

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 13/16 (2006.01)

G06F 13/42 (2006.01)



# [12] 发明专利说明书

专利号 ZL 200480013182.0

[45] 授权公告日 2008 年 12 月 17 日

[11] 授权公告号 CN 100444141C

[22] 申请日 2004.5.10

[21] 申请号 200480013182.0

[30] 优先权

[32] 2003.5.13 [33] US [31] 60/470,078

[86] 国际申请 PCT/US2004/014441 2004.5.10

[87] 国际公布 WO2004/102403 英 2004.11.25

[85] 进入国家阶段日期 2005.11.14

[73] 专利权人 先进微装置公司

地址 美国加利福尼亚州

[72] 发明人 R·S·波尔青 F·D·韦伯

G·R·塔尔博特 L·D·休伊特

R·W·里夫斯 S·A·帕特尔

R·V·拉费特拉 D·E·久利克

M·D·赫梅尔 P·C·米兰达

[56] 参考文献

US5822766A 1998.10.13

WO0223353A2 2002.3.21

US4604689A 1986.8.5

审查员 孟宪超

[74] 专利代理机构 北京纪凯知识产权代理有限公司

代理人 戈泊程伟

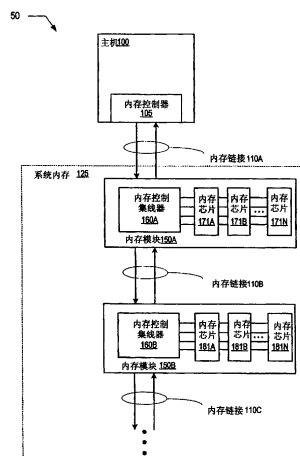
权利要求书 2 页 说明书 19 页 附图 8 页

[54] 发明名称

通过串行内存互连以将主机连接至内存模块的系统与方法

[57] 摘要

本发明公开了一种包括主机(100)的系统(50)，所述主机(100)耦接至串行连接的内存模块(150A-B)链路。在一个实施例中，各内存模块包括内存控制集线器(160)，用来控制内存模块上多个内存芯片(261)的存取。内存模块经由多个内存链接(110)在链路中串行耦接至主机。各内存链接可包括用于向主机传输事务处理的上行链接(211)，以及将源自主机的事务处理传输至链路中下一个内存模块的下行链接(212)。上行链接和下行链接可用信息包传输事务处理，所述信息包包括控制和配置信息包以及内存存取信息包。内存控制集线器可将第一内存链接的第一下行链接上所接收的事务处理在第二内存链接的第二下行链接上传输，而与译码该事务处理无关。



1. 一种通过串行内存互连以将主机(100)连接至多个内存模块(150A, 150B)的系统(50), 所述系统(50)的特征为:

所述内存模块(150A, 150B)经由多个内存链接(110A, 110B, 110C)而在链路中串行耦接至所述主机, 其中, 各内存链接包括用于向所述主机传输事务处理的上行链接(211A), 以及将源自所述主机的事务处理传输至所述链路中下一上内存模块的下行链接(212A), 其中, 所述多个内存模块的每一个包括内存控制集线器(160A, 160B), 所述内存控制集线器(160A, 160B)配置以控制对多个内存芯片(261A 至 N)的存取, 以及其中, 所述上行链接和所述下行链接各为单方向链接, 包括多个配置成用信息包来传输事务处理的信号, 而所述信息包包括控制和配置信息包以及内存存取信息包, 其中, 至少一部分信息包包括控制、地址和数据信息, 以及其中, 所述控制、地址和数据信息共享所给定上行链接或下行链接的相同线路。

2. 如权利要求 1 所述的系统, 其中, 配置所述内存控制集线器, 以接收第一内存链接的第一下行链接(212A)上的事务处理, 并在第二内存链接的第二下行链接(212B)上传输所述事务处理而与译码所述事务处理不相干。

3. 如权利要求 2 所述的系统, 其中, 所述内存控制集线器包括 DRAM 控制器(250), 且配置所述 DRAM 控制器(250), 以响应将所述事务处理译码成具有内存地址的内存命令而存取所述多个内存芯片, 而所述内存地址匹配与所述内存控制集线器相关的内存地址。

4. 如权利要求 1 所述的系统, 其中, 配置所述内存控制集线器, 以接收第一内存链接的第一上行链接上的事务处理并在第二内存链接的第二上行链接上传输所述事务处理, 以及其中, 所述内存控制集线器包括控制单元, 且配置所述控制单元以将局部事务处理选择性地注入到正在所述第二上行链接上传输的事务处理序列中。

5. 如权利要求 1 所述的系统, 其中, 所述主机包括内存控制器(105), 所述内存控制器(105)配置成发出内存请求事务处理, 而不用知道与所述各内存模块相关的内存大小或与所述任一个内存模块相关的地址范围。

6. 如权利要求 5 所述的系统, 其中, 所述内存控制器进一步配置成在接收对先前的内存读取请求事务处理的响应之前, 发出内存读取请求事务处理。

7. 一种通过串行内存互连以将主机(100)连接至第一内存模块(150A)与第二内存模块(150B)的方法, 所述方法的特征为以下步骤:

经由第一内存链接(110A)将所述主机(100)连接至所述第一内存模块(150A), 该第一内存链接包括上行链接以及下行链接; 以及

经由第二内存链接(110B)将所述第一内存模块连接至所述第二内存模块(150B), 该第二内存链接包括上行链接以及下行链接;

其中, 经由所述第一内存链接和所述第二内存链接将所述第一内存模块和所述第二内存模块串行连接至所述主机;

以及其中, 每一个所述上行链接和所述下行链接各为单方向链接, 包括多个配置成用信息包来传输事务处理的信号, 而所述信息包包括控制和配置信息包以及内存存取信息包, 其中, 至少一部份信息包包括控制、地址和数据信息, 以及其中, 所述控制、地址和数据信息共享所给定上行链接或下行链接的相同线路; 以及

其中, 所述第一内存模块配置成接收所述第一内存链接上的事务处理, 并在所述第二内存链接上传输所述事务处理而与译码所述事务处理不相干。

## 通过串行内存互连以将主机连接至内存模块 的系统与方法

### 技术领域

本发明涉及计算机系统内存，尤其涉及内存模块配置(memory module configurations)和内存子系统布局(subsystem topology)。

### 背景技术

许多计算机系统使用主系统内存，可依据终端用户的需要而配置。在这样的系统中，主板或系统板可包括许多内存扩充插口。可按需要将一个或多个小电路板，称之为内存模块，插入插口以增大计算机系统的内存容量。各内存模块一般包括多个提供指定内存容量的内存装置。内存装置通常用某些类型的动态随机存取存储器(DRAM)而实现。DRAM 类型的一些例子包括同步 DRAM (SDRAM)以及各种类型的双倍数据速率 SDRAM (DDR SDRAM)。

在常规的计算机系统中，内存模块经由内存总线连接至内存/DRAM 控制器，该内存总线包括地址、控制和数据信号。在一些计算机系统中，地址、控制和数据信号可被多路复用，而因此共享同一组线路。在其它的计算机系统中，地址、控制和数据信号可使用独立的线路。在其中任何一种情况下，各地址信号和控制信号路由(routed)至各扩充插口，而使得内存模块当被插入时是并行连接至内存/DRAM 控制器的。在某些系统中内存/DRAM 控制器可位于与系统处理器相同的集成电路(IC)芯片上，而在其它的系统中，内存/DRAM 控制器可位于芯片组的一个 IC 中（例如，北桥接器(Northbridge)）。

虽然计算机系统处理器的工作速度继续提高，但是主系统内存的相关性能并未以相同的速率提高。这也许至少部分是由于内存总线架构的带宽的上述增量改进。

### 发明内容

本发明公开了包括多个内存模块的系统的各种实施例。在一个实施例中，主机耦接到(coupled)串行连接的内存模块链路(chain)。

在一个特定的实施中，各内存模块可包括内存控制集线器(hub)，该内存控制集线器可控制对内存模块上的多个内存芯片的存取。此外，内存模块经由多个内存链接(link)在链路中串行耦接至主机。各内存链接可包括用于向主机传输事务处理(conveying transaction)的上行链接(uplink)，以及用于将源自主机的事务处理传输到链路中下一个内存模块的下行链接(downlink)。上行链接和下行链接可以各是包括多个信号的单方向链接，该包括多个信号的单方向链接可用信息包(packets)来传输事务处理，而该信息包包括控制和配置信息包以及内存存取信息包。

在另一个特定的实施中，内存控制集线器可将第一内存链接的第一下行链接上所接收到的事务处理译码(decode)。此外，内存控制集线器可在第二内存链接的第二下行链接上传输该事务处理，而与译码该事务处理不相干(independent of decoding the transaction)。

在又一个特定的实施中，内存控制集线器可包括 DRAM 控制器，该 DRAM 控制器可响应将事务处理译码成内存命令而存取内存芯片，而该内存命令具有与内存控制集线器的相关内存地址相匹配的内存地址。

