保持消费者数据的存储器设备的百分比)或许是最重要的。必须仔细考虑与各种方案相关联的折中，因为一个属性中的提高经常可以导致另一个中的降低。RAIM系统的示例可以在例如2010年6月24日提交的名称为“使用虚拟ECC的解码的RAIM系统”的美国专利公开2011/0320918(其内容通过引用全文并入)中以及2010年6月24日提交的名称为“冗余存储器系统中的错误校正和检测”的美

国专利公开 2011/0320914（其内容通过引用全文并入）中找到。

[0007] 高可用性系统（诸如 RAIM 系统）可以包括在各种子系统中的多个时钟域。包括不同时钟域的子系统的高效集成提出建立同步定时、检测同步问题和恢复同步的多种挑战。

发明内容

[0008] 实施例包括用于双异步和同步存储器系统的方法、系统和计算机程序产品。用于双异步和同步存储器操作的系统包括存储器控制器和经由同步信道耦接到存储器控制器的存储器缓冲器芯片。存储器缓冲器芯片包括存储器缓冲器单元，配置为在嵌套域（nest domain）与存储器控制器同步通信；以及存储器缓冲器适配器，配置为在存储器域（memory domain）与至少一个存储器接口端口通信。至少一个存储器接口端口可操作为存取至少一个存储器设备。边界层连接到嵌套域和存储器域，其中边界层可配置为在嵌套和存储器域之间以同步传输模式（synchronous transfer mode）操作，并且在嵌套和存储器域之间以异步传输模式（synchronous transfer mode）操作。

[0009] 用于存储器子系统中的双异步和同步存储器操作的计算机系统实现方法，包括建立存储器控制器和存储器缓冲器芯片之间的同步信道。模式选择器基于存储器缓冲器芯片的操作模式，确定用于存储器缓冲器芯片的存储器域锁相环的参考时钟源。嵌套域锁相环的输出作为参考时钟源提供给基于同步的操作模式的存储器缓冲器芯片中的存储器区域锁相环。嵌套域锁相环可操作同步到存储器控制器的存储器控制器锁相环。单独的参考时钟独立于嵌套域锁相环作为参考时钟提供给基于异步的操作模式的存储器区域锁相环。

[0010] 提供一种用于同步双异步和同步存储器子系统的计算机程序产品。该计算机程序产品包括由处理电路可读并且存储由执行方法的处理电路执行的指令的有形存储介质。该方法包括通过经由同步信道耦接到存储器缓冲器芯片的存储器控制器初始化和校准同步信道的路线（lane）。通过存储器控制器对于同步信道计算帧来回行程等待时间（frame round trip latency）。基于计算的帧来回行程等待时间建立帧传输协议。通过存储器控制器对于存储器缓冲器芯片建立同步参考。存储器缓冲器芯片包括相对于存储器控制器的同步操作模式和异步操作模式。

附图说明

[0011] 在说明书完结时在权利要求中书特别地指出和清楚地请求看作实施例的主题。实施例的上述和其他特征和优点从以下结合附图的详细描述中是显然的，在附图中：

[0012] 图 1 描绘根据实施例的存储器系统；

[0013] 图 2 描绘根据实施例的平面配置的存储器子系统；

[0014] 图 3 描绘根据实施例的缓冲 DIMM 配置的存储器子系统；

[0015] 图 4 描绘根据实施例的具有双异步和同步存储器操作模式的存储器子系统；

[0016] 图 5 描绘根据实施例的存储器子系统信道和接口；

[0017] 图 6 描绘根据实施例的用于提供存储器子系统中的同步操作的处理流程；

[0018] 图 7 描绘根据实施例的用于在存储器子系统中的嵌套和存储器域之间建立对准的处理流程；

[0019] 图 8 描绘根据实施例的同步存储器子系统的定时图；以及

[0020] 图 9 图示根据实施例的计算机程序产品。

具体实施方式

[0021] 示例性实施例提供包括可配置存储器子系统的存储器系统，该可配置存储器子系统可以在异步或在完全同步操作模式下运行。存储器系统包括在嵌套域与存储器子系统同步通信的处理子系统。存储器子系统还包括相对于嵌套域可以被同步或者异步运行的存储器域。

[0022] 图 1 描绘可以是较大计算机系统结构的部分的示例存储器系统 100。控制处理器（CP）系统 102 是包括配置为与存储器控制单元（MCU）106 接口的至少一个处理器 104 的处理子系统。处理器 104 可以是处理来自系统控制器（未示出）的读取、写入和配置请求的多核处理器或模块。MCU106 包括也称为存储器控制器的同步存储器控制器（MCS，memory controller synchronous），其控制与用于存取存储器子系统 112 中的多个存储设备的多个信道 110 的通信。MCU106 和 MCS108 可以包括一个或多个处理电路，或者处理可以由或者结合处理器 104 执行。在图 1 的示例中，存在可以作为虚拟信道 111 支持并行存储器存取的 5 个信道 110。在实施例中，存储器系统 100 是 5 信道独立存储器冗余阵列（RAIM）系统，其中信道 110 中的 4 个提供对数据列和校验位存储器的存取，并且第 5 个信道提供对存储器子系统 112 中的 RAIM 奇偶位的存取。

[0023] 信道 110 的每个是包括下行总线 114 和上行总线 116 的同步信道。给定信道 110 的每个下行总线 114 可以包括数量不同于对应的上行总线 116 的路线或链路。在图 1 的示例中，每个下行总线 114 包括 n 个单向高速串行路线，并且每个上行总线 116 包括 m 个单向高速串行路线。命令和 / 或数据的帧可以作为分解为用于串行通信的各个路线的分组在信道 110 的每个上传送和接收。在实施例中，分组以约每秒 9.6 吉比特（Gbps）传送，并且每个传送路线按信道 110 串行传送 4 比特组。存储器子系统 112 按下行总线 114 的路线接收、去偏斜、去串行化每个 4 比特组，以按信道 110 重构来自 MCU106 的帧。同样地，存储器子系统 112 可以按信道 110、按上行总线 116 的路线，向 MCU106 传送作为 4 比特组的分组的帧。每个帧可以包括一个或多个分组，也称为传输分组。

[0024] CP 系统 102 还可以包括与处理器 104 接口的高速缓存器子系统 118。CP 系统 102 的高速缓存器子系统接口 112 提供到高速缓存器子系统 118 的通信接口。高速缓存器子系统接口 122 可以经由 MCU106 从存储器子系统 112 接收数据，以存储在高速缓存器子系统 118。

[0025] 图 2 描绘根据实施例在平面配置 200 中作为图 1 的存储器子系统 112 的实例的存储器子系统 112a 的示例。图 2 的示例仅描绘存储器子系统 112a 的一个信道 110 ；然而，将理解存储器子系统 112a 可以包括多个如图 2 中描绘的平面配置 200 的实例，例如 5 个实例。如图 2 所示，平面配置 200 包括经由连接器 206 连接到多个动态随机存取存储器（DRAM）设备 204 的存储器缓冲器芯片 202。DRAM 设备 204 可以组织为一个或多个双列直插存储器模块（DIMM）208 的横列（rank）。连接器 206 的每个耦接到双数据速率（DDR）端口 210，也称为存储器缓冲器芯片 202 的存储器接口端口 210，其中每个 DDR 端口 210 可以耦接到多于一个的连接器 206。在图 2 的示例中，存储器缓冲器芯片 202 包括 DDR 端口 210a、210b、210c 和 210d。DDR 端口 210a 和 210b 每个耦接到一对连接器 206 和共享存储器缓冲器适配器（MBA）

212a。DDR 端口 210c 和 210d 每个可以耦接到单个连接器 206 和共享存储器缓冲器适配器(MBA)212b。DDR 端口 210a-210d 是用于向 DRAM 设备 204 发出存储器命令和读取与写入存储器数据的 JEDEC 兼容存储器接口。

[0026] MBA212a 和 212b 包括用于管理对 DRAM 设备 204 的存取，以及控制定时、刷新、校准等的存储器控制逻辑。MBA212a 和 212b 可以并行操作，以便 DDR 端口 210a 或 210b 上的操作可以与 DDR 端口 210c 或 210d 上的操作并行执行。

[0027] 存储器缓冲器芯片 202 还包括接口 214，其配置为经由信道 110 与 MCU106 的对应接口 216 通信。在接口 214 和 216 之间建立同步通信。这样，包括存储器缓冲器单元(MCU)218 的存储器缓冲器芯片 202 的部分操作在与 CP 系统 102 的 MCS108 同步的嵌套域 220。边界层 222 划分存储器域 224 与嵌套域 220。MBA212a 和 212b 与 DDR 端口 210a-210d、以及 DRAM 设备 204 在存储器域 224。嵌套域 220 和存储器域 224 之间的定时关系是可配置的，以便存储器域 224 可以相对于嵌套域 220 异步操作，或者存储器域 224 可以相对于嵌套域 220 同步操作。边界层 222 可配置为在嵌套域 220 和存储器域 224 之间以同步传输模式和异步传输模式操作。存储器缓冲器芯片 202 还可以包括一个或多个多输入移位寄存器(MISR)226，如本文进一步描述。例如，MBA212a 可以包括一个或多个 MISR226a，并且 MBA212b 可以包括一个或多个 MISR226b。MISR226 的其他实例可以包括在存储器系统 100 中的其他地方。作为进一步的示例，一个或多个 MISR226 可以个别地或在横跨 MBU218 和 MBA212a 与 212b 的层级中和 / 或在 MCU106 中定位。

[0028] 边界层 222 是异步接口，该异步接口允许变化频率的不同 DIMM208 或 DRAM 设备 204 安装在存储器域 224 中，而不需要改变嵌套域 220 的频率。这允许 CP 系统 102 在存储器安装或升级期间保持完整，从而允许习惯配置中的更大灵活性。在异步传输模式下，握手协议可以用于跨嵌套和存储器域 220、224 之间的边界层传递命令和数据。在同步传输模式下，存储器域 224 的定时是调整为与嵌套域 220 对准的相位，以便嵌套和存储器域 220、224 的周期性对准出现在其中命令和数据可以跨过(cross)边界层 222 的对准周期(alignment cycle)。

[0029] 嵌套域 220 主要负责重构和解码源同步信道分组、应用任何必需的寻址翻译、执行一致性(coherency)动作(诸如，目录查找和高速缓存器存取)、以及将存储器分派到存储器域 224。存储器域 224 可以包括队列、调度程序、动态电源管理控制、用于校准 DDR 端口 210a-210d 的硬件引擎以及用于恢复和管理可校正和不可校正错误的维护、诊断和检测引擎。在嵌套或存储器域可以存在其他功能。例如，可以存在具有对应目录的嵌入 DRAM(eDRAM)的高速缓存器。如果对于一些应用创建高速缓存器并且其他实例不使用其，则可以存在通过将特定阵列电压(例如，VCS)连接到地的节电(power saving)。这些功能可以并入 MBU218，或者位于嵌套域 220 中的其他地方。存储器域 224 中的 MBA212a 和 212b 还以包括用以发起 DRAM 设备 204 的自治存储器操作的逻辑，诸如刷新和周期性校准序列，以便维持正确数据和信号完整性。

