

[19] 中华人民共和国国家知识产权局

[51] Int. Cl⁷

G06F 11/30

G06F 13/00



[12] 发 明 专 利 说 明 书

专利号 ZL 01103095. X

[45] 授权公告日 2005 年 10 月 12 日

[11] 授权公告号 CN 1222879C

[22] 申请日 2001.2.6 [21] 申请号 01103095. X

[30] 优先权

[32] 2000. 2. 7 [33] US [31] 09/498,812

[71] 专利权人 自由度半导体公司

地址 美国得克萨斯

[72] 发明人 威廉·C·摩耶尔

迈克尔·D·菲兹西蒙斯

理查德·G·柯林斯

审查员 刘 栩

[74] 专利代理机构 中国国际贸易促进委员会专利
商标事务所

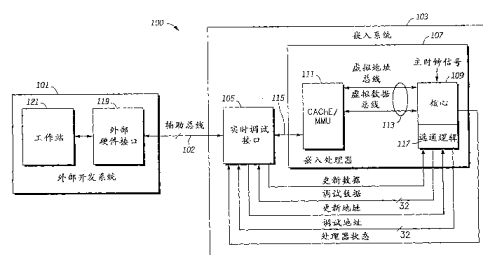
代理人 王以平

权利要求书 1 页 说明书 13 页 附图 4 页

[54] 发明名称 实时处理器调试系统

[57] 摘要

一种实时处理器调试系统，在实时操作期间对核心处理器之虚拟总线的地址和数据信号进行选择采样，降低功耗和性能影响。嵌入处理器的选通逻辑电路，向嵌入系统芯片或集成电路(IC)的调试接口，提供核心处理器的虚拟总线信号，包括地址和数据信号。选通逻辑电路从调试接口接收更新数据和更新地址信号，以控制相应调试地址和数据总线上的转换。核心处理器向调试接口提供处理器状态信号，调试接口确定核心处理器执行的特定事务。



1. 一种集成电路，其特征在于包括：

一个中央处理器，用于在中央处理器的正常操作期间，提供操作信号；

一个逻辑电路，具有一个用于接收操作信号的第一输入端，一个用于接收调试更新信号的第二输入端，和一个输出端；以及

一个实时调试电路，具有提供调试更新信号的一个第一输出端，连接到该逻辑电路之输出端的一个第一输入端，和一个第二输出端，其中第二输出端连接到可从集成电路之外部访问的辅助总线。

2. 权利要求 1 的集成电路，其中集成电路具有调试方式，并且实时调试电路具有作为调试方式的子集的可选方式。

3. 权利要求 2 的集成电路，其中在可选方式持续的过程中认定调试更新信号。

4. 权利要求 3 的集成电路，其中中央处理器生成到达实时调试电路的处理器状态信号，并且响应表示处理器地址流改变的处理器状态信号，认定调试更新信号。

5. 权利要求 3 的集成电路，其中中央处理器生成到达实时调试电路的处理器状态信号，并且响应表示处理器数据访问的处理器状态信号，认定调试更新信号。

6. 一种在具有调试电路和中央处理器的集成电路中的实时调试系统中使用的方法，其中中央处理器生成供集成电路内部使用的操作信号，该方法的特征在于包括以下步骤：

在集成电路的正常操作期间，进入调试方式；

进入可选方式，该可选方式为调试方式的子集；

响应调试电路在可选方式期间认定调试信号，将操作信号连接到调试电路；以及响应调试电路在可选方式期间取消认定调试信号，阻塞调试电路中的操作信号。

实时处理器调试系统

技术领域

本发明涉及实时处理器调试系统；更具体地说，涉及在实时操作期间对核心处理器之虚拟总线的地址和数据信号进行选择采样，以便降低功耗，以及将由于总线加载而引起的性能影响降到最低程度的调试系统。