## 附图说明

图 1 为包括串行连接的内存模块链路的系统的一个实施例的方块图。

图 2 为如图 1 中所示内存模块的内存模块的一个实施例的方块图。

图 3 为如图 2 中所示下行链接控制单元的下行链接控制单元的一个实施例的方块图。

图 4 为如图 2 中所示上行链接控制单元的上行链接控制单元的一个实施例的方块图。

图 5A 为配置读取信息包的一个实施例的图。

图 5B 为配置写入信息包的一个实施例的图。

图 5C 为内存读取信息包的一个实施例的图。

图 5D 为内存写入信息包的一个实施例的图。

图 6 为计算机系统的一个实施例的方块图。

虽然本发明易于有各种修改和替代形式，但是本发明的特定实施例在附图中用举例的方式来示意并在此详细说明。然而，应了解，并不欲用附图和详细说明来限制本发明于所公开的特定形式，相反，本发明意在涵盖所有落入由所附权利要求所界定的本发明的精神和范围内的修改、等效和替代。应注意的是，标题的目的仅用来组织内容，而并非欲用来限制或解释说明书或权利要求。再者，要注意的是，词汇“可以(may)”是以容许的意义（即，具有潜在性、能够）而不是以强制的意义（即，必须）而使用于整个本申请。词汇“包括(include)”及其派生意味着“包括，但不局限于”。词汇“连接(connected)”意味着“直接连接或间接连接”，而词汇“耦接(coupled)”意味着“直接耦接或间接耦接”。

### 具体实施方式

现在参照图 1，显示了包括串行连接的内存模块链路的系统的一个实施例的方块图。系统 50 包括经由内存链接 110A 而耦接至系统内存 125 的主机 100。例如，可配置系统 50 以作为诸如计算机系统或服务器系统的计算装置的一部分而工作。系统内存 125 包括经由内存链接 110B 而耦接至内存模块 150B 的内存模块 150A。显示出内存模块 150B 耦接至内存链接 110C，该内存链接 110C 可按需要耦接至额外的内存模块（图中未显示）以形成耦接至主机 100 的串行连接的内存模块链路。值得注意的是，虽然在链路中显示了两个内存模块，但是预期可用这种方式来连接一个或多个内存模块。应进一步注意的是，包括其后跟有参考字母的参考数字的组件一般地可单独由该参考数字而指代。例如，当一般地指代所有的内存模块时，可用内存模块 150 来表示。

在所示意的实施例中，内存模块 150A 包括内存控制集线器 160A，该内存控制集线器 160A 耦接至指定为内存芯片 171A 至 171N 的多个内存装置，此处 N 可以是所希望的任何数。在一个实施例中，内存控制集线器 160A 可以经由任意类型的内存互连(memory interconnect)而耦接至内存芯片。例如，在一个实施例中，内存互连可以是典型的地

址、控制和数据总线配置。

类似地，内存模块 150B 包括内存控制集线器 160B，该内存控制集线器 160B 耦接至指定为内存芯片 181A 至 181N 的多个内存装置，此处 N 可以是所希望的任何数。在一个实施例中，内存控制集线器 160B 可以如上所述经由任何类型的内存互连而耦接至内存芯片。值得注意的是，各内存芯片 171A 至 171N 和 181A 至 181N 例如可以是任何类型的内存装置，诸如内存装置的 DRAM 族中的内存装置。

在所示意的实施例中，内存链接 110A-110C 形成内存互连。在一个实施例中，内存链接 110A-110C 的每一个形成由两组单方向线所实施的点对点内存互连。一组单方向线被称为下行链接并被配置成沿下游(downstream)方向将事务处理输送离开主机 100。另一组单方向线被称为上行链接并被配置成沿上游(upstream)方向朝主机 100 输送事务处理。此外，在一个实施例中，各组单方向线可用多个微分信号对(differential signal pairs)来实现。在一个实施例中，各内存链接 110 包括 18 位下行链接和 16 位上行链接，其中各位为微分信号对。如下文中将结合图 5A 至图 5D 的说明而作更详细的说明，可配置由内存链接 110 所形成的内存互连以传输信息包。

一般而言，来自主机 100 的所有事务处理向下游流动经过下行链接上的所有内存模块 150，而所有响应事务处理从响应的内存模块 150 向上游流动，经过上行链接上的各上游内存模块 150。更具体地，在一个实施例中，主机 100 可请求寻回(retrieve)或储存系统内存 125 内的数据。响应于主机 100 发出请求，内存控制器 105 例如发起诸如内存读取事务处理或内存写入事务处理的相应事务处理。内存控制器 105 经由内存链接 110A 将该事务处理发送至系统内存 125。在所示意的实施例中，该事务处理由内存模块 150A 的内存控制集线器 160A 所接收。

响应于接收事务处理，配置内存控制集线器 160A 将所接收的事务处理经由内存链接 110B 发送至内存模块 150B，而并不译码该事务处理。这被称为向下游转发(forwarding)事务处理。因此，由给定内存模块 150 的给定内存控制集线器 160 在下行链接上所接收的各事务处理被转发至链路上的下一个内存模块 150，而并不译码该事务处理，该链路上的下一个内存模块 150 耦接至下行链接。在一个实施例中，译码

事务处理可与转发事务处理并行发生。在其它的实施例中，译码事务处理可在已转发了事务处理后发生。可在下文图 3 的说明中看到下游转发功能的更详细说明。

同样地，若内存控制器 105 发起例如读取请求事务处理，则具有相应于请求中的地址的内存位置的内存模块 150 将以所请求的数据响应。该响应将在内存模块的上行链接上朝向主机 100 发送。若在发送内存模块和主机 100 之间有任何中间(intervening)内存模块，则中间内存模块将在其上行链接上沿上游方向将响应事务处理转发至主机 100 或链路中的下一个内存模块。此外，当响应内存模块准备发送响应时，它可将该响应注入到上行链接上正向上游转发的事务处理序列中。在下文图 5 的说明中，可看到上游转发功能的更详细说明。

在一个实施例中，可配置内存控制器 105 以向系统内存 125 提出请求，而不用知道特定地址与内存模块 150A 和 150B 的哪一个相关。举例而言，在系统配置序列期间，可给各内存模块 150 分配内存地址的一个范围。各内存控制集线器 160 可包括逻辑（图 1 中未显示），该逻辑可译码输入请求(incoming request)的地址。因此，给定内存模块 150 的内存控制集线器 160 可响应于对具有在分配给该给定内存模块 150 的地址范围内的地址的内存请求译码，而发起内存读取循环或内存写入循环至该给定内存模块 150 上的内存芯片。如将在下文中结合图 2 的说明而更详细说明的那样，在一个实施例中，各内存控制集线器 160 可包括 DRAM 控制器(图 1 中未显示)，用于发起内存循环至该 DRAM 控制器所连接到的内存芯片。

此外，在一个实施例中，内存控制器 105 在接收对先前的内存存取请求的响应之前，可发起后续的内存存取请求。在这样的实施例中，内存控制器 105 可保持追踪待处理的请求(outstanding request)，并可因此以不同于响应被送出时的次序来处理响应。

值得注意的是，在一替代实施例中，各内存链接 110 可形成点对点内存互连，该点对点内存互连实施成一组双向线。这样，事务处理在该组双向线上既可向上游流动又可向下游流动。在这样的实施例中，可用多个微分信号对来实现双向线。

参照图 2，显示了诸如图 1 中所示内存模块的内存模块的一个实施



例的方块图。为了清楚和简明的目的，与图 1 中所示组件相对应的组件被相同地标号。内存模块 150 包括经由内存总线 265 而耦接到内存芯片 261A 至 261N 的内存控制集线器 160。内存控制集线器 160 包括耦接到 DRAM 控制器 250 的控制单元 240。DRAM 控制器 250 耦接至内存芯片 261A-261N。控制单元 240 包括上行链接控制 241 和下行链接控制 242。如上所述，内存总线 265 可以是任何类型的内存互连。在所示意的实施例中，内存控制集线器 160 沿上游方向耦接至内存链接 110A，且沿下游方向耦接至内存链接 110B。进一步注意到，内存总线 265 的工作频率与内存链接 110 的工作频率不相干。

在所示意的实施例中，可配置上行链接控制单元 241 以接收和转发从下游的另一个内存模块所接收到的信息包。上游信息包的接收和转发产生上游事务处理序列。此外，可配置上行链接控制单元 241 以将源自内存模块 150 内的信息包注入到事务处理流(transaction stream)中。

在所示意的实施例中，可配置下行链接控制单元 242 以接收源自主机的信息包，并且若下游连接有内存模块，则转发那些信息包至下游的内存模块。此外，可配置下行链接控制单元 242 以复制和译码信息包。在一个实施例中，若信息包包括分配给内存模块 150 的地址范围内的地址且该信息包为内存存取请求，则下行链接控制单元 242 可传递与该信息包相关的命令至 DRAM 控制器 250。在一个实施例中，配置 DRAM 控制器 250，以响应于从内存控制集线器 160 来的内存命令而发起内存循环至内存芯片 261A-261N。然而，若信息包不是内存请求，而是配置信息包，则下行链接控制单元 242 可传递与该信息包相关的配置命令至控制单元 240 的核心逻辑（图中未显示）来处理。值得注意的是，在一个实施例中，若信息包不包括分配给内存模块 150 的地址范围内的地址，则内存控制集线器 160 可舍弃或抛弃该信息包，如果内存模块 150 是链路中的最后一个内存模块的话。

在一个实施例中，配置内存控制集线器 160 以接收模块存在信号（图中未显示），当下游内存模块起动该模块存在信号时，其指示上游内存模块有下游内存模块存在。在这样的实施例中，若内存控制集线器 160 接收事务处理并判定没有下游内存模块存在，则内存控制集线

器 160 可舍弃该事务处理。在一个特定的实施中，当将给定的内存模块 150 插入插口中时，所插入的内存模块可施加接地信号(signal ground)至模块存在信号。因此，有效(active)模块存在信号是有效低信号。