[0030] 图 3 描绘根据实施例的作为在缓冲 DIMM 配置 300 中图 1 的存储器子系统 112 的实例的存储器子系统 112b。缓冲 DIMM 配置 300 可以包括存储器子系统 112b 中的多个缓冲 DIMM302，例如缓冲 DIMM302 的 5 个或更多实例，其中单个缓冲 DIMM302 为了说明的目的描绘在图 3 中。缓冲 DIMM302 包括图 2 的存储器缓冲器芯片 202。如在图 2 的示例中，CP 系

统 102 中的 MCU106 的 MCS108 经由接口 216 在信道 110 上同步通信。在图 3 的示例中，信道 110 接口到连接器 304，例如插座，其耦接到缓冲 DIMM302 的连接器 306。连接器 306 和存储器缓冲器芯片 202 的接口 214 之间的信号路径启用接口 214 和 216 之间的同步通信。

[0031] 如在图 2 的示例中，如图 3 描绘的存储器缓冲器芯片 202 包括嵌套域 220 和存储器域 224。类似于图 2，存储器缓冲器芯片 202 可以包括一个或多个 MISR226，诸如 MBA212a 中的一个或多个 MISR226a 和 MBA212b 中的一个或多个 MISR226b。在图 3 的示例中，MBU218 跨边界层 222 从嵌套域 220 到存储器域 224 中的 MBA212a 和 / 或 MBA212b 传递命令。MBA212a 与 DDR 端口 210a 和 210b 接口，并且 MBA212b 与 DDR 端口 210c 和 210d 接口。不是如图 2 的平面配置 200 中与一个或多个 DIMM208 上的 DRAM 设备 204 接口，DDR 端口 210a-210d 可以直接与缓冲 DIMM302 上的 DRAM 设备 204 接口。

[0032] 存储器子系统 112b 还可以包括为电压轨（voltage rail）312 提供电压源的电源管理逻辑 310。电压轨 312 是为存储器缓冲器高速缓存器 314 供电的本地高速缓存器电压轨。存储器缓冲器高速缓存器 314 可以是 MBU218 的部分。电源选择器 316 可以用于确定电压轨 312 是来源于电源管理逻辑 310 还是连结（tie）到地 318。电压轨 312 可以在不使用存储器缓冲器高速缓存器 314 时连结到地 318，由此减少功率消耗。当使用存储器缓冲器高速缓存器 314 时，电源选择器 316 将电压轨 312 连结到电源管理逻辑 310 的电压供应。栅栏（fencing）和时钟选通也可以用于更好地隔离电压和时钟域。

[0033] 如可以参考图 2 和 3 所见，在实施例中可以支持多个存储器子系统配置。DRAM 设备 204 的变化大小和配置可以具有不同的地址格式要求，因为横列的数量和插槽、行、列、组、组群和 / 或端口的总体细节在实施例中可以跨不同的 DRAM 设备 204 变化。也可以实现各种堆叠架构（例如，3 晶片（die）堆叠或 3DS），其可以包括封装架构中主横列和从横列。DRAM 设备 204 的这些不同配置的每个可以要求唯一的地址映射表。因此，一般位可以由 MCU106 使用以引用 DRAM 设备 204 中的特别位，而不需具有实际 DRAM 拓扑的完整知识，从而将 DRAM 设备 204 的物理实现与 MCU106 分离。存储器缓冲器芯片 202 可以将一般位映射到附接到存储器缓冲器芯片 202 的（多个）特定类型的 DRAM 中的实际位置。一般位可以编程为保持任何适当的地址字段，取决于特定计算机系统，包括但不限于存储器基址、横列（包括主和从）、行、列、组、组群和 / 或端口。

[0034] 图 4 描绘根据实施例作为具有双异步和同步存储器操作模式的图 1 的存储器子系统 112 的实例的存储器子系统 112c。存储器子系统 112c 可以在图 2 的平面配置 200 或者图 3 的缓冲 DIMM 配置 300 中实现。如在图 2 和 3 的示例中，CP 系统 102 中的 MCU106 的 MCS108 经由接口 216 在信道 110 上同步通信。图 4 描绘接口 216 的多个实例作为接口 216a-216n，其配置为与存储器缓冲器芯片 202a-202n 的多个实例通信。在实施例中，每 CP 系统 102 存在 5 个存储器缓冲器芯片 202a-202n。

[0035] 如在图 2 和 3 的示例中，如图 4 中描绘的存储器缓冲器芯片 202 包括嵌套域 220 和存储器域 224。还类似于图 2 和 3，存储器缓冲器芯片 202a 可以包括一个或多个 MISR226，诸如 MBA212a 中的一个或多个 MISR226a 和 MBA212b 中的一个或多个 MISR226b。在图 4 的示例中，MBU218 跨边界层 222 从嵌套域 220 到存储器域 224 中的 MBA212a 和 / 或 MBA212b 传递命令。MBA212a 与 DDR 端口 210a 和 210b 接口，并且 MBA212b 与 DDR 端口 210c 和 210d 接口。使用锁相环（PLL）402、404 和 406 建立和维持嵌套域 220 和存储器域 224。

[0036] PLL402 是配置为向 CP 系统 102 的 MCU106 中的 MCS108 和接口 216a-216n 提供主时钟 408 的存储器控制器 PLL。PLL404 是耦接到存储器缓冲器芯片 202a 的 MBU218 和接口 214 以提供多个嵌套域时钟 405 的嵌套域 PLL。PLL406 是耦接到 MBA212a 和 212b 以及 DDR 端口 210a-210d 以提供多个存储器域时钟 407 的存储器域 PLL。PLL402 由参考时钟 410 驱动以建立主时钟 407。PLL404 具有用于同步到嵌套域 220 中的主时钟 405 的参考时钟 408。PLL406 可以使用单独参考时钟 414 或 PLL404 的输出 416 以提供参考时钟 418。单独的参考时钟 414 独立于 PLL404 操作。

[0037] 模式选择器 420 基于操作模式 422 确定参考时钟 418 的源，以使存储器域 224 相对于嵌套域 220 同步或者异步操作。当操作模式 422 是异步操作模式时，参考时钟 418 基于作为参考时钟源的参考时钟 414，以便 PLL406 由单独的参考时钟 414 驱动。当操作模式 422 是同步操作模式时，参考时钟 418 基于采用 PLL404 作为参考时钟源用于同步时钟对准的 FSYNC 块 492 的输出 416。这确保 PLL404 和 406 具有基于参考时钟 408 的有关时钟源。即使PLL404 和 406 可以在同步操作模式下同步，PLL404 和 406 也可以配置为以相对于彼此不同的频率操作。另外的倍频和频率导数（derivative）（诸如两倍速率、二分之一速率、四分之一速率等）可以基于 PLL402、404 和 406 的每个中的乘法器和除法器设置的每个生成。例如，嵌套域时钟 405 可以包括 PLL404 的第一频率的倍数，同时存储器域时钟 407 可以包括 PLL406 的第二时钟的倍数。

[0038] 在操作的异步模式中，每个存储器缓冲器芯片 202a-202n 分配给独立信道 110。用于个别高速缓存器线路（cache line）的所有数据可以自包含在附接到公共存储器缓冲器芯片 202 的图 2 和 3 的 DRAM 设备 204 中。该类型的结构适用于低端成本有效系统，其可以按照需求要求调节信道 110 的数量以及 DRAM 速度和容量。此外，该结构可以适用于高端系统，其采用诸如双信道 110 上的镜像存储器的特征以提供在信道停机（outage）的情况下的高可用性。

[0039] 当实现为 RAIM 系统时，存储器缓冲器芯片 202a-202n 可以配置在操作的同步模式下。在 RAIM 配置中，存储器数据跨多个物理存储器信道 110（例如，5 个信道）分条，其可以用作图 1 的单个虚拟信道 111 以便提供用于连续操作的错误校正码（ECC）保护，即使当整个信道 110 故障时。在 RAIM 配置中，相同虚拟信道 111 的所用存储器缓冲芯片 202a-202n 同步操作，因为每个存储器缓冲芯片 202 负责一致线路（coherent line）的部分。

[0040] 为了支持和维护同步操作，MCU106 可以检测一个信道 110 变得暂时或永久失能（incapacitated）的情形，从而导致该信道 110 相对于其他信道 110 丧失同步地（out of sync）操作。在许多情况下下面的情形是可恢复的，诸如一个或多个存储器缓冲器芯片 202a-202n 的接口 216a-216n 和 / 或接口 214 之一上的间歇传输错误。信道 110 上的通信可以利用传输上的鲁棒循环冗余码（CRC），其中检测到的 CRC 错误触发恢复重传序列。存在重传要求检测和重传之间的一些干预或延迟的情况。包括用于每个信道的重放缓冲器的重放系统可以用于支持对于故障信道 110 的恢复重传序列。可以暂停重放系统的部分可编程时间段，以确保要存储在重放缓冲器中的源数据在发起自动恢复之前已经存储。暂停重放时的时间段可以用于对其他子系统进行调整，诸如电压控制、时钟、调谐逻辑、电源控制等，这可以有助于防止导致故障的错误情况的重现。暂停重放也可以去除 MCU106 在故障信道 110 上重新发出存储的剩余部分的需求，并且可以提高重放时成功的潜力。

[0041] 虽然恢复重传序列最终可以将故障信道 110 还原到完全操作状态，但是总体存储器子系统 112 在恢复期间保持可用。容忍暂时的丧失同步条件允许存储器操作通过使用剩余的好(即，无故障)信道 110 来继续，直到恢复序列完成。例如，如果数据已经开始传送回图 1 的高速缓存器子系统 118，则在其已经发送之后可能需要处理故障数据的方式。虽然利用间隙返回数据是一种选项，但是另一种选项是延迟数据传输的开始直到知道所有的错误状态。当存在无间隙要求时延迟可以导致降低的性能。在恢复故障信道 110 之后，MCU106 将恢复的信道 110 与剩余的好信道 110 重新同步，从而跨图 1 的虚拟信道 111 的所有信道 110 重新建立完全功能接口。

[0042] 为了支持可以另外地使用去偏斜逻辑处理的定时对准问题，MCU106 和存储器缓冲器芯片 202 可以支持标签的使用。命令完成和数据目的地路由信息可以存储在使用接收的标签访问的标签目录 424 中。用于错误恢复的机制(包括读取或写入命令的重试)可以在用于每个个别信道 110 的存储器缓冲器芯片 202 中实现。由 MCU106 发出到存储器缓冲器芯片 202 的每个命令可以被分配 MCU106 中的命令标签，并且分配的命令标签随命令在各个信道 110 中发送到存储器缓冲器芯片 202。各个信道 110 发送回包括数据标签或完成标签的应答标签。对应于分配的命令标签的数据标签在每个信道中从缓冲器芯片返回，以将从各个信道 110 返回的读取数据与原始读取命令相关。对于分配的命令标签的完成标签也在每个信道 110 中从存储器缓冲器芯片 202 返回，以指示读取或写入命令完成。