背景技术

一些包含核心处理器、片上存储器以及外部存储器接口模块的嵌入系统是很普遍的。现有的生成系统还在集成高速缓冲存储器。系统级的代码开发人员必须为嵌入系统编写软件或应用程序代码，以实现该系统需要的特定功能。由于各种各样的原因，包括添加或修改功能性，删除不需要的功能部件，或删除“程序错误”或软件错误，开发人员必须多次修改大部分软件。因此，为了确保正确运行，必须在应用程序代码开发期间调试软件。在运行期间，嵌入系统的应用程序代码可能还需要一直进行连续不断的调整，或周期调整。在某些配置中，某些参数、特征值或常数表可能需要在实时操作期间进行调整或修改。此处所指的调试过程包括，初始应用程序代码开发，以及运行期间的实时校正或调整功能。

某些旧式系统，包括不带高速缓冲存储器的系统，提供外部核心处理器总线，从而能够轻而易举地获得核心处理器周期，以便进行调试。在具有高速缓冲存储器的嵌入系统环境中，系统级的调试是比较困难的，这是由于核心处理器的管脚或总线不再可见的缘故。特别是，高速缓冲存储器通常位于嵌入核心处理器和外部物理总线之间，从而不能从外部到达核心处理器总线。例如，事实上，在不带高速缓冲存储器的现有嵌入系统中，“显示周期”支持不是一个可选项。

传统调试方法称为后台调试方式(BDM)。BDM是一种静态调试

过程，其中暂停处理器以便代码开发人员执行调试操作。代码开发人员可以设置断点，以便在特定断点停止处理器，或者单步执行各指令，以监视处理器的进程并确定问题和程序错误。BDM 对某些应用程序是足够的，但并不能满足实时操作，包括动态调整或校正。此外，某些调试操作需要在不停止嵌入核心处理器的条件下，测试并改正应用程序代码。例如，通常在汽车引擎运转的情况下，调试控制汽车引擎的嵌入处理器系统，从而暂停核心处理器不是一个可选项，这是由于它会有效停止引擎的缘故。

人们正在开发动态调试方法，以便在不暂停处理器的情况下，监视处理器的操作。在能够从外部达到高速缓冲存储器的物理总线的情况下，从外部提供核心处理器总线是不实际的，也是不可行的，这是由于它会导致嵌入处理器的过多数目的管脚。期望开发一种调试接口，该接口监视处理器的总线，并经由内部辅助总线或端口提供消息。通过监视核心处理器和高速缓冲存储器之间的地址和数据总线进行的动态调试，是一个消耗功率的侵入过程。高速缓冲存储器和核心处理器之间的总线通常称为“虚拟”总线。连接到虚拟总线的大缓冲器引起总线加载，后者会影响嵌入处理器系统的性能。此外，监视地址和数据总线信号并向外部辅助总线提供相应消息的动态过程，将消耗宝贵的能量。便携通信设备通常是电池供电的手持设备，并主要使用嵌入处理器系统。此类便携通信设备包括诸如个人数字助理（PDA）、蜂窝电话、寻呼机、全球定位系统（GPS）模块之类的设备。在此类便携通信设备中，必须将功耗降到最低程度，以延长电池寿命。

将在正常系统运行期间执行动态校正的系统的功耗，降到最低程度也是同样重要的。汽车应用是此类需求的重要示例。例如，在某些汽车应用中，实时校正（不断）调谐要求在处理器和应用程序运行时，运行调试模块。强烈希望将嵌入系统的功耗降到最低程度。

期望以将由于嵌入处理器系统上之总线加载而引起的功耗和性能影响降到最低程度的方式，执行动态调试。

发明内容

为克服现有技术的不足，本发明提供了一种集成电路，其特征在于包括：一个中央处理器，用于在中央处理器的正常操作期间，提供操作信号；一个逻辑电路，具有一个用于接收操作信号的第一输入端，一个用于接收调试更新信号的第二输入端，和一个输出端；以及一个实时调试电路，具有提供调试更新信号的一个第一输出端，连接到该逻辑电路之输出端的一个第一输入端，和一个第二输出端，其中第二输出端连接到可从集成电路之外部访问的辅助总线。