值得注意的是，虽然在上述实施例中当施加接地信号时模块存在信号是有效的，但是预期在其它实施例中，可施加其它的电压电平至模块存在信号，以指示内存模块存在。

参照图 3，显示下行链接控制单元的一个实施例的方块图。在图 3 所示实施例中，下行链接控制单元 342 可以代表图 2 中所示的下行链接控制单元 242。耦接下行链接控制单元 342 以接收下行链接 312A 上的下游事务处理，并发送下行链接 312B 上的那些下游事务处理。在一个实施例中，下行链接 312A 和 312B 可分别代表图 2 的下行链接 212A 和 212B。值得注意的是，如上所述，下行链接 312A 和 312B 包括多个位。例如，在一个实施中，下行链接 312A 和 312B 可各为 18 位下行链接。此外，配置下行链接控制单元 342 以接收来自上游内存模块或主机 100 的忙碌信号(busy signal)371，并将忙碌信号 372 发送至下游内存模块。

在所示意的实施例中，下行链接控制单元 342 包括接收输入事务处理的相位校正单元(phase alignment unit)310。相位校正单元 310 耦接至发送单元 315 并耦接至数据恢复单元 320。数据恢复单元 320 耦接至同步先进先出缓冲器(FIFO)325，该同步 FIFO 325 耦接至地址译码逻辑 330。在一个实施例中，同步 FIFO 325 耦接至诸如 DRAM 控制器 250 的 DRAM 控制器，并耦接至图 2 中控制单元 240 内的核心逻辑。值得注意的是，在图 3 所示的实施例中，相位校正单元 310、发送单元 315 和数据恢复单元 320 可用逐位方式(in a bit-wise manner)独立地操作下行链接的各位。预期在其它实施例中，相位校正单元 310、发送单元 315 和数据恢复单元 320 可基本上同时操作下行链接的所有位。

为了提供所需的内存带宽，内存控制集线器 160 将所接收到的事务处理有效地转发至下游内存模块是很重要的。因此，由下行链接控制单元 342 所接收的事务处理被盲目地向下游转发。在一个实施例中，接收到的各位由相位校正单元 310 相位校正，并送至发送单元 315，而不用译码，用于在下行链接 312B 上传输。此外，接收到的各位送至数

据恢复单元 320 作处理。

在一个实施例中，配置相位校正单元 310，以便对于各位而动态地调整局部取样时钟的相位，以允许下行链接控制单元 342 在所接收到的数据眼(data eye)的中央取样。举例而言，在一个实施例中，从主机提供参考时钟至各内存控制集线器 160 内的锁相环路(phase locked loop, PLL) (图中未显示)。PLL 产生局部取样时钟以及下行链接控制单元 342 内的发送时钟。配置相位校正单元 310 以追踪输入数据流的平均相位。还可以配置相位校正单元 310 以补偿由处理变化所引起的静态偏移(static skew)，追踪由电压和温度变化引起的数据相位的低频变化，并补偿任何低频相位抖动(jitter)，该低频相位抖动可在参考时钟中但未由集线器的 PLL 所追踪到。相位校正单元 310 过滤也许由发送器所产生的符号间干扰、串音或反射噪声和高频相位噪声所引起的高频抖动。

为了促进局部取样时钟的相位校正，也许须提供下行链接 312A 的各位上的充分数目的转变(transition) (或传输密度)。在一个实施例中，通过加密(scrambling)在各位上发送的数据并且然后在接收时将所接收的数据解密(de-scrambling)，而提供转变密度。在一个实施例中，通过对数据与虚拟随机二进制序列(PRBS)进行异或(XOR)运算，而加密数据。线性反馈移位寄存器(LFSR)可用给定的种子多项式(seed polynomial)而产生 PRBS。在另一个实施例中，可通过发送具有预定数据有效载荷(payload)的同步信息包，而提供转变密度。在这样的实施例中，预定的数据有效载荷可包括具有足够数目的转变的随机模式或虚拟随机模式。

不论转变密度是如何得到的，配置数据恢复单元 320 以恢复数据位。在使用加密的实施例中，可配置数据恢复单元 320，以使用与用于加密数据相同的 PRBS 来解密所接收的数据。反之，在用同步信息包以获得转变密度的实施例中，一旦相位校正单元 310 已完成校正局部取样时钟，则数据恢复单元 320 就可舍弃或抛弃同步信息包数据。

在所示意的实施例中，配置同步 FIFO 325 以储存控制单元 240 的核心逻辑所使用的进来的数据位。举例而言，由于事务处理以信息包的方式送出，所以所接收的位可储存于同步 FIFO 325 中，并重新格式化用于译码。然后由地址译码逻辑 330 来译码所接收的信息包。如上

所述，可给各内存模块 150 分配内存地址的一个范围。此外，可给各内存控制集线器 160 分配集线器地址以供配置事务处理使用。若译码出事务处理地址并指示有匹配，则可提取并处理命令信息。在一个实施例中，若事务处理是内存存取请求，且该内存存取请求具有与内存控制集线器 160 的相关内存地址相匹配的地址，则可将译码命令送至 DRAM 控制器 250。若事务处理是配置事务处理，则可将译码命令送至控制单元 240 的核心逻辑。此外，若地址译码逻辑 330 未发现匹配，则可舍弃或抛弃该信息包。

在所示意的实施例中，耦接相位校正单元 335 以接收输入的忙碌信号 371。相位校正单元 335 还耦接至数据恢复单元 350，该数据恢复单元 350 接着又耦接至忙碌脉冲注入单元 355。忙碌脉冲注入单元 355 耦接至发送单元 360。

在所示意的实施例中，相位校正单元 335 和数据恢复单元 350 分别以与相位校正单元 310 和数据恢复单元 320 相似的方式工作。然而，在一个实施例中，忙碌信号 371 为单位(single bit)微分信号而非多位信号。使用忙碌信号 371 和忙碌信号 372 以确保接近内存模块的串联链路底部的集线器将给接近串联链路顶部的集线器留下足够的闲置时间(idle time)，以注入信息包。在一个实施例中，忙碌信号 371 包括将被所有上游内存模块注入到上游事务处理序列中的信息包的数目的指示。因此，忙碌信号 372 包括将被上游所有内存模块注入到上游事务处理序列中的信息包的总数的指示，以及将被内存控制集线器 160 局部地注入的信息包的数目的指示。

在一个实施例中，配置忙碌脉冲注入单元 355，以接收对应于将被上游所有内存模块注入到上游事务处理序列中的信息包的数目的数据。此外，配置忙碌脉冲注入单元 355 以接收从控制单元 240 的核心逻辑来的指示，该指示对应于将被内存控制集线器 160 局部地注入到上游事务处理序列中的信息包的数目。因此，配置忙碌脉冲注入单元 355 以产生忙碌信号 372。在一个实施例中，忙碌信号 371 和 372 可针对各注入信息包而包括一个脉冲。因此，例如，用来指示注入了 5 个信息包的诸如忙碌信号 372 的信号将包括 5 个脉冲。然而，预期在其它实施例中，对于各注入信息包可包括任何数目的脉冲。如下文中将

进一步说明的那样，信息包可作为短信息包或长信息包而送出。因此，在一个实施例中，对于注入的长信息包，忙碌脉冲注入单元 355 可针对各长信息包而包括两个脉冲。

此外，配置忙碌脉冲注入单元 355，以提供对应于从上游接收的忙碌脉冲数目的指示至图 2 的上行链接控制单元 241。在一个实施例中，若内存模块为串行连接链路中的最后一个模块，则上行链接控制单元 241 可依据忙碌信号 371 上所接收的忙碌脉冲的数目，而导致一个或多个无操作(NOP)信息包被注入到上游事务处理序列中。举例而言，在一个实施例中，对于所接收的每两个忙碌脉冲，上行链接控制单元 241 可导致一个 NOP 信息包被注入到上游事务处理序列中。然而，预期在其它的实施例中可使用接收的忙碌脉冲对 NOP 信息包的其它比例。

在所示意的实施例中，配置发送单元 360 以接收来自忙碌脉冲注入单元 355 的忙碌信号。在一个实施例中，进一步配置发送单元 360，以便如上所述，在发送忙碌信号 372 之前用 LFSR 加密数据。或可替代地，发送单元 360 可如上所述注入同步信息包。

参照图 4，显示了上行链接控制单元的一个实施例的方块图。在图 4 所示意的实施例中，上行链接控制单元 441 可代表图 2 中所示的上行链接控制单元 241。耦接上行链接控制单元 441，以接收上行链接 411B 上的上游事务处理，并发送上行链接 411A 上的上游事务处理。此外，配置上行链接控制单元 241，以便将信息包注入至上行链接 411B 上所接收的上游事务处理序列中。在一个实施例中，上行链接 411A 和 411B 可分别代表图 2 中的上行链接 211A 和 211B。值得注意的是，如上所述，上行链接 411A 和 411B 包括多位。例如，在一个实施中，上行链接 411A 和 411B 可各为 16 位上行链接。

在所示意的实施例中，上行链接控制单元 441 包括相位校正单元 410，该相位校正单元 410 接收进来的上游事务处理。相位校正单元 410 耦接至数据恢复单元 415。

为了提供所需的内存带宽，内存控制集线器 160 有效地转发所接收的上游事务处理至上游内存模块或主机是很重要的。与下游控制单元 342 相似，由上行链接控制单元 441 所接收的事务处理向上游转发。然而，与下游控制单元 342 相反，上行链接控制单元 441 可将局部产