[0043] 标签目录 424 也与可以包括数据标签表和完成标签表的标签表相关联，可以维持在 MCU106 中，以记录和检查返回的数据和完成标签。基于标签表确定何时所有与 MCU106 通信的当前工作信道返回对应于特定命令的标签。对于对应于读取命令的数据标签，当确定对应于读取命令的数据标签已经从当前工作信道 110 的每个接收时，认为读取的数据可用于传递到图 1 的高速缓存器子系统 118。对于对应于读取或写入命令的完成标签，当确定对应于读取或写入命令的完成标签已经从当前工作信道 110 的每个接收到时，从存储器控制单元和系统观点指示读取或写入为完成。MCU106 中的标签检查机制可以通过从信道 110 的列表移除永久故障的信道 110 以登记在标签表中来说明该信道 110。没有读取或写入命令需要维持在 MCU106 中用于重试命令、释放 MCU106 中的排队资源。

[0044] 支持高速同步通信的定时和信号调整也在用于信道 110 的接口等级管理。图 5 根据实施例更详细地描绘信道 110 和接口 214 和 216 的示例。如之前参考图 1 所述，每个信道 110 包括下行总线 114 和上行总线 116。下行总线 114 包括多个下行路线 502，其中每个路线 502 可以是差分串行信号路径以建立接口 216 的驱动器缓冲器 504 和接口 214 的接收器缓冲器 506 之间的通信。类似地，上行总线 116 包括多个上行路线 512，其中每个路线 512 可以是差分串行信号路径以建立接口 214 的驱动器缓冲器 514 和接口 216 的接收器缓冲器 516 之间的通信。在示例性实施例中，4 位的组 508 按帧在活动传送路线 502 的每个上串行传送，并且 4 位的组 510 按帧在活动传送路线 512 的每个上串行传送；然而，可以支持其他组大小。路线 502 和 512 可以是一般数据路线、时钟路线、备用路线或其他路线类型，其中一般数据路线可以发送命令、地址、标签、帧控制或数据位。

[0045] 在接口 216 中，命令和 / 或数据存储在传送先入先出(FIFO)缓冲器 518 中以作为帧 520 传送。帧 520 通过串行化器 522 串行化并且通过驱动器缓冲器 504 作为串行数据的组 508 在路线 502 上传送到接口 214。在接口 214 中，在接收器缓冲器 506 处接收的串行数

据通过去串行化器 524 去串行化并且在接收 FIFO 缓冲器 526 中被捕获，其中接收的帧 528 可以被分析和重构。当从接口 214 发送数据回接口 216 时，要传送的帧 530 存储在接口 214 的传送 FIFO 缓冲器 532 中，通过串行化器 534 串行化，并且通过驱动器缓冲器 514 作为串行数据的组 510 在路线 512 上传送到接口 216。在接口 216 中，在接收器缓冲器 516 处接收的串行数据通过去串行化器 536 去串行化，并且在接收 FIFO 缓冲器 538 中捕获，在接收 FIFO 缓冲器 538 中接收的帧 540 可以被分析和重构。

[0046] 接口 214 和 216 每个可以包括各自的训练逻辑 544 和 546 的实例，以配置接口 214 和 216。训练逻辑 544 和 546 训练下行总线 114 和上行总线 116 二者，以将源同步时钟与路线 502 和 512 上的传输适当对准。训练逻辑 544 和 546 也可以建立充分的数据眼（data eye）以确保成功的数据捕获。进一步的细节参考图 6 的过程 600 描述。

[0047] 图 6 描绘根据实施例的用于在存储器子系统中提供同步操作的过程 600。为了实现跨全部多个信道 110 的高可用性完全同步存储器操作，跨信道 110 采用初始化和同步化过程。该过程 600 参考图 1-5 的元件描述。

[0048] 在块 602，初始化和校准每个信道 110 的路线 502 和 512。训练逻辑 544 和 546 可以对驱动器缓冲器 504 和 514 执行阻抗校准。训练逻辑 544 和 546 还可以执行接收器缓冲器 506 和 516 的静态偏移校准和 / 或采样锁存器（未示出），以及随后的导线测试（wire testing）以检测信道 110 的传输介质中的缺陷。导线测试可以通过发送针对路线 502 和 512 检查时钟和数据路线差分对的两侧的导线连续性的慢模式（pattern）来执行。导线测试可以包括驱动简单重复模式，以设置相位旋转器采样点、同步串行化器 522 与去串行化器 524 以及串行化器 534 与去行串化器 536，并且执行基于路线的去偏斜。数据眼优化也可以通过发送更复杂的训练模式来执行，该训练模式还用作功能数据加扰模式。

[0049] 训练逻辑 544 和 546 可以使用复杂的训练模式来优化各种参数，诸如最终接收器偏移、最终接收器增益、峰化振幅、决定反馈均衡、最终相位旋转器调整、最终偏移校准、加扰器和解扰器同步、以及用于 FIF0518、526、532 和 538 的加载到卸载延迟调整。

[0050] 当检测到路线 502 和 512 中的任何不起作用的路线时，调用动态备用过程，以用对应下行总线 114 或上行总线 116 的可用备用路线代替不起作用 / 损坏路线。可以进行最终调整，以读取接收 FIFO 缓冲器 526 和 538 的数据 FIFO 卸载指针，从而确保充分的定时余量。

[0051] 在块 604，基于计算的帧来回行程等待时间建立帧传输协议。一旦信道 110 能够可靠地在两个方向上传送帧，对于解码帧建立参考开始点。为了建立与嵌套时钟 405 和主时钟 408 之间的公共参考的同步，通过训练逻辑 546 和 544 执行帧锁定序列。训练逻辑 546 可以通过在下行总线 114 上发送包括固定模式（诸如，所有）的帧到训练逻辑 544 来发起帧锁定序列。训练逻辑 544 锁定到下行总线 114 上接收的固定模式帧。然后训练逻辑 544 在上行总线 116 上发送该固定模式帧到训练逻辑 546。训练逻辑 546 锁定到上行总线 114 上接收的固定模式帧。训练逻辑 546 和 544 连续生成帧节拍（frame beat）。一旦完成帧锁定序列，则检测的帧开始参考点用作所有随后内部时钟域的对准标记。

[0052] 肯定的确收帧（acknowledgment frame）协议可以用于训练逻辑 544 和 546 确认返回传送侧的每个帧的接收的情况。这可以通过分配给每个传送帧的顺序处理标识符的使用完成。为了传送侧精确地预测返回确收，可以执行称为帧来回行程等待时间（FRTL）的另一训练序列，以说明信道 110 的传输介质中的传播延迟。

[0053] 在示例性实施例中，训练逻辑 546 下行地发出空分组，并且开始下行帧定时器。训练逻辑 544 用上行确收帧应答并且同步地开始上行来回行程定时器。当从训练逻辑 544 接收第一个上行确收帧时，训练逻辑 546 设置下行来回行程等待时间值。训练逻辑 546 响应于来自训练逻辑 544 的上行确收帧，在下行总线 114 上发送下行确收帧。当检测到下行确收帧时，训练逻辑 544 设置上行来回行程延迟值。训练逻辑 544 发出第二上行确收帧以关闭回路。此时，训练逻辑 544 进入信道互锁状态。训练逻辑 544 开始发出空闲帧直到对于训练逻辑 544 传送的第一空闲帧接收肯定的确收。训练逻辑 546 检测第二上行确收帧并且进入信道互锁状态。训练逻辑 546 开始发出空闲帧直到对于训练逻辑 546 传送的第一空闲帧接收肯定的确收。当接收肯定的确收时，训练逻辑 546 完成信道互锁并且允许正常的通信量流过信道 110。

[0054] 在块 606，对于多个存储器缓冲器芯片 202a-202n 建立公共同步参考。在完全同步的多信道结构的情况下，建立相对同步点，以确保即使当存储器缓冲器芯片 202a-202n 也正在生成其自身的自治刷新和校准操作时，在存储器缓冲器芯片 202a-202n 上以相同的方式执行从 CP 系统 102 发起的操作。可以通过锁定到每个存储器缓冲器芯片 202 中的嵌套和存储器域 220 和 224 之间的固定频率比率完成同步。在示例性实施例中，来自嵌套和存储器域 220 和 224 二者的 PLL404 和 406 互锁，以便它们具有固定的重复关系。这确保两个域在重复的间隔具有相同的边缘对准边界(例如，上升边缘对准)，其也对准到用于高速源同步接口 214 以及 MBU218 的帧解码和执行逻辑的潜在时钟(underlying clock)。跨越所有潜在时钟域的公共上升边缘称为对准或“黄金”参考周期。

[0055] 通过使用对准参考周期来管理存储器缓冲器芯片 202a-202n 中的所有执行和仲裁决定来达到多信道操作同步。由于在相同虚拟信道 111 中的所有存储器缓冲器芯片 202a-202n 具有相同的相对对准参考周期，所以所有它们的队列和仲裁器(未示出)在锁定步骤逻辑地保持。这导致跨越所有信道 110 的相同顺序的操作。即使信道 110 可以具有固有物理偏斜，并且每个存储器缓冲器芯片 202 相对于其他的存储器缓冲器芯片 202 在不同的绝对时间执行给定操作，公共对准参考周期也向横越嵌套和存储器域 220 和 224 之间的边界层 222 的信道操作的机会，提供内部生成的刷新和校准操作之中的保证的定时关闭和等效仲裁。

[0056] 如之前参考图 4 所述，每个存储器缓冲器芯片 202 包括两个分立的 PLL，PLL404 和 PLL406，用于驱动嵌套和存储器域 220 和 224 的潜在时钟 405 和 407。当操作在异步模式下时，每个 PLL404 和 406 具有不拥有彼此的固有相位关系的全异参考时钟输入 408 和 414。然而，当运行在同步模式下时，存储器 PLL406 变为具有模式选择器 420 的嵌套 PLL404 的从设备，所述模式选择器接管向存储器 PLL406 提供参考时钟 418 的功能，以便存储器域时钟 407 对准到公共对准参考点。公共外部参考时钟，主时钟 408，可以分发到相同的虚拟信道 111 中的所有存储器缓冲器芯片 202a-202n 的嵌套 PLL404。PLL404 可以配置到外部反馈模式以确保所有 PLL404 将其输出嵌套时钟 405 对准到公共存储器子系统参考点。该公共点由专用同步逻辑使用以将基于 PLL404 输出 416 的合适的参考时钟 418 驱动到存储器域 PLL406 并且达到对目标对准周期(即，“黄金”周期)的锁定。

[0057] 图 7 描绘根据实施例的用于在存储器子系统 112 中的嵌套和存储器域 220 和 224 之间建立对准的过程 700。参考图 1-6 的元件描述过程 700。过程 700 首先在嵌套域 220