生的信息包注入到上游事务处理序列中。

在所示意的实施例中，相位校正单元 410 与相位校正单元 310 以相似的方式工作。因此，配置相位校正单元 410，以便对于各位而动态地调整局部取样时钟的相位，以允许上行链接控制单元 441 如上所述在所接收的数据眼的中央取样。类似地，配置数据恢复单元 415，以便如上述结合了数据恢复单元 320 的描述那样恢复数据位。

在所示意的实施例中，数据恢复单元 415 耦接至多路复用器 430，而该多路复用器 430 耦接至发送单元 435。若上行链接控制单元 441 没有信息包要注入到上游事务处理序列中，则配置上行链接控制单元 441 以便在上行链接 411A 上转发所接收的事务处理。在一个实施例中，信息包边界追踪单元 450 可给多路复用器 430 提供控制信号，这可允许发送单元 435 加密和发送所接收的事务处理。在一个实施例中，发送单元 435 可如上所述用 LFSR 来加密要发送的数据，而在其它的实施例中，发送单元 435 可如上所述注入同步信息包。

然而，在一个实施例中，当接收到来自集线器核心逻辑的要注入的信息包并储存在注入 FIFO 425 内时，则从数据恢复单元 415 接收来的数据储存在保持 FIFO 420 内，用于后来的传输。当完成了当前被转发的信息包的传输时，储存在注入 FIFO 425 内的数据被提供至多路复用器 430 的输入端。此外，信息包边界追踪单元 450 追踪从上行链接 411B 所接收的信息包的信息包边界。信息包边界追踪单元 450 通过在正确时间给多路复用器 430 提供控制信号而确保注入的信息包不会与转发的信息包相冲突；由此允许发送单元 435 加密和发送将要注入的等待信息包。假设有将被注入的信息包储存在注入 FIFO 425 内且有将被转发的信息包储存在保持 FIFO 420 内，则信息包边界追踪单元 450 可依据公平算法(fairness algorithm)而选择地控制多路复用器 430 以将任一种类型的信息包传至发送单元 435。该公平算法可以是配置成针对任何给定内存模块而防止注入信息包的不足(starvation)或转发信息包的不足的任何算法。值得注意的是，可以舍弃上行链接 411B 上接收的 NOP 信息包，而不用储存在保持 FIFO 420 内或转发。

值得注意的是，在一替代实施例中，为了有助于转发事务处理交通量(traffic)的改进等待时间(latency)，相位校正单元 410 可耦接至保持

FIFO 420 和多路复用器 430, 如虚线所示。当没有注入的交通量时, 该实施例可提供改进的等待时间。

进一步值得注意的是, 在一个实施例中, 在主机 100 和各内存模块 150 的各内存控制集线器 160 之间通讯之前, 可执行起始序列, 以使存在于主机 100 内和各内存控制集线器 160 内的任何加密/解密逻辑、同步 FIFO 和相位校正逻辑同步。举例而言, 在一个实施中, 在复位(reset)期间, 起始序列可包括主机 100 向下游发送一个或多个同步模式。该同步模式可由各内存控制集线器 160 向上游转发回主机 100。可使用同步模式以允许包含了加密/解密逻辑的接收器锁住同步。

### 内存互连

共同参照图 1 与图 2, 例如, 内存互连包括一个或多个高速点对点内存链接, 诸如内存链接 110A-110C, 各包括诸如上行链接 211A 的上行链接和诸如下行链接 212A 的下行链接。如上所述, 在一个实施例中, 下行链接可以是 18 位链接而上行链接可以是 16 位链接。这样, 18 位下行链接可以包括 16 个控制、地址和数据(CAD)信号, 1 个忙碌信号和 1 个控制(CTL)信号。给定的上行链接可以包括 16 个控制、地址和数据(CAD)信号。然而, 预期在替代实施例中, 诸如上行链接 211A 的上行链接还可包括 CTL 信号。

除了高速链接外, 也可将其它的信号提供至各内存模块 150。举例而言, 在一个实施例中, 可自主机 100 给各内存模块 150 提供复位信号、电源良好(OK)信号和参考时钟。而且, 可在各内存模块之间提供其它的信号。例如, 如上所述, 可在内存模块之间提供下一个内存模块存在信号。

一般而言, 传输在内存链接 110 上的事务处理的类型可归类为配置和控制事务处理以及内存事务处理。在一个实施例中, 可用配置和控制事务处理来配置内存控制集线器 160。举例而言, 可用配置和控制事务处理来存取配置寄存器、给内存模块分配内存地址范围或给内存控制集线器分配集线器地址。可用内存事务处理来存取内存芯片 261A-261N 内的内存位置。

因此, 在一个实施例中, 支持两种类型的寻址(addressing): 集线

器寻址和内存寻址。使用集线器寻址，8个集线器位识别被存取的特定内存控制集线器。在一个实施例中，FFh的集线器地址可指示至所有内存控制集线器的广播(broadcast)。使用内存寻址，各集线器译码地址位的上部分以判定哪一个集线器应接受此请求，并译码地址位的下部分以判定将要存取的内存位置。在一个实施例中，有40个地址位，但是预期可按需要使用其它数目的地址位。

在一个实施例中，配置各内存链接以用一个或多个信息包来传输事务处理。信息包包括控制和配置信息包以及内存存取信息包，各信息包依据该信息包所载有的命令的类型而可包括数据有效载荷。这样，可用构成内存链接110的线路组来传输控制、地址和数据。

信息包一般可由以下来表征：各信息包包括许多传输一位信息的位位置(bit positions)。各信息包划分成数个位时间(bit times)，且在给定的位时间期间，取样信息包所有的位位置。这样，控制信息和数据共享给定链接的相同线路（例如，CAD线路）。如将在下文中作更详细说明的那样，在一个实施例中，信息包为位对(bit pairs)的倍数，且在偶数的位时间取样每个信息包的第一位时间。信息包以控制标头(header)开始，该控制标头可以是一个位对或二个位对的长度。在一个实施例中，控制标头的最初五位为命令码。下面的表1示意了各种类型的信息包和它们相关的命令码。然而，应当注意的是，显示于第一列的实际的码是为了示意的目的，且对于各给定的命令可使用其它的码。

表 1. 信息包类型和命令码

码	标头长度 (位时间)	命令	说明	方向	正常响应	地址 类型
00h	-	NOP	零操作/ 闲置状态	上下	-	-
04h	2	AddrSet	地址组	下	AddrAck	集线器
05h	2	AddrAck	地址确认	上	-	-
06h	2	Ack	确认	上	-	-
07h	2	Nak	非确认/ 错误	上	-	-



08h	2	SRdResp	短读取 响应	上	-	-
09h	2	LRdResp	长读取 响应	上	-	-
0Ah	2	ConfigRd	配置读取	下	RdResp	集线器
0Ch	2	ConfigWr	配置写入	下	Ack	集线器
0Eh	2	DIMMCtl	DIMM 控制	下	Ack	集线器
10h	4	SMemRd	短内存 读取	下	RdResp/ Ack	内存
11h	4	LMemRd	长内存 读取	下	RdResp	内存
12h	4	BlkMemWr	区块 内存写入	下	Ack	内存
13h	4	SbytMemWr	短字节 内存写入	下	Ack	内存
14h	4	LbytMemWr	长字节 内存写入	下	Ack	内存

再者，在一个实施例中，用错误侦测码(error detecting code, EDC)来发送信息包（除了 NOP 信息包外）。值得注意的是，在一个实施例中，EDC 为 32 位循环冗余码(cyclic redundancy code, CRC)，虽然其它的实施例可按需要使用其它的 EDC's。此外，首先在最有效(most significant)位时间发送地址以加速内存控制集线器 160 内的译码，而数据以最不有效(least significant)字节首先送出。然而，应注意的是，在其它实施例中可预期地址以最不有效位时间首先送出，而数据以最有效字节首先送出。信息包可载有字节允许(byte enables)和/或数据的有效载荷。不具有有效载荷的信息包被称为仅具有标头的信息包(header-only packet)。在一个实施例中，数据短读取的大小可达到编程高速缓存线大小(programmed cache line size)的一半。此外，长读取和区块写入(block writes)的数据大小可达到编程高速缓存线大小。再者，字

节写入的数据大小可以最大为 64 字节,而无关于高速缓存线大小设定。

除了包含在信息包内的控制标头和命令码信息外,可用 CTL 信号来传输关于各信息包的信息。如下面表 2 所示,显示了一些示例性的 CTL 编码。

表 2. 用于下游的 CTL 编码

偶数	奇数	CAD 的内容
0	0	数据或字节允许有效载荷
1	1	控制标头
0	1	具有有效载荷的信息包的 CRC
1	0	仅具有标头的信息包的 CRC

信息包的标头部分和有效载荷部分的不同 CTL 值可提供足够的信息以允许仅具有标头的信息包被插入到另一信息包的有效载荷内。当仍在链接上发送写入信息包时允许读取命令发出,这对于减少读取命令的等待时间也许很有用处。表 3 示意了示例性信息包,包括以表格形式的有效载荷。表 3 中的信息包还显示,在位时间 4-7 期间仅具有标头的信息包被插入到有效载荷中。

表 3. 具有有效载荷的信息包和插入在有效载荷中的  
仅具有标头的信息包

位时间	CTL	CAD
0	1	标头 1 位[15:0]
1	1	标头 1 位[31:16]
2	0	数据位[15:0]
3	0	数据位[31:16]
4	1	标头 2 位[15:0]
5	1	标头 2 位[31:16]
6	1	CRC2 位[15:0]
7	0	CRC2 位[31:16]
8	0	数据位[47:32]
9	0	数据位[64:48]
10	0	CRC1 位[15:0]
11	1	CRC1 位[31:16]

图 5A 至图 5D 示意了可在图 1 的内存链接 110A 至 110C 上传输的示例性信息包。现参照图 5A，显示了配置读取信息包的一个实施例的图。在所示意的实施例中，配置读取信息包 510 为 16 位宽并包括 4 个位时间或 2 个位对。

在位时间 0 期间，5 位命令码（例如，0Ah）在位位置 0-4 传输。位位置 5-7 保留。8 位标记符(tag)在位位置 8-15 传输。在一个实施例中，例如，主机 100 可使用标记符值来匹配待处理的下游请求与上游响应信息包。若未被使用，则可将标记符字段(tag field)设定值 00h。在位时间 1 期间，8 位集线器地址在位位置 0-7 传输。此外，8 位配置寄存器数目在位位置 8-15 传输。在位时间 2 和 3 期间，CRC 的位 0-15 和 16-31 分别在位位置 0-15 传输。

参照图 5B，显示了配置写入信息包的一个实施例的图。在所示意的实施例中，配置写入信息包 515 为 16 位宽并包括 6 个位时间或 3 个位对。

在位时间 0 期间，5 位命令码（例如，0Ch）在位位置 0-4 传输。位位置 5-7 保留。8 位标记符在位位置 8-15 传输。在位时间 1 期间，8 位集线器地址在位位置 0-7 传输。此外，8 位配置寄存器数目在位位置 8-15 传输。在位时间 2 和 3 期间，数据有效载荷的位 0-15 和 16-31 在位位置 0-15 传输。在位时间 4 和 5 期间，CRC 的位 0-15 和 16-31 分别在位位置 0-15 传输。

参照图 5C，显示了内存读取信息包的一个实施例的图。在所示意的实施例中，内存读取信息包 520 为 16 位宽并包括 6 个位时间或 3 个位对。在位时间 0 期间，5 位命令码（例如，10h 或 11h）在位位置 0-4 传输。位位置 5-7 保留。8 位标记符在位位置 8-15 传输。

在位时间 1 期间，应返回的数据的长度在位位置 0-5 传输。在一个实施例中，值 00h 指示没有数据，值 01h 指示 2 个位对的数据，值 02h 指示 4 个位对的数据，以及等等。零长度读取使确认信息包(ACK)被返回到请求者。在一个实施例中，一半高速缓存线或少于一半高速缓存线的读取可造成短 RdResp，而多于一半高速缓存线的读取可造成单个长 RdResp 或两个短 RdResp。可由软件将高速缓存线大小编程到主机 100 和各内存控制集线器 160 的配置寄存器。位 6-7 保留。DRAM 中请

求位置的地址位 39-32 在位位置 8-15 传输。

在位时间 2 期间, DRAM 中请求位置的地址位 31-16 在位位置 0-15 传输, 而在位时间 3 期间, DRAM 中请求位置的地址位 3-15 在位位置 3-15 传输。在位时间 3 期间, 还在位位置 0-1 传输信息包优先级(packet priority)。在一个实施例中, 优先级可指示信息包相对于其它请求的优先级。举例而言, 一个优先级可以是延迟具有较低优先级的所有请求, 即使它们已经在进行中, 并在它们之前执行这一请求。位位置 2 保留。在位时间 4 和 5 期间, CRC 的位 0-15 和 16-31 分别在位位置 0-15 传输。

参照图 5D, 显示了区块内存写入信息包的一个实施例的图。在所示意的实施例中, 区块内存写入信息包 525 为 16 位宽并包括 8 个位时间或 4 个位对。在位时间 0 期间, 5 位命令码 (例如, 12h) 在位位置 0-4 传输。位位置 5-7 保留。8 位标记符在位位置 8-15 传输。

在位时间 1 期间, 数据有效载荷中传输的数据的长度在位位置 0-5 传输。在一个实施例中, 值 00h 指示没有数据, 值 01h 指示 2 个位对的数据, 值 02h 指示 4 个位对的数据, 以及等等。位 6-7 保留。被写入的 DRAM 中位置的地址位 39-32 在位位置 8-15 传输。

在位时间 2 期间, 被写入的 DRAM 中位置的地址位 31-16 在位位置 0-15 传输, 而在位时间 3 期间, 被写入的 DRAM 中位置的地址位 3-15 在位位置 3-15 传输。在位时间 3 期间, 还在位位置 0-1 传输信息包优先级。位位置 2 保留。

在位时间 4 和 5 期间, 数据有效载荷的第一位对的位 0-15 和 16-31 在位位置 0-15 传输。若正在写入更多的数据, 则后续的位对可传输后续数据有效载荷的位 0-15 和 16-31。在位时间  $4+2N$  和  $5+2N$  期间, CRC 的位 0-15 和 16-31 分别在位位置 0-15 传输。

应注意的是, 虽然仅显示了 4 种类型的信息包, 但是可考虑使用对应于表 3 中所列命令码的其它类型的信息包。应进一步注意的是, 虽然显示了示例性信息包的各种字段具有特定的位数, 但是可考虑在其它实施例中, 所选用信息包的各种字段可按需要包括其它的位数。

图 6 为计算机系统的一个实施例的方块图。计算机系统 600 包括处理节点 612A-612D, 各由相干的信息包接口链接(coherent packet interface links) 615A-615D 所互连。相干信息包接口的各链接 615 可形

成高速点对点链接。处理节点 612A-612D 可各包括一个或多个处理器。计算机系统 600 还包括 I/O 节点 620, 该 I/O 节点 620 经由不相干的信息包接口 650A 耦接至处理节点 612A。I/O 节点 620 例如可以由不相干的信息包接口 650B 而连接到链路布局中的另一个 I/O 节点 (图中未显示)。处理节点 612A 显示为主节点, 并可包括经由 NC 信息包接口 650A 而与 I/O 节点 620 通讯的主桥接器(host bridge)。处理节点 612B-612D 也可包括用于与其它 I/O 节点 (图中未显示) 通讯的主桥接器。由 NC 信息包接口 650A-650B 所形成的不相干信息包接口链接也可被称为点对点链接。I/O 节点 620 连接至一对外围总线 625A-625B。

图 6 还显示了耦接至处理节点 612A 和 612B 的各自系统内存 (例如, 617A 和 617B)。在所示意的实施例中, 处理节点 612A 和 612B 各示意如图 1 中所示的主机, 且各系统内存 617 可在上面结合图 1 和图 2 的描述所说明的配置中实施。而且, 各处理节点 612A 和 612B 以及它们各自系统内存 617 之间的互连可以反映上面图 1 和图 2 中所说明的包括内存链接 110C 的内存互连。值得注意的是, 在其它实施例中, 可使用其它数目的处理节点。再者, 例如, 可考虑各处理节点 612C 和 612D 相似地连接至各自的系统内存, 诸如系统内存 617。

在所示意的实施例中, 相干信息包接口的各链接 615 实施成单方向线组 (例如, 线 615A 用来将信息包从处理节点 612A 发送至处理节点 612B, 而线 612B 用来将信息包从处理节点 612B 发送至处理节点 612C)。其它组的线 615C-615D 用来在图 1 中所示的其它处理节点之间发送信息包。可用高速缓存相干的方式来操作相干的信息包接口 615, 用于处理节点之间的通讯 (“相干的链接”)。再者, 不相干的信息包接口 650 可以用不相干的方式来操作, 用于 I/O 节点之间的通讯和 I/O 节点与诸如处理节点 612A 的主桥接器的主桥接器之间的通讯 (“不相干的链接”)。经由相干链接的两个或多个节点的互连可称为 “相干的组构(coherent fabric)”。类似地, 经由不相干链接的两个或多个节点的互连可称为 “不相干的组构(non-coherent fabric)”。应注意的是, 要从一个处理节点发送至另一个处理节点的信息包可通过一个或多个中间节点。例如, 如图 6 中所示, 由处理节点 612A 发送至处理节点 612C 的信息包可经由处理节点 612B 或处理节点 612D。可使用任何适当的

路由算法。计算机系统 600 的其它实施例可比图 6 中所示实施例包括有更多或更少的处理节点。

诸如不相干信息包接口 650 的信息包接口的一个例子可与 HyperTransport™ 技术兼容。外围总线 625A 和 625B 显示普通的外围总线(common peripheral bus), 诸如外围组件互连(peripheral component interconnect, PCI)总线。然而, 应了解, 可使用其它类型的总线。

还应注意, 可考虑使用其它的计算机系统配置。例如, 预期上面图 1 至图 5 中所描述的系统内存配置可与采用了包含北桥接器的处理器芯片组的计算机系统结合使用。在这样的实施例中, 北桥接器内的内存控制器可用作主机。

虽然已相当详细地描述了上述实施例, 但是对于本领域的技术人员而言, 一旦完全理解了上述公开, 许多的改变和修改将是显而易见的。下列的权利要求将被解释为包括所有这样的改变和修改。