随后在存储器域 224 中建立对准或“黄金周期。”存储器缓冲器芯片 202 的所有内部计数器和定时器通过过程 700 对准到对准周期。

[0058] 在块 702，嵌套域时钟 405 与来自块 604 的先前帧锁定的帧开始信号对准。嵌套域 220 可以对于嵌套域时钟 405 使用多时钟频率，例如来节电。可以使用较高速度的时钟来定义帧开始，并且这样，存在帧开始可以处于较慢速度的嵌套域时钟 405 的稍后相位的可能性。这会造成在对准周期上将不会执行帧解码的情形。为了避免这一点，如果需要，帧开始信号可以延迟一个或多个周期，以便其总是与较慢速度的嵌套域时钟 405 对准，从而边缘对准帧开始与嵌套域时钟 405。用于嵌套域时钟 405 的时钟对准可以由 PLL404 和 / 或另外的电路（未示出）管理。在块 704，存储器域时钟 407 断开并且存储器域 PLL406 置入旁路模式。

[0059] 在块 706，MCS108 向所有存储器缓冲器芯片 202a-202n 发出使用正常帧协议的超级同步（“超级同步”）命令。MCS108 可以采用匹配已建立的频率比率模数计数器（modulo counter），以便其仅在固定时段发出任何类型的同步命令。这从 MCS108 观点建立了用于整个存储器子系统 112 的主参考点。即使超级同步命令可以在不同的绝对时间到达存储器缓冲器芯片 202a-202n，每个存储器缓冲器芯片 202 也可以使用在其上该命令解码为内部对准周期的嵌套周期。因为存储器缓冲器芯片 202a-202n 之中的偏斜是固定的，所以每个存储器缓冲器芯片 202a-202n 上的对准周期将具有相同的固定偏斜。该偏斜转化为无误差情况下的固定操作偏斜。

[0060] 在块 708，可以作为模式选择器 420 的部分的存储器缓冲器芯片 202 的同步逻辑使用超级同步解码作为参考来触发驱动存储器域 PLL406 的参考时钟 418 的重新对准。超级同步解码转化为一周期脉冲信号 494，与将 FSYNC 块 492 中的模数计数器 496 复位为零的嵌套域时钟 405 同步。FSYNC 块 492 中的模数计数器 496 的时段设置为所有存储器和嵌套时钟频率的最小公倍数，嵌套时钟频率具有标记对应于 MCS108 先前建立的参考点的同步点的上升边缘。FSYNC 时钟 416 的上升边缘变为 PLL406 的参考时钟以建立存储器域时钟。通过使 PLL406 的较低频率输出回到外部反馈端口，嵌套时钟 405 和存储时钟 407 全都具有对准到主参考点的公共时钟边缘。因此，FSYNC 块 492 提供同步时钟对准逻辑。

[0061] 在块 710，存储器域 PLL406 脱离旁路模式，以锁定到基于 PLL404 的输出 416 的新参考时钟 418 而不是参考时钟 414。在块 712，存储器域时钟 407 回到打开。存储器域时钟 407 现在边缘对准到与嵌套域时钟 405 相同的对准参考周期。

[0062] 在块 714，规则的随后同步命令由 MCS108 在对准周期上发送。该同步命令可以用于复位管理内部存储器操作命令的生成、执行和仲裁的各种计数器、定时器和 MISR226。通过在对准周期上执行复位，所有存储器缓冲器芯片 202a-202n 启动它们各自具有相同逻辑参考点的内部定时器和计数器。如果一个存储器缓冲器芯片 202 上的仲裁器在特定对准周期上识别来自处理器发起的存储器操作和内部发起的命令二者的请求，则剩余的存储器缓冲器芯片 202 上的对应仲裁器将也明白相同的相对对准周期上的相同的请求。因此，所有存储器缓冲器芯片 202a-202n 将进行相同的仲裁决定并且维持相同的操作顺序。

[0063] 实施例可以提供在存储器缓冲器芯片 202 处内部生成的命令，以包括 DRAM 刷新命令、DDR 校准操作、动态电源管理、错误恢复、存储器诊断等。任何时候需要这些操作之一，其必须跨越进入嵌套域 220 并且经历与 MCS108 发起的同步操作相同的仲裁。在黄金周期

上执行仲裁以确保所有存储器缓冲器芯片 202 遵守相同的仲裁队列并且生成相同的结果。该结果在确保在每个存储器缓冲器芯片 202 中的定时和处理变化无效的黄金周期上跨边界层 222 分派。

[0064] 在正常的无错误情况下，将跨所有存储器缓冲器芯片 202a-202n 维持操作的顺序。然而，存在一个信道 110 可能丧失与其他信道 110 的同步的情形。一个这样的发生是在一个或多个接口 214 和 216 上间断传输错误的存在。示例性实施例包括基于硬件的恢复机制，其中在信道 110 上传送的所有帧保持在重放缓冲器中规定的时间段。该时间包含足够长的窗口以保证帧已经到达接收侧、已经针对错误被检查、以及指示没有错误传输的肯定的确收已经返回到发送者。一旦确定，从重放缓冲器中收回（retire）该帧。然而，在错误传输的情况下，如果错误是一次性事件，该帧与多个随后帧一起被自动重传或重放。在许多情况下，重放是足够的并且正常操作可以重新开始。在某些情况下，信道 110 的传输介质对于着手动态修复以用来自路线 502 或 512 的备用路线代替有缺陷的路线的点已经变得损坏。当完成该修复过程时，发送原始帧的重放并且正常操作可以再次重新开始。

[0065] 另一较不常见的发生可以是证明为导致存储器缓冲器芯片 202 中的内部错误的锁存器扰动的片上干扰。这可以导致一个存储器缓冲器芯片 202 不同于剩余的存储器缓冲器芯片 202 执行其操作的情形。尽管存储器系统 100 继续正确地操作，但是如果信道 110 没有相互步调一致地操作，则可以存在显著的性能劣化。在示例性实施例中，MISR226 监视和检测这样的情形。MISR226 接收源自从管理存储器缓冲器芯片 202 的同步操作的关键定时器和计数器的输入，诸如刷新开始、DDR 校准定时器、功率节流（power throttling）等。作为集体形成签名（signature）的位的组合接收对 MISR226 的输入。MISR226 的一个或多个位作为上行帧有效载荷的部分不断传送到 MCU106，其监视从存储器缓冲器芯片 202a-202n 的 MISR226 接收的位。信道 110 之间的物理偏斜的存在导致来自 MISR226 的位跨越信道 110 在不同的绝对时间到达。因此，并入学习过程以校准 MISR226 的检查为信道 110 中的导线延迟（wire delay）。

[0066] 在示例性实施例中，MCU106 中的 MISR 检测并入两个特别方面以便监视信道 110 的同步性。首先，MCU106 监视在上行总线 116 上从存储器缓冲器芯片 202a-202n 的每个接收的 MISR 位，并且在 MISR 位中见到的任何差别指示丧失同步情况。尽管这不造成数据完整性问题的任何风险，但是这负面地影响性能，因为 MCU106 可以引起等待整个高速缓存器线路存取跨越信道 110 完成的另外的等待时间。另一方面是监视与存储器操作相关联的处理序列标识符（即，标签），并且当操作完成时比较相关联的“数据”标签或“完成”标签。再一次，考虑信道 110 的偏斜以便执行精确比较。在一个示例中，该偏斜可以证明在多达 30 个周期的最快和最慢信道 110 之间的差别中。如果标签是 7 位宽，在 5 个信道 110 并且跨越信道 110 最大 30 周期的差别的情况下，这会典型地要求 5×7×30=1050 个锁存器来执行简单化的比较。可以存在等同于作为对准到帧之后大约 4 周期的偏斜的大约 40 位时间的一些情况。为了进一步减少锁存器的数量，MISR 可以并入在 MCU106 中以将标签编码到位流中，其随后流水线化以消除偏斜。通过比较跨越所有信道 110 的 MCU106 的 MISR 的输出，检测到的差别指示丧失顺序处理情况。

[0067] 在这些情形中的任一，受折磨的（afflicted）信道 110 相对于其他信道 110 至少可以暂时丧失同步或丧失顺序（out of order）操作。存储器子系统 112 的连续可用性可以

通过各种恢复和自愈机制提供。可以使用数据标签以便在丧失同步或丧失顺序的情况下，MCU106 继续起作用。每个读取命令可以包括允许 MCS108 处理在不同时间或甚至以不同顺序从不同信道 110 接收的数据传递的相关联数据标签。这允许即使在信道 110 丧失同步时的情形下也正常工作。

[0068] 对丧失同步的情况，分级 MISR226 的组可以用于累积用于任何有关同步事件的签名。有关同步事件的示例包括存储器刷新开始、周期性驱动器（ZQ）校准开始、周期性存储器校准开始、电源管理窗口开始、以及其他运行（run off）同步计数器的事件。来自校准定时器、刷新定时器等的一个或多个位可以用作对 MISR226 的输入，以提供可以有助于在 MCU106 验证跨信道同步的时间变化签名。可以在存在数据的速度匹配的需要的地方插入分级 MISR226。例如，在 MBA212a 和 MBU218 之间、在 MBA212a 和 MBU218 之间、在 MBU218 和上行总线 116 之间、以及在接口 216a-216n 和 MCS108 之间可能需要速度匹配。

[0069] 对于丧失顺序的情况，分段（staging）从每个信道 110 在帧中接收的每个标签可以用于对导线延迟去偏斜并且比较它们。每信道 110MISR 可以用于从在 MCU106 接收的标签创建签名位流，并且执行基于标签 / 签名的去偏斜而不是基于硬件锁存器的去偏斜。基于 7 位宽标签的先前示例，在 5 个信道 110 和跨越信道 110 的最大 30 周期差别的情况下，MISR 的使用将 1050 个锁存器减少到约 7×5+30×5=185 个锁存器，加上另外的支持锁存器。

[0070] 为了最小化性能影响，MCS108 尝试保持所有信道 110 步伐一致，这暗示所有命令以相同的顺序执行。当执行读取命令时，相关联的数据标签用于确定哪个数据对应于哪个命令。该方法还可以允许基于资源可用性或定时依赖性重新排序命令并且获得更好的性能。可以在保证所有信道 110 步伐一致的同时重新排序命令，以便重新排序跨不同信道 110 是相同的。在该情况下，标签可以用于将数据匹配到来自存储器的数据的请求者，不管在处理数据请求的同时命令顺序改变的事实。

[0071] 当传递已经开始时并且为了等待对于传递还没有出现的情况的恢复，可以执行标记信道 110 错误。数据一可用，来自存储器子系统 112 的数据块就可以递送到图 1 的高速缓存器子系统接口 122，而不等待完整的数据错误检测。该设计实现基于信道错误罕见的假设。数据一从所有信道 110 可用，数据就可以跨时钟域从 MCS108 异步发送到高速缓存器子系统接口 122，但是在对于所有帧数据错误检测完整之前。如果在数据块传递已经开始之后检测到数据错误，例如在单独的异步接口从 MCS108 向高速缓存器子系统接口 122 发送指示，以拦截正在进行中的数据块传递，并且使用冗余信道信息完成传递。强制定时要求，以确保拦截及时出现，从而防止损坏数据传播到图 1 的高速缓存器子系统 118。可以采用可编程向下计数计数器以强制定时要求。

[0072] 如果在到高速缓存器子系统 118 的块数据传递已经开始之前检测到数据错误，停止传递直到已经对于任何数据错误检查所有帧。假定错误很少发生，性能影响是最小的。这减少了信道冗余的使用，并且可以导致在存在 DRAM 设备 204 中先前存在错误的情况下避免可能的不可校正错误。