### 产业适用性

本发明通常用于计算机系统内存。

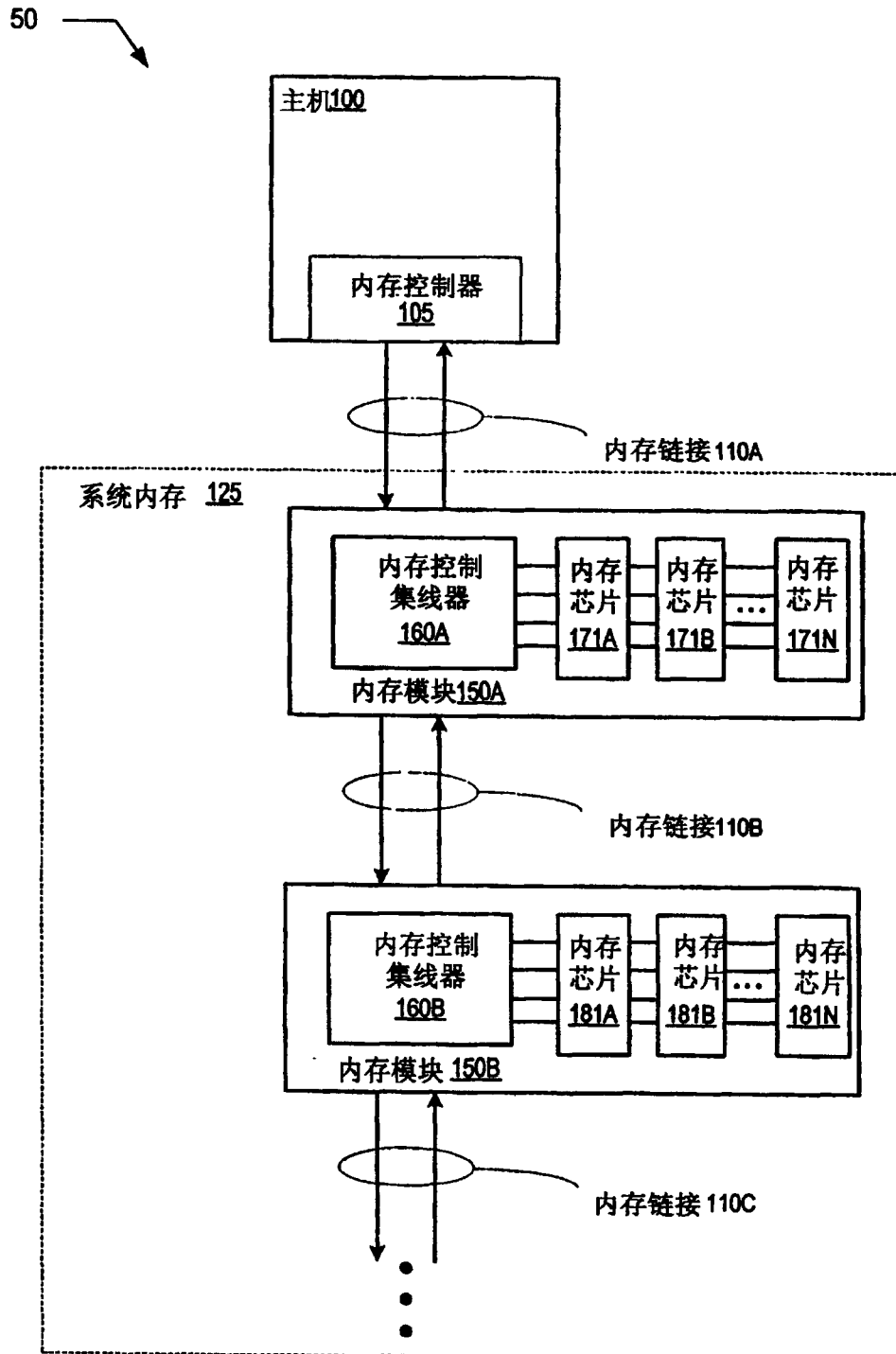


图1

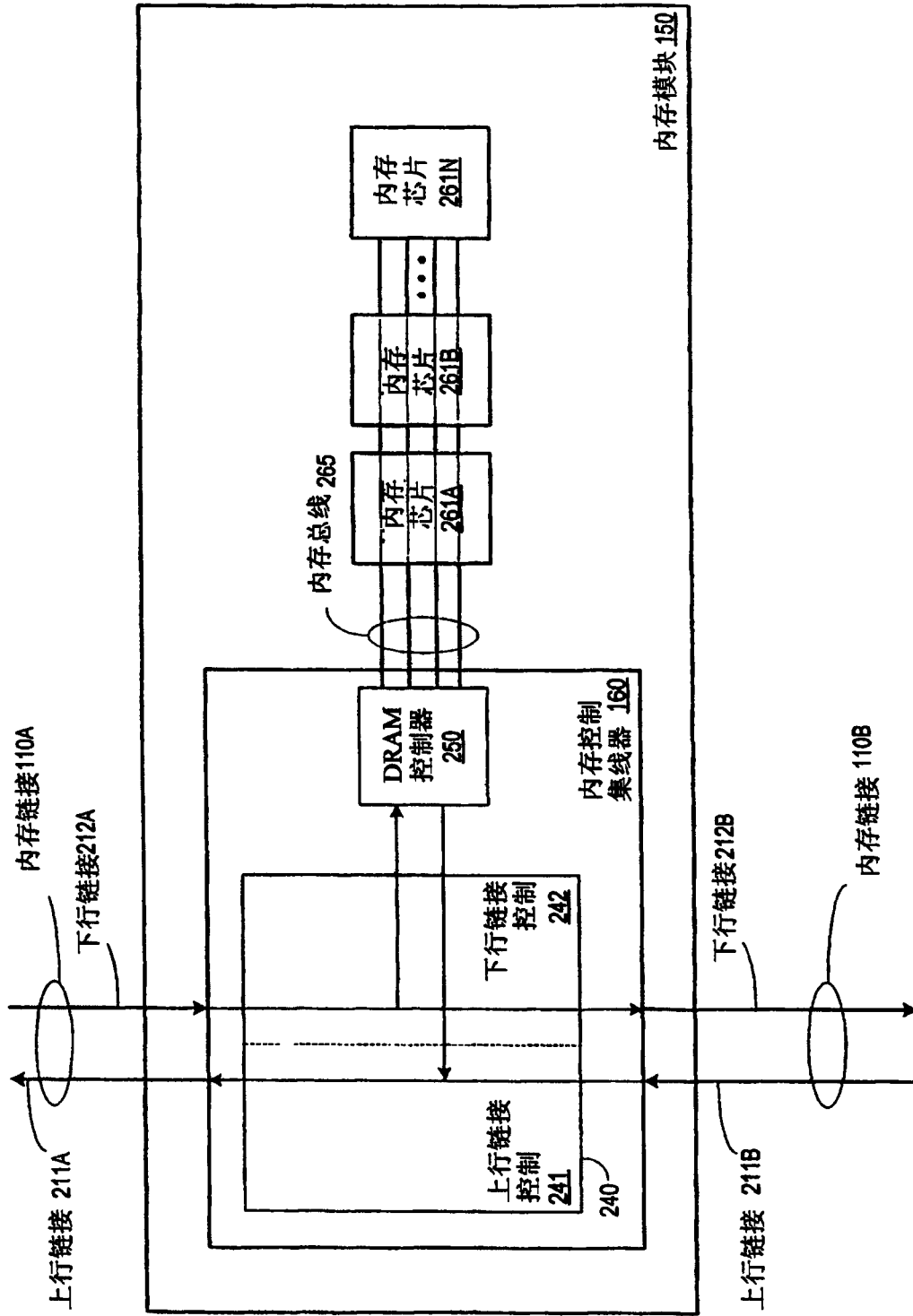


图2



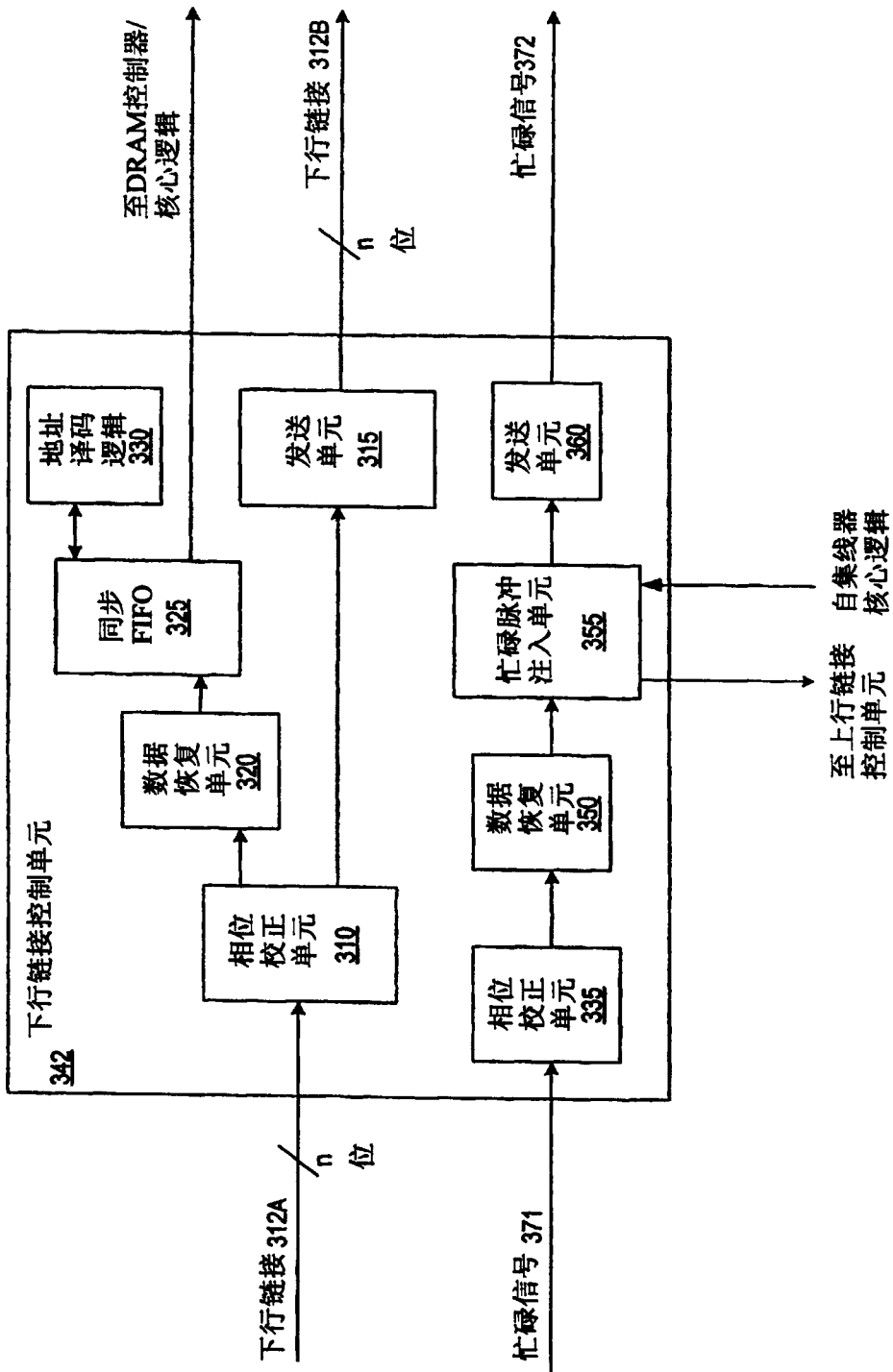


图3

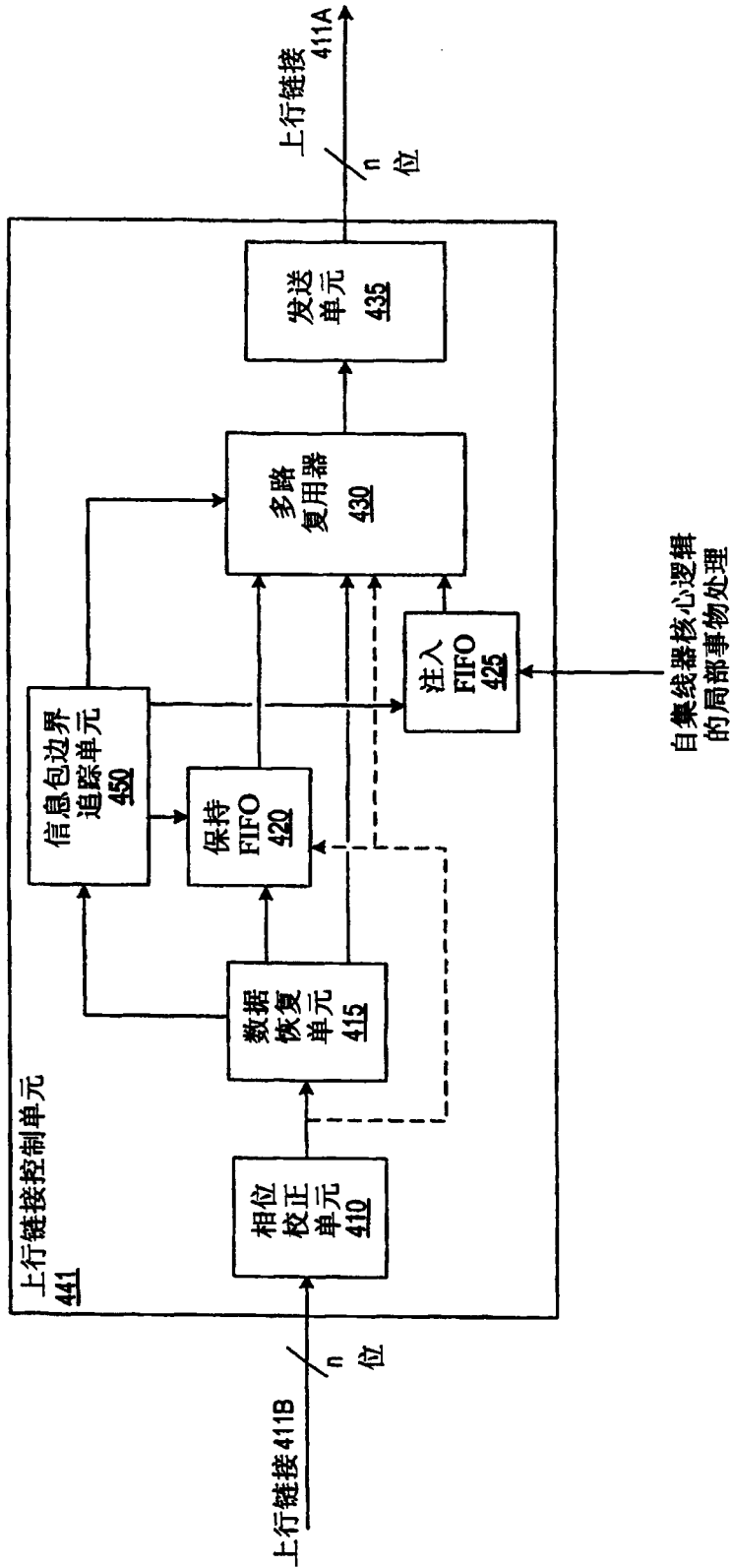


图4

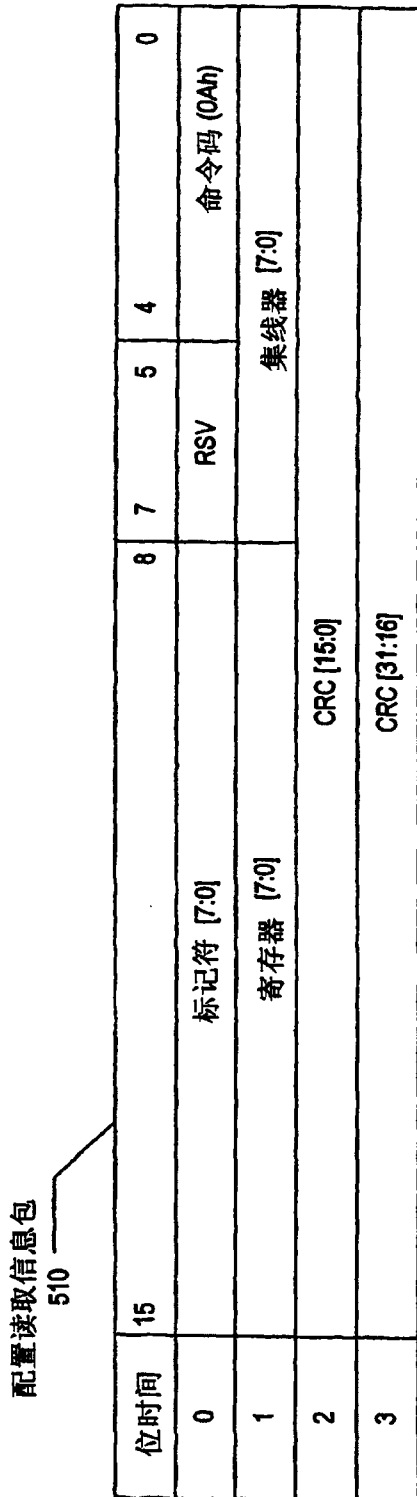


图5A

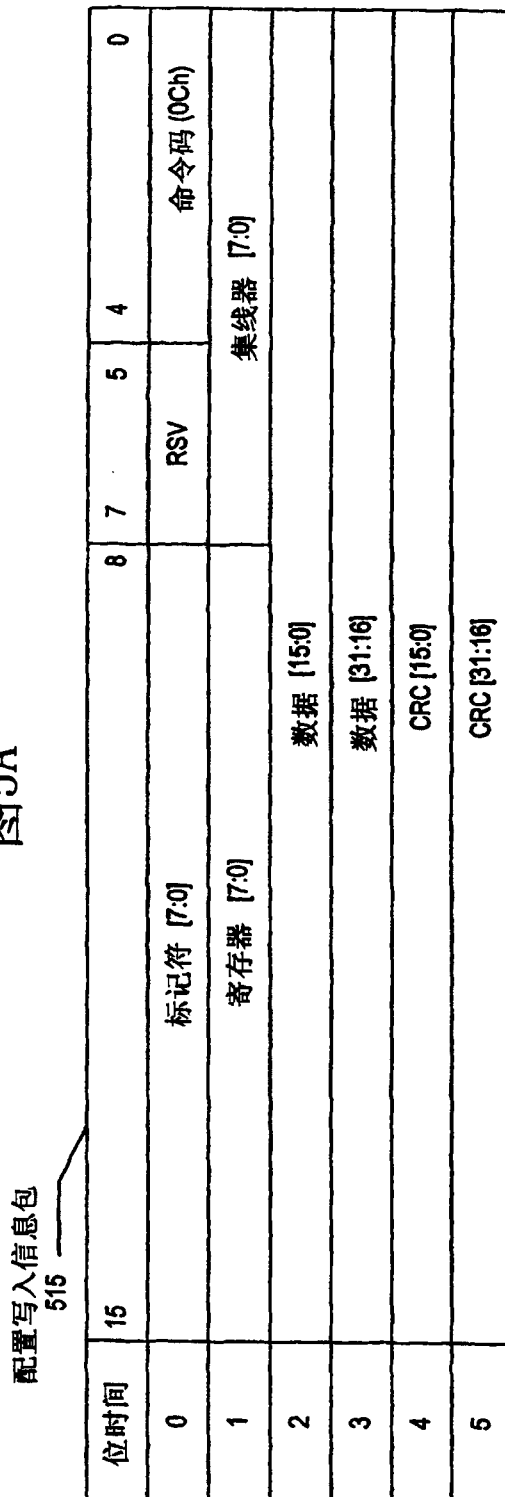