[0073] MCU106 还可以包括基于按命令类型或目的地的可配置延迟功能以延迟到上行元件（诸如，高速缓存器）的数据块传递，直到对于该块数据错误检测完成。命令或目的地信息可用于使得这样的选择作为对标签目录的输入。这可以选择性提高系统可靠性并且简化错误处理，同时最小化性能影响。

[0074] 为了支持其他同步问题，MCU106 可以在信道故障的情况下重新建立跨多个信道的同步，而不使潜在恢复机制的控制用在故障信道上。可编程静默序列增加地尝试通过停止在可编程时间间隔上的存储和其他下行命令来恢复信道同步。静默序列可以等待来自存储器缓冲器芯片 202a-202n 的完成指示并且注入跨所有信道 110 的同步命令以复位潜在的计数器、定时器、MISR226 和其他时间敏感电路到对准参考周期。如果故障信道 110 保持丧失同步，可以在计划控制下重试静默序列。在许多情况下，干扰的潜在根源原因可以自愈，从而导致先前故障的信道 110 被重新激活并与剩余的信道 110 重新同步。在极端错误情况下，静默和恢复序列无法恢复故障信道 110，并且故障信道 110 永久离线。在包括 5 个信道 110 的 RAIM 架构中，一个信道 110 的故障允许剩余的四个信道 110 以降低的保护等级操作。

[0075] 图 8 描绘根据实施例的同步存储器子系统的示例定时图 800。定时图 800 包括用于存储器缓冲器芯片 202 的多个信号的定时。在图 8 的示例中，图 4 的两个嵌套域时钟 405 描绘为较高速度嵌套域时钟频率 802 和较低速度嵌套域时钟频率 804。图 4 的两个存储器域时钟 407 在图 8 中描绘为较高速度存储器域时钟频率 806 和较低速度存储器域时钟频率 808。定时图 800 还描绘用于嵌套域流水线 810、边界层 812、参考计数器 814、存储器队列 816 和 DDR 端口 210 的 DDR 接口 818 的示例定时。在实施例中，较高速度嵌套域时钟频率 802 是约 2.4GHz，较低速度嵌套域时钟频率 804 是约 1.2GHz，较高速度存储器域时钟频率 806 是约 1.6GHz，并且较低速度存储器域时钟频率 808 是约 0.8GHz。

[0076] 时钟周期的重复模式作为用于较低速度嵌套域时钟频率 804 的周期“B”、“C”、“A”的序列描绘在图 8 中。周期 A 表示对准周期，其中存储器缓冲器芯片 202 中的其他时钟和定时器复位为与对准周期 A 的上升边缘对准。当接收超级同步命令时，较高和较低速度存储器域时钟频率 806 和 808 停止，并且基于在时钟同步窗口 820 之后导致对准的同步点重新启动。一旦实现对准，对准周期 A（也称为“黄金”周期）用作用于相同虚拟信道 111 中的所有存储器缓冲器芯片 202a-202n 的公共逻辑参考。命令和数据仅在对准周期 A 上跨过边界层 222。规则的同步命令可以用于复位每个储器缓冲器芯片 202a-202n 中的计数器和定时器，以便所有计数参考对准周期。

[0077] 在图 8 中，在时钟边缘 822，较高和较低速度嵌套域时钟频率 802 和 804、较高和较低速度存储器域时钟频率 806 和 808 以及嵌套域流水线 810 全部对准。嵌套域流水线 810 的同步命令在较高速度存储器域时钟频率 806 的时钟边缘 824 传递给边界层 812。在周期 B 的时钟边缘 826，在嵌套域流水线 810 中接收读取命令。在较高速度存储器域时钟频率 806 的时钟边缘 828，读取命令传递给边界层 812，参考计数器 814 开始计数 0，并且同步命令传递给存储器队列 816。在较高速度存储器域时钟频率 806 的时钟边缘 830，参考计数器 814 递增到 1，读取命令传递到存储器队列 816 和 DDR 接口 818。在与对准周期 A 对准的较高速度存储器域时钟频率 806 的时钟边缘 832，参考计数器 814 递增到 2，并且刷新命令在存储器队列 816 中排队。在嵌套域 220 和存储器域 224 的时钟和信号之间实现对准，用于跨越图 2 的边界层 222 发送命令和数据。

[0078] 所属技术领域的技术人员知道，本发明的各个方面可以实现为系统、方法或计算机程序产品。因此，本发明的各个方面可以具体实现为以下形式，即：完全的硬件实施方式、完全的软件实施方式（包括固件、驻留软件、微代码等），或硬件和软件方面结合的实施方式，这里可以统称为“电路”、“模块”或“系统”。此外，在一些实施例中，本发明的各个方面

还可以实现为在一个或多个计算机可读介质中的计算机程序产品的形式，该计算机可读介质中包含计算机可读的程序代码。

[0079] 可以采用一个或多个计算机可读介质的任意组合。计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件，或者任意以上的组合。计算机可读存储介质的更具体的例子(非穷举的列表)包括：具有一个或多个导线的电连接、便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本文件中，计算机可读存储介质可以是任何包含或存储程序的有形介质，该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0080] 现在参考图9，在一个示例中，计算机程序产品900例如包括一个或多个存储介质902以在其上存储计算机可读程序代码部件或逻辑904，来提供或便利本文描述的实施例的一个或多个方面，其中所述介质可以是有形的和/或非暂时的。

[0081] 当在有形介质(包括但不限于电子存储器模块(RAM)、快闪存储器、紧凑盘(CD)、DVD、磁带等)上创建和存储时，程序代码经常称为“计算机程序产品”。计算机程序产品介质由优选在计算机系统中的处理电路典型可读以供处理电路执行。可以使用例如编译程序或汇编程序来汇编指令创建这样的程序代码，以当执行指令时执行本发明的各方面。

[0082] 技术效果和益处包括可以在异步或在完全同步操作模式下运行的可配置存储器子系统。处理器子系统和存储器子系统的多个存储器缓冲器芯片之间的同步通信允许利用多种配置、包括平面配置和缓冲DIMM配置的同步和异步定时选项、使用通用存储器缓冲器芯片，设计高可靠性存储器系统。大量同步、对准、错误检测和恢复特征进一步增强示例性实施例中的可靠性和灵活性。

[0083] 在此使用的术语仅用于描述特定实施例的目的，并且不意图限制实施例。如在此使用的，单数形式“一”、“一个”和“这个”意图也包括复数形式，除非上下文中另外清楚地指示。将进一步理解术语“包括”和/或“包含”，当在本说明书中使用时，规定声明的特征、整数、步骤、操作、元件、和/或组件的存在，但是不排除一个或多个其他特征、整数、步骤、操作、元件、组件、和/或上述的组的存在或增加。

[0084] 权利要求中的所有部件或步骤加功能元件的对应结构、材料、动作和等效物意图包括用于与如具体请求保护的其他请求保护的元件组合执行功能的任何结构、材料、或动作。已经为了说明和描述的目的呈现本发明的说明书，但是不意图以公开的形式穷举或限制实施例。许多修改和变化对本领域的技术人员将是显然的，而不背离实施例的范围和精神。选择和描述实施例以便最好地说明原理和实践应用，并且能使本领域的技术人员理解具有各种修改的各种实施例，如适合于预期的特定使用。

[0085] 可以以一种或多种程序设计语言的任意组合来编写用于执行本发明操作的计算机程序代码，所述程序设计语言包括面向对象的程序设计语言—诸如Java、Smalltalk、C++等，还包括常规的过程式程序设计语言—诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户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。在涉及远程计算机的情形中，远程计算机可以通过任意种类的网络——包括局域网(LAN)

或广域网 (WAN)—连接到用户计算机，或者，可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0086] 下面将参照根据本发明实施例的方法、装置(系统)和计算机程序产品的流程图和/或框图描述本发明。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，都可以由计算机程序指令实现。这些计算机程序指令可以提供给通用计算机、专用计算机或其它可编程数据处理装置的处理器，从而生产出一种机器，使得这些计算机程序指令在通过计算机或其它可编程数据处理装置的处理器执行时，产生了实现流程图和/或框图中的一个或多个方框中规定的功能/动作的装置。

[0087] 也可以把这些计算机程序指令存储在计算机可读介质中，这些指令使得计算机、其它可编程数据处理装置、或其他设备以特定方式工作，从而，存储在计算机可读介质中的指令就产生出包括实现流程图和/或框图中的一个或多个方框中规定的功能/动作的指令的制造品(article of manufacture)。

[0088] 计算机程序指令还可以加载到计算机、其他可编程数据处理装置、或其他设备上以使得在计算机、其他可编程装置、或其他设备上执行一系列操作步骤从而产生计算机实现的过程，以便在计算机或其他编程装置上执行的指令提供用于实现流程图和/或框图的一个或多个框中规定的功能/动作的过程。

[0089] 附图中的流程图和框图显示了根据本发明的多个实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上，流程图或框图中的每个方框可以代表一个模块、程序段或代码的一部分，所述模块、程序段或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意，在有些作为替换的实现中，方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如，两个连续的方框实际上可以基本并行地执行，它们有时也可以按相反的顺序执行，这依所涉及的功能而定。也要注意的是，框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合，可以用执行规定的功能或动作的专用的基于硬件的系统来实现，或者可以用专用硬件与计算机指令的组合来实现。

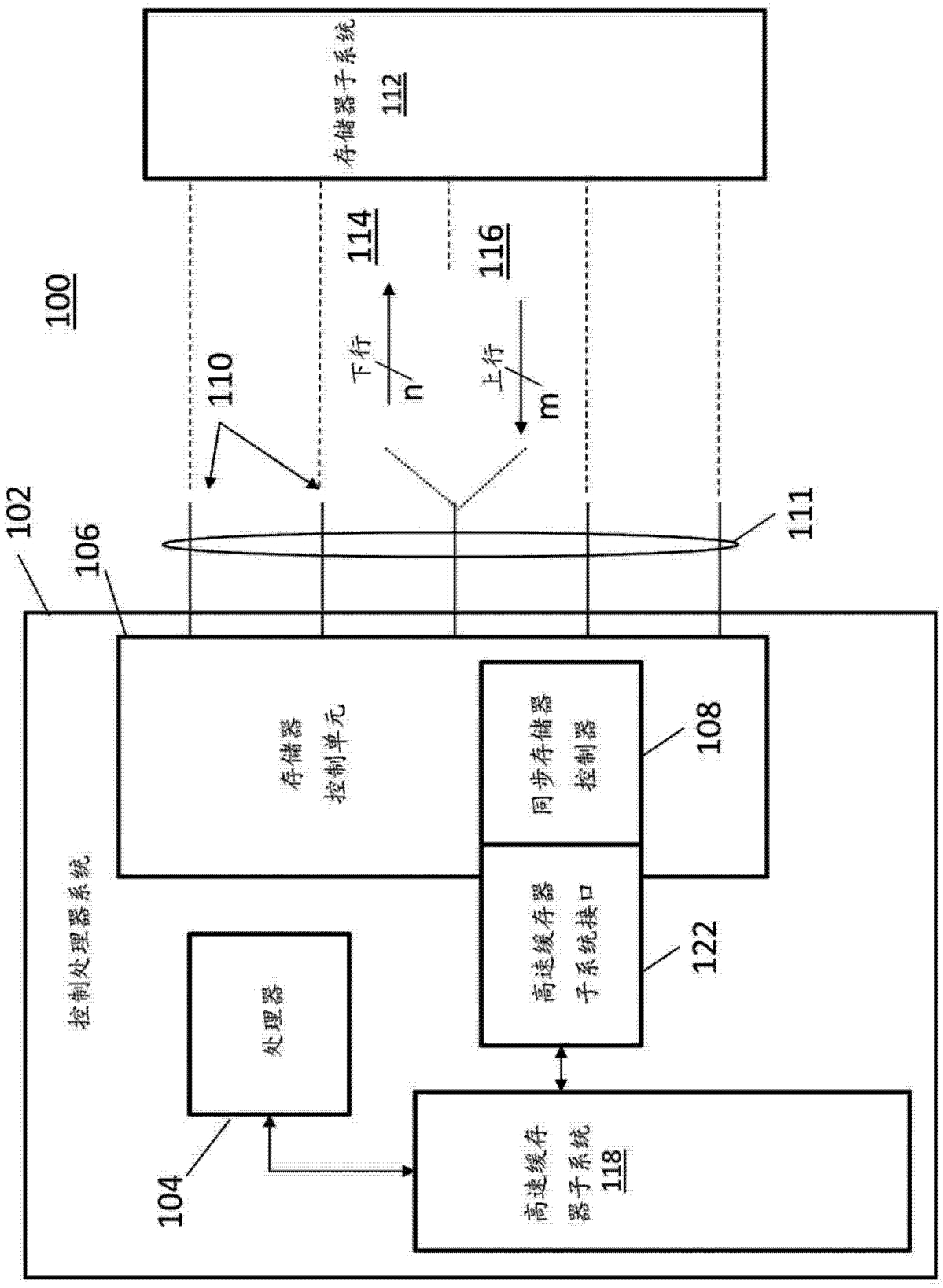

存储器子系统
112
100
114
116
下行
n
上行
m
110
102
106
111
存储器
控制单元
同步存储器
控制器
108
高速缓存器
子系统接口
122
控制处理器系统
处理器
104
高速缓存
器子系统
118

图 1

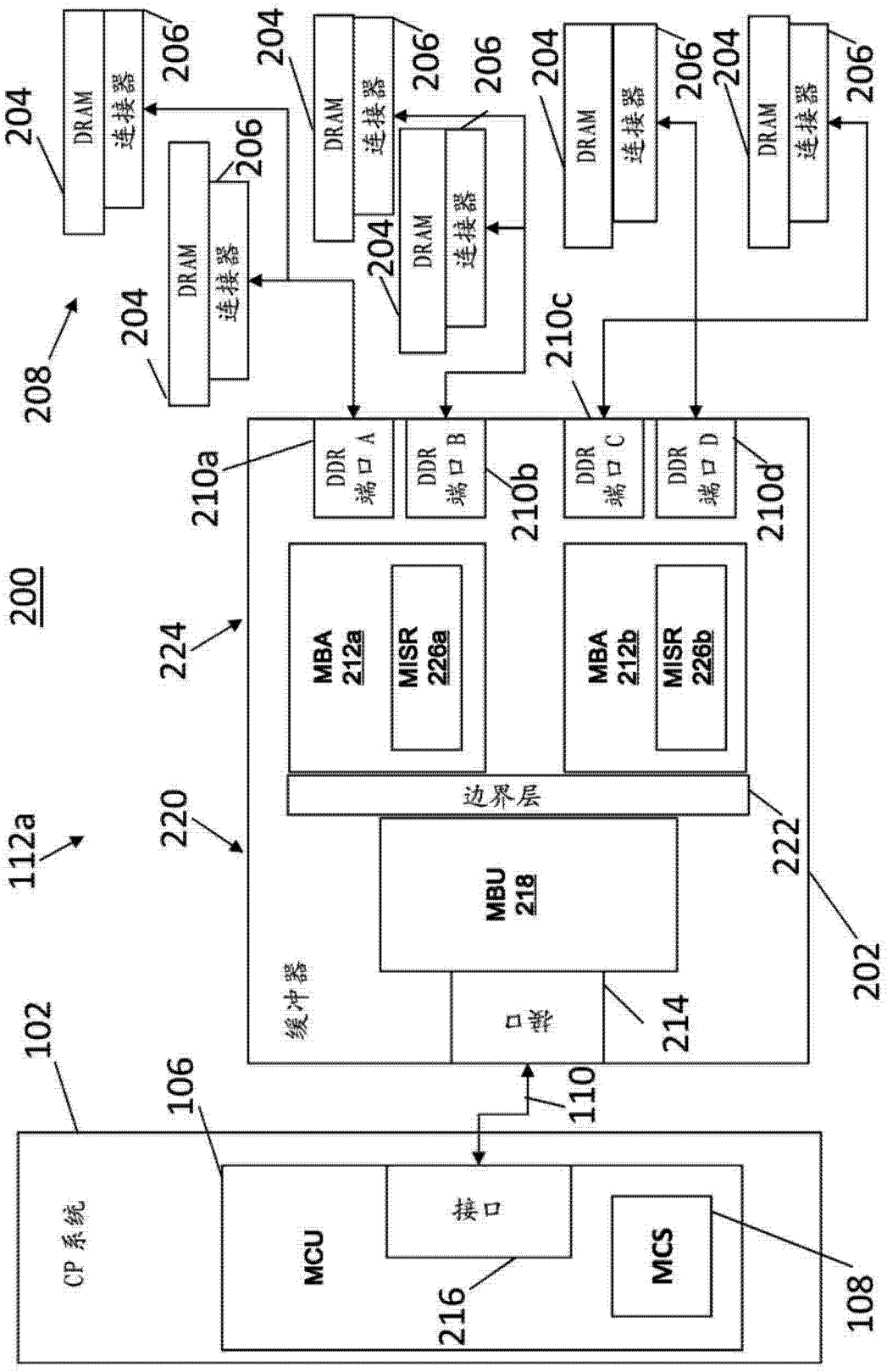

200
202
204
206
208
210a
210b
210c
210d
DRAM
连接器
DDR 端口 A
DDR 端口 B
DDR 端口 C
DDR 端口 D
MBA 212a
MISR 226a
MBA 212b
MISR 226b
边界层
MBU 218
缓冲器
端口
214
220
222
224
112a
102
106
110
CP 系统
MCU
接口
216
MCS
108

图 2

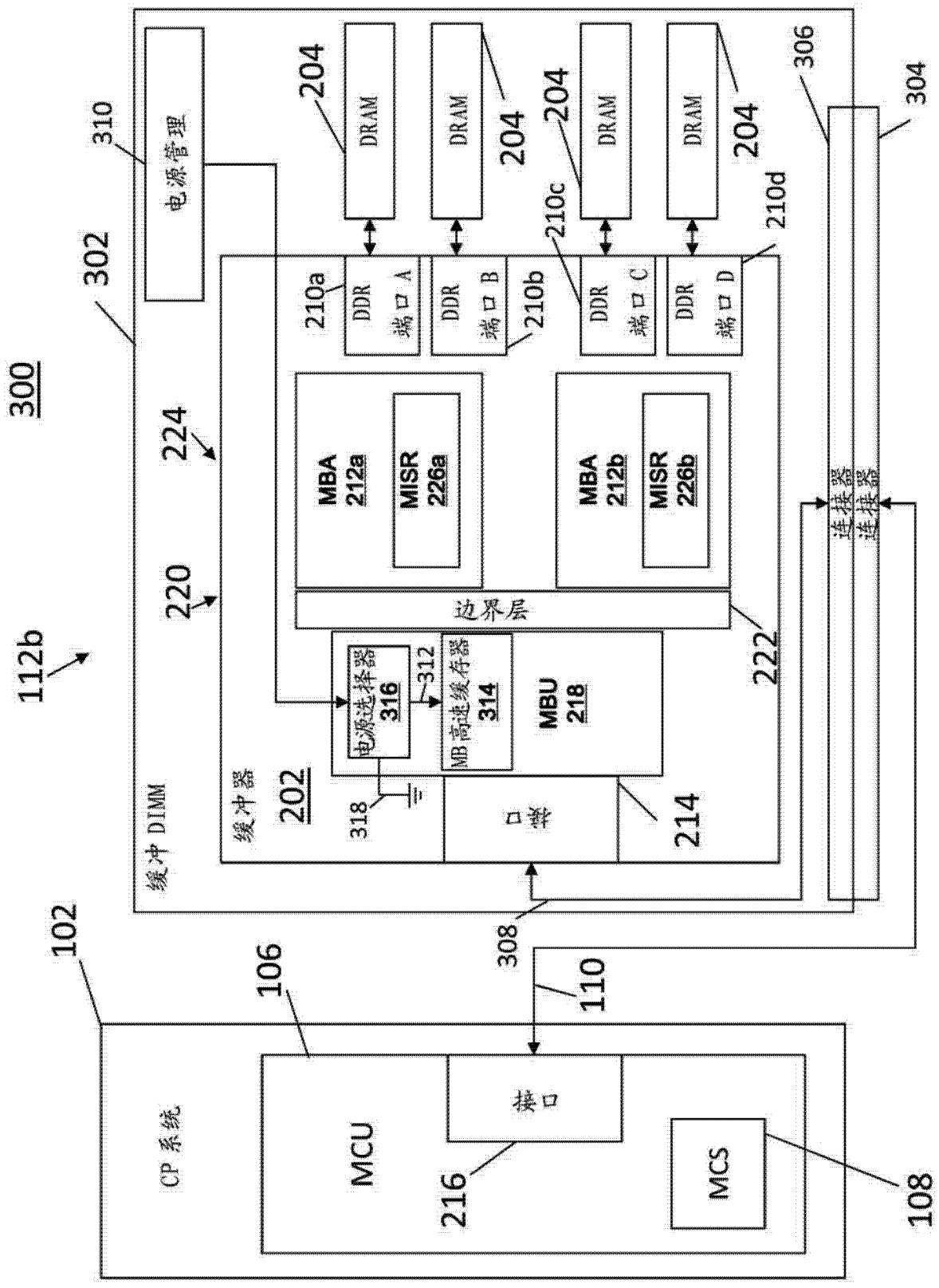

300
302
310
电源管理
204
DRAM
204
DRAM
204
DRAM
204
DRAM
306
304
210a
DDR 端口 A
210b
DDR 端口 B
210c
DDR 端口 C
210d
DDR 端口 D
224
MBA 212a
MISR 226a
MBA 212b
MISR 226b
220
边界层
222
112b
电源选择器 316
312
MB 高速缓存器 314
MBU 218
缓冲器 202
缓冲 DIMM
318
口群
214
连接器
连接器
102
106
308
110
CP 系统
MCU
接口
216
MCS
108