图5B

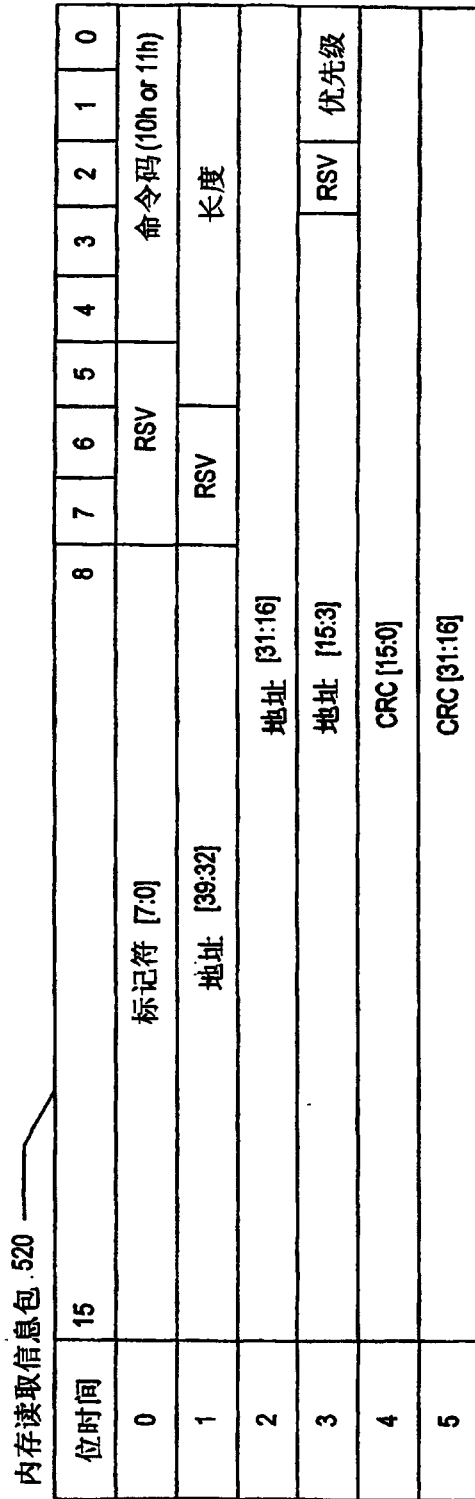


图5C

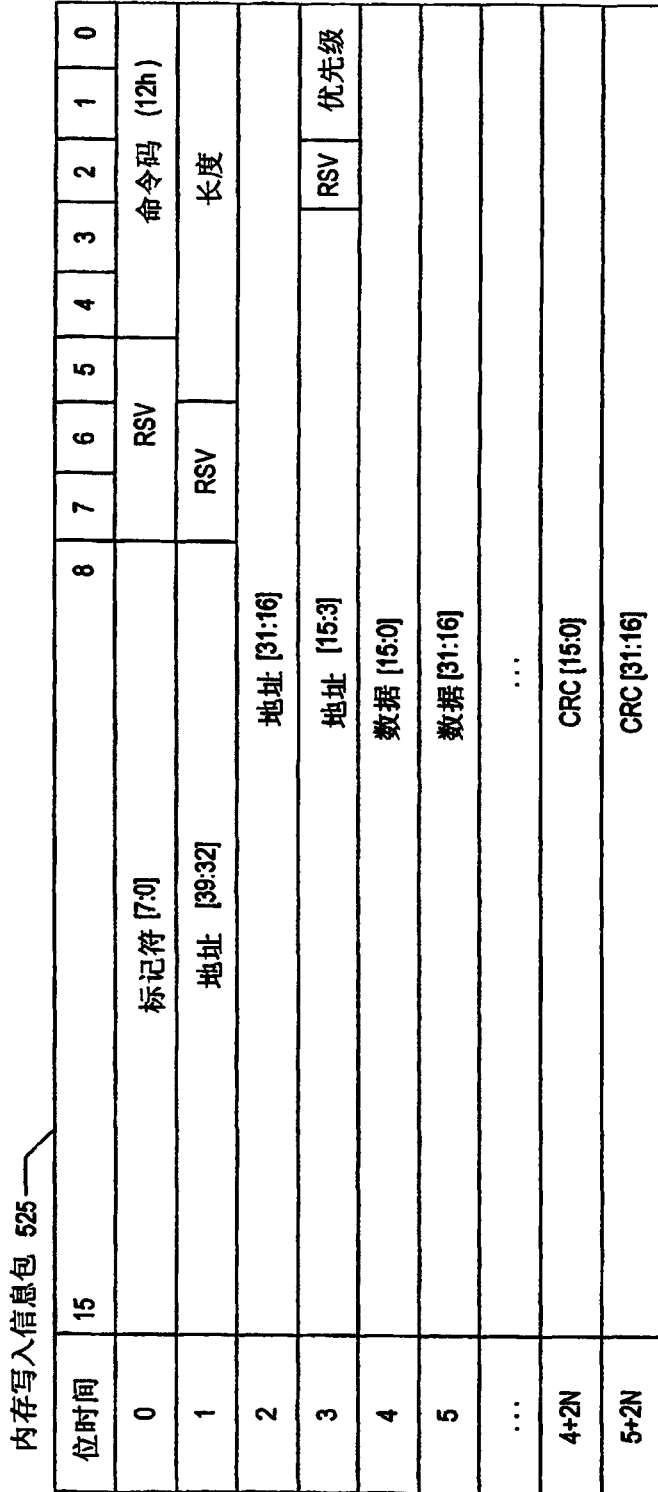


图 5D

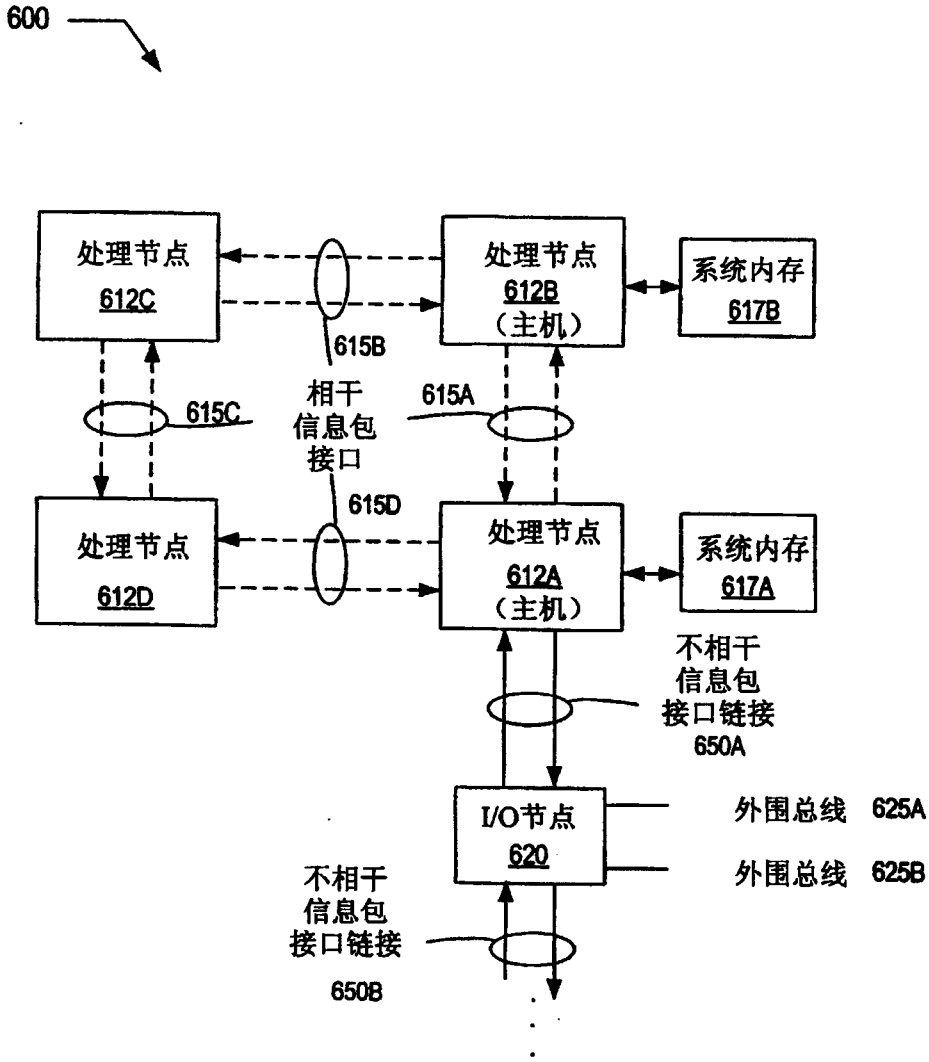


图6