图 3

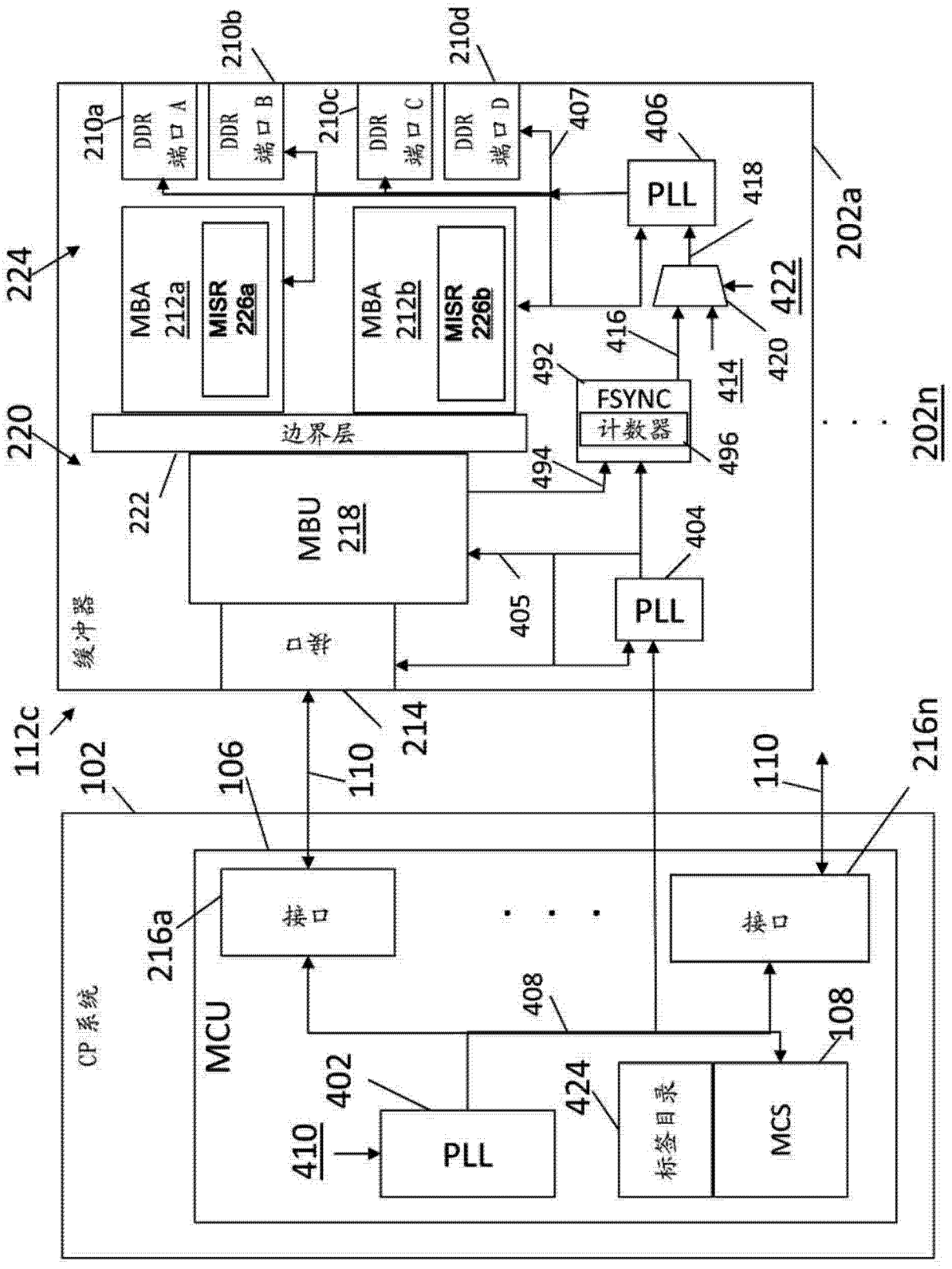

210b
210d
210a
DDR 端口 A
DDR 端口 B
210c
DDR 端口 C
DDR 端口 D
407
406
PLL
418
202a
224
MBA 212a
MISR 226a
MBA 212b
MISR 226b
422
416
420
414
492
FSYNC
计数器
496
220
边界层
494
222
MBU 218
404
PLL
405
缓冲器
接口
214
112c
102
106
110
110
216n
216a
接口
接口
CP 系统
MCU
408
108
402
424
410
PLL
标签目录
MCS

图 4

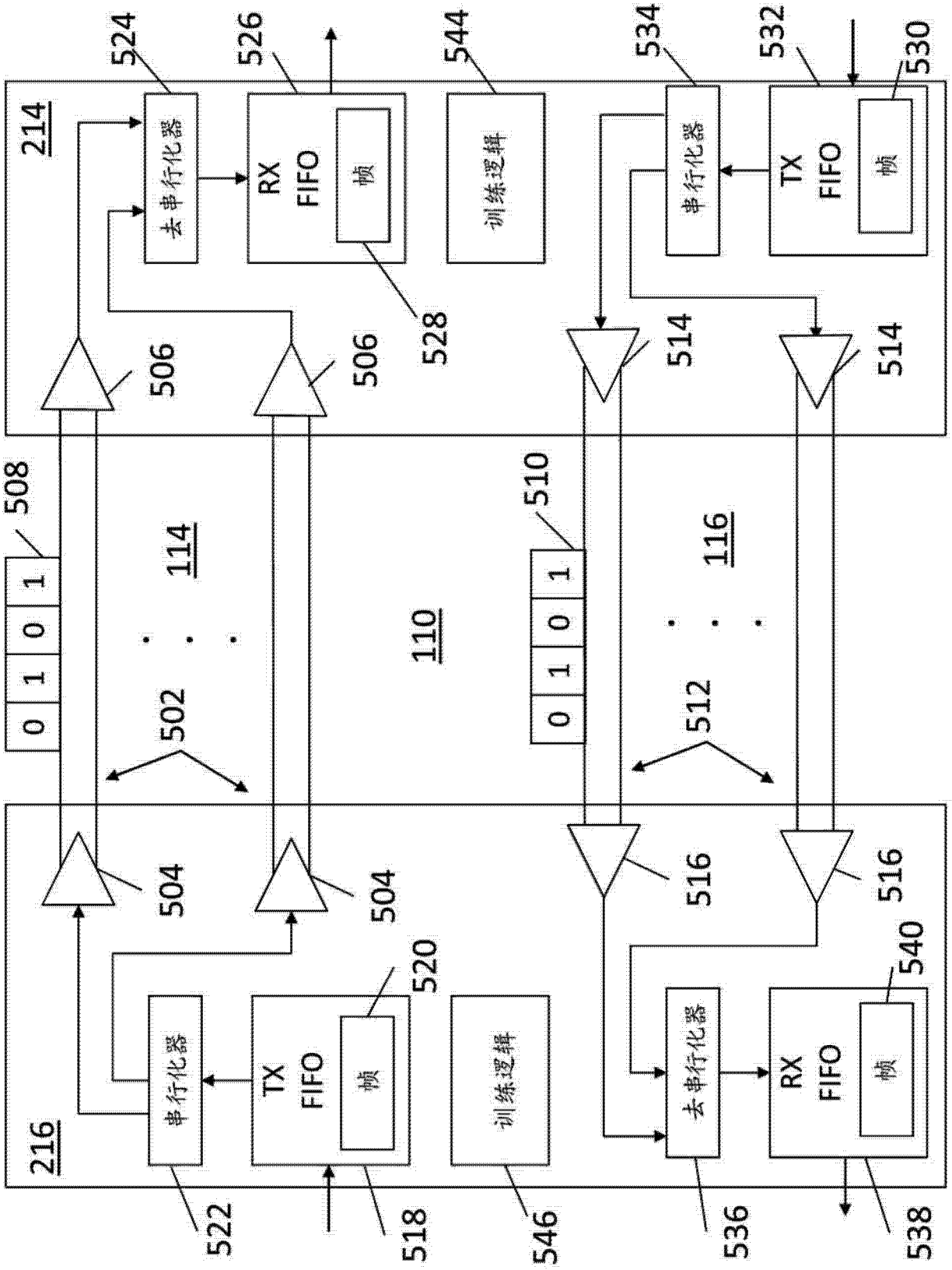

214
去串行化器
RX FIFO
帧
训练逻辑
串行化器
TX FIFO
帧
524
526
544
534
532
530
528
506
506
514
514
508
510
114
116
110
0 1 0 1
0 1 0 1
502
512
504
504
516
516
216
串行化器
TX FIFO
帧
训练逻辑
去串行化器
RX FIFO
帧
520
540
522
518
546
536
538

图 5

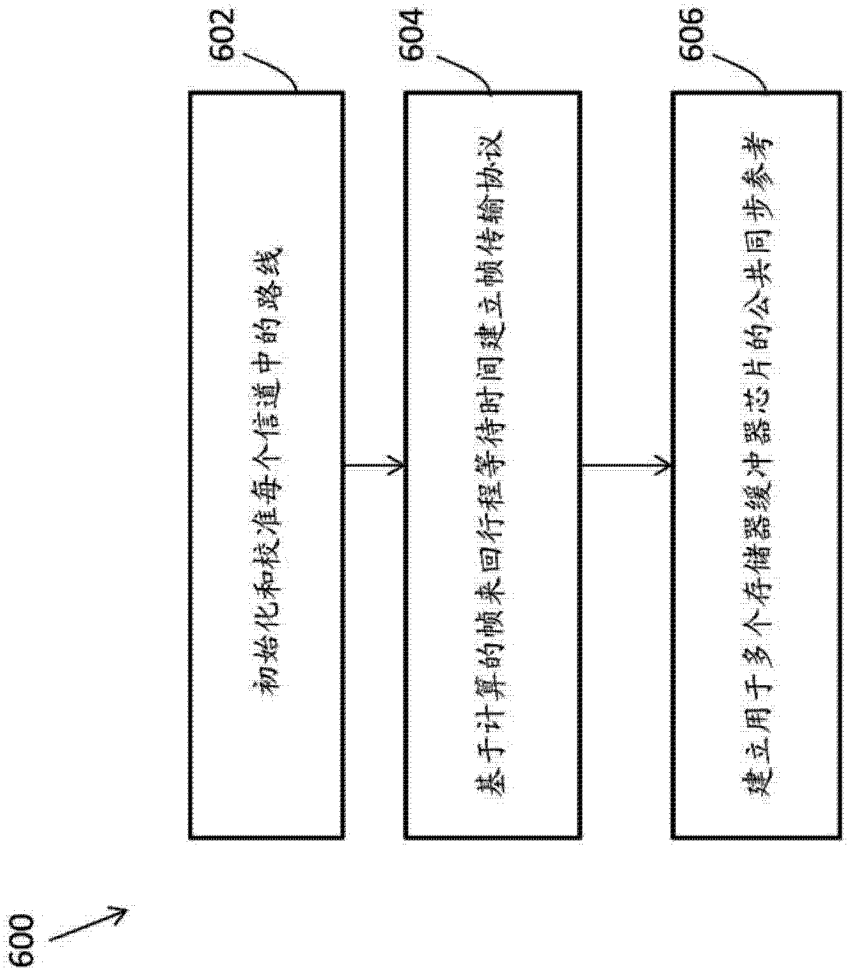

602
604
606
初始化和校准每个信道中的路线
基于计算的帧来回行程等待时间建立帧传输协议
建立用于多个存储器缓冲器芯片的公共同步参考
600

图 6

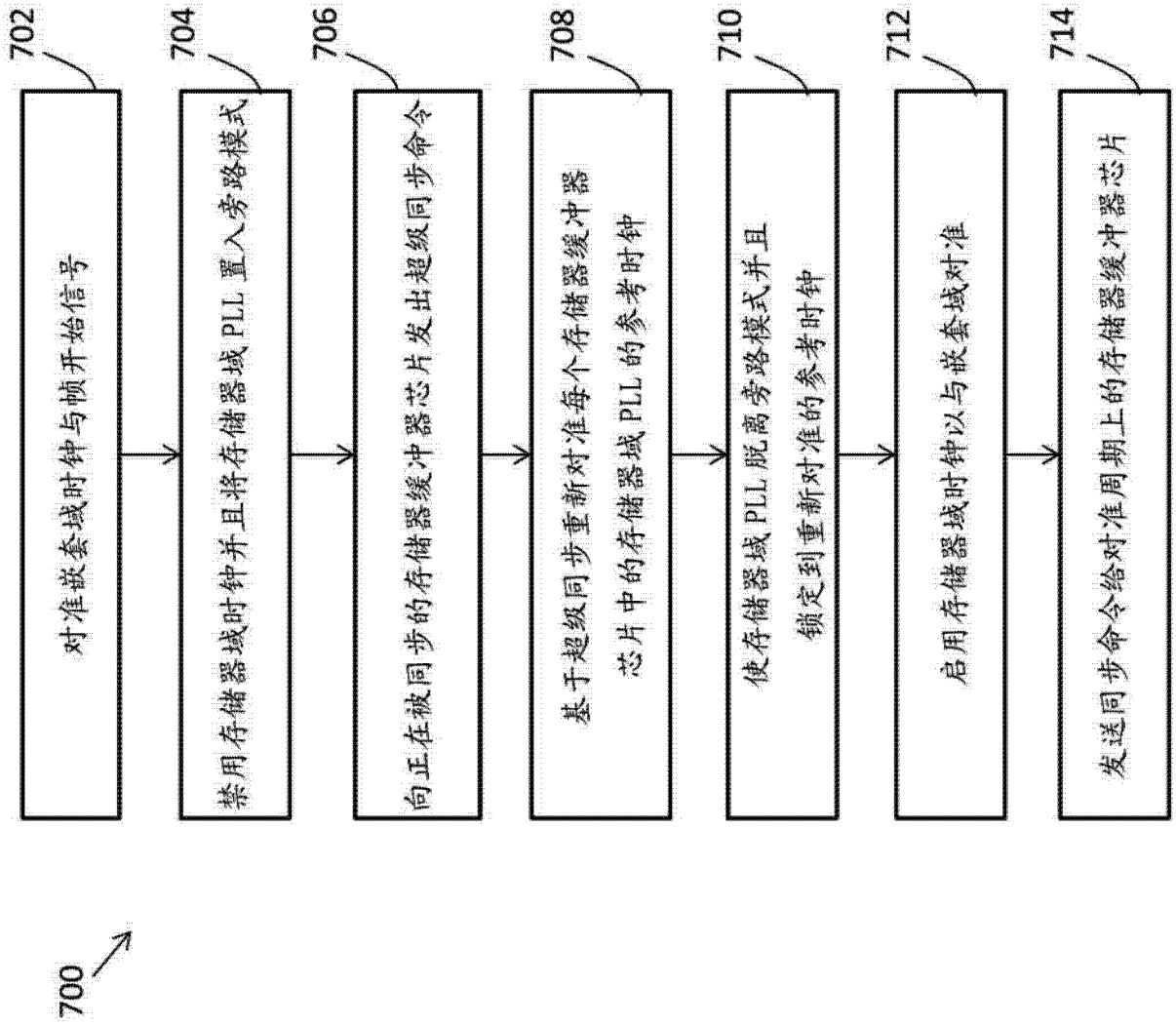

702
对准嵌套域时钟与帧开始信号
704
禁用存储器域时钟并且将存储器域 PLL 置入旁路模式
706
向正在被同步的存储器缓冲器芯片发出超级同步命令
708
基于超级同步重新对准每个存储器缓冲器芯片中的存储器域 PLL 的参考时钟
710
使存储器域 PLL 脱离旁路模式并且锁定到重新对准的参考时钟
712
启用存储器域时钟以与嵌套域对准
714
发送同步命令给对准周期上的存储器缓冲器芯片
700

图 7

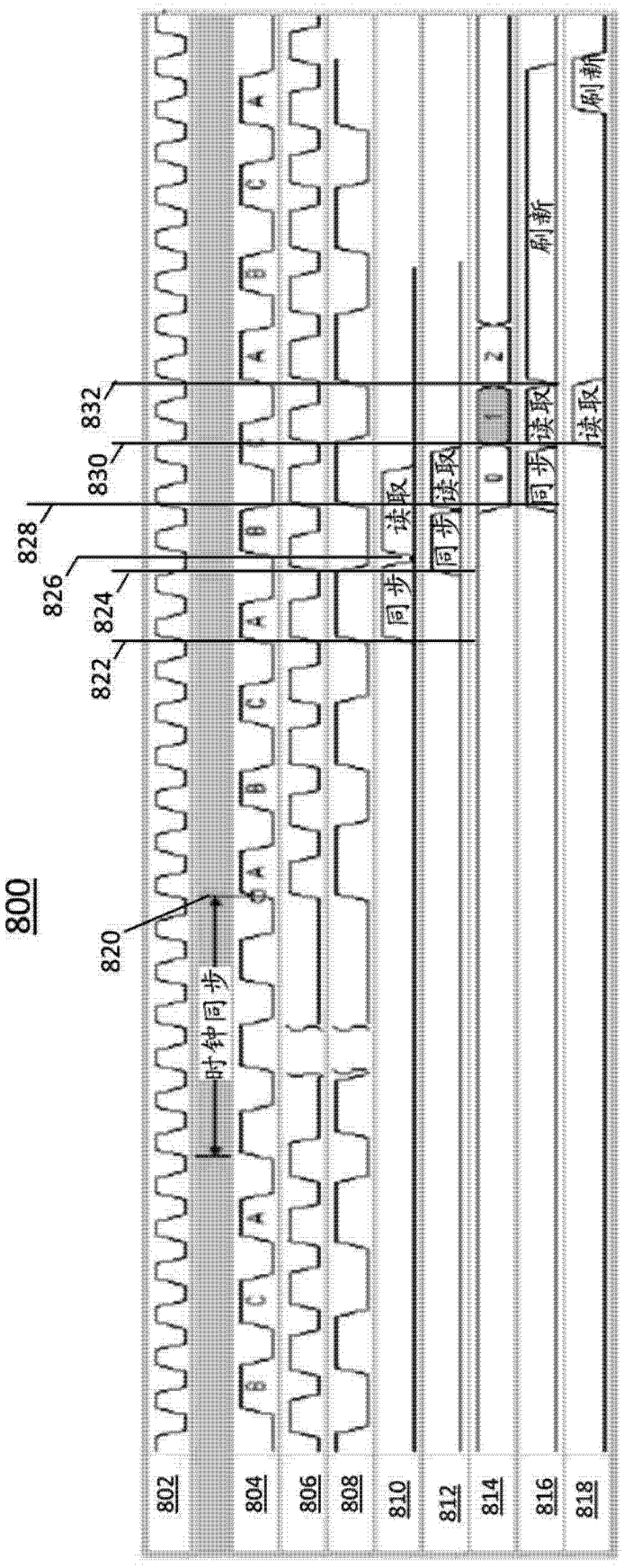

800
820
822
824
826
828
830
832
时钟同步
同步
读取
同步
读取
同步
读取
刷新
刷新
802
804
806
808
810
812
814
816
818

图 8

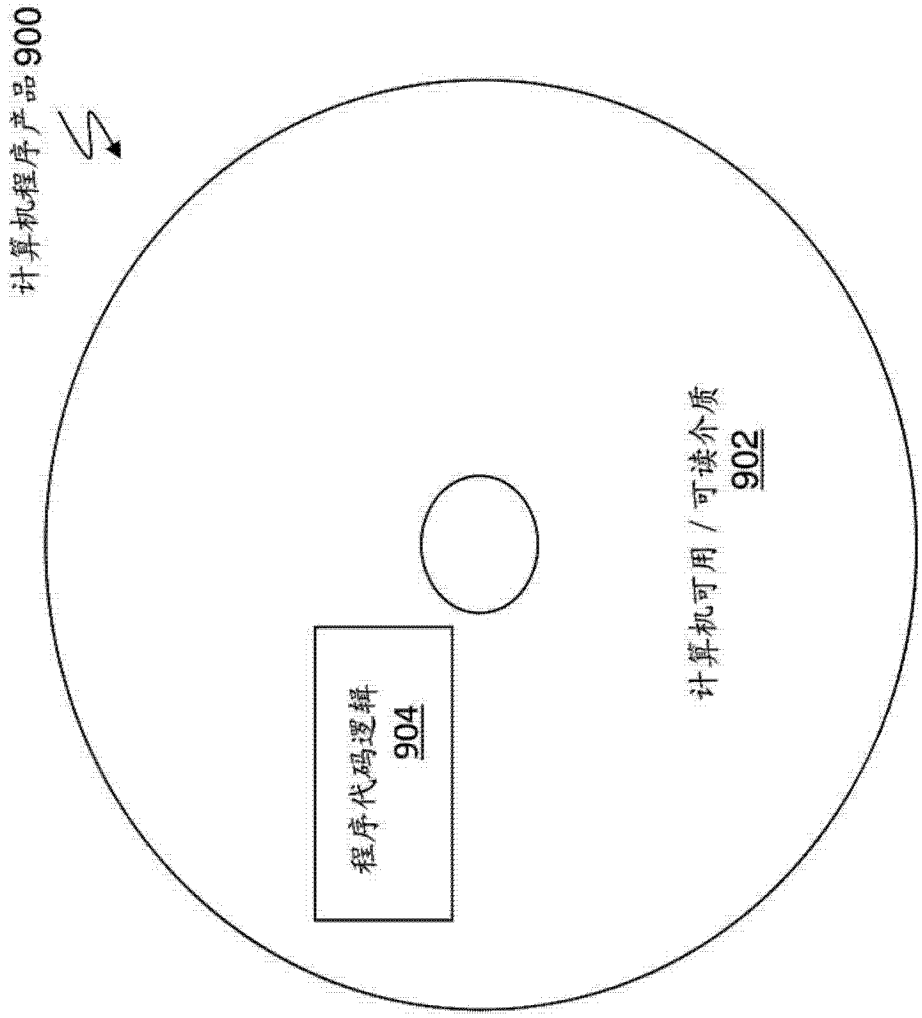


图 9