本发明还提供了一种在具有调试电路和中央处理器的集成电路中的实时调试系统中使用的方法，其中中央处理器生成供集成电路内部使用的操作信号，该方法的特征在于包括以下步骤：在集成电路的正常操作期间，进入调试方式；进入可选方式，该可选方式为调试方式的子集；响应调试电路在可选方式期间认定调试信号，将操作信号连接到调试电路；以及响应调试电路在可选方式期间取消认定调试信号，阻塞调试电路中的操作信号。

附图说明

通过连同附图一起考虑最佳实施方式的详细说明，将更好地理解本发明，其中附图为：

图 1 为调试系统的示意框图，说明根据本发明的实时处理器调试系统。

图 2 为根据本发明之实施方式实现的虚拟地址总线选通逻辑的示意图。

图 3 为根据本发明之实施方式实现的虚拟数据总线选通逻辑的示意图。

图 4 为一定时图，说明根据本发明之某一实施方式的实时调试接口系统的程序跟踪虚拟地址选通操作。

图 5 为一定时图，说明根据本发明之某一实施方式的实时调试接口系统的程序和数据跟踪虚拟地址和数据选通操作。

具体实施方式

当前，正在开发 Nexus 标准或 GEPDIS（全球嵌入处理器调试接

口标准),以便执行嵌入系统的动态调试,并解决与实时、非侵入调试支持有关的其他问题和争论。GEPDIS,另外称为 IEEE-ISTO(电气和电子工程师协会 — 工业标准和技术组织)论坛 5001,描述了一种嵌入调试接口标准,该标准是为了在不暂停嵌入系统处理器之核心情况下对非侵入调试方法进行标准化而开发的。GEPDIS 中定义了许多调试能力,以便通过提供程序流和数据流的可视性,来监视程序执行。该可视性由在专用多比特或多管脚串行接口或辅助端口上,向外部开发系统提供的信息消息序列组成。然后,将程序流消息与该程序的静态映象结合,以重建嵌入处理器的实际指令执行序列。数据流消息跟踪处理器读取并写入到预定地址范围。

使用包括同步消息处理和数据跟踪消息处理的程序跟踪消息处理,实现 GEPDIS 动态调试。程序跟踪消息处理的实现,要求监视核心处理器执行的指令提取序列和状态信息,后者表示流改变(COF)事件,包括直接或间接的流改变事件。直接 COF 事件包括与程序计数器有关的分支,而间接 COF 事件包括寄存器间接分支和例外引向。程序跟踪同步消息的实现,要求传送当前执行的指令地址或相当邻近的指令地址。数据跟踪消息处理的实现,要求监视数据存取地址,并且有条件地提供关联数据。数据跟踪消息处理包括读取消息处理,或在一个或多个定义地址范围内的数据读取,以及写消息处理,或在一个

或多个定义地址范围内的数据写入。

GEPDIS 意在描述一种实时动态的非侵入调试系统。GEPDIS 期望一种嵌入调试接口，该接口访问虚拟总线，并经由辅助总线或端口，与外部系统通信。然而，GEPDIS 不能明确解决总线加载和功耗。直接向调试接口提供虚拟总线信号是侵入的，并且可能会引起错误操作。可以使用缓冲处理来降低调试接口的影响，但仍会造成很多性能影响，这是由于虚拟总线的所有信号转变均对调试信息有不良影响。调试总线上的功耗是很大的，并且在某些实施方式中，可能是过大的。

图 1 为调试系统 100 的示意框图，该系统包括一个连接到嵌入系统 103 之辅助总线 102 的外部开发系统 101。开发系统 101 用于执行软件的实时调试操作，其中嵌入系统 103 之嵌入处理器 107 的处理器核心正在执行该软件。然而，可以理解，也可以将实时校正系统（未示出）连接到辅助总线 102，以便执行实时校正，其中本发明的原理相同。

在所示实施方式中，将嵌入系统 103 插入到集成电路（IC）或芯片上，然后插入到诸如与调试系统 100 关联的较大系统之电路板的适当插槽中。嵌入系统 103 包括一个实时调试接口 105，它经由独立的物理总线 115，连接到嵌入处理器 107。嵌入处理器 107 包括一个中央处理器（CPU）或核心 109，它经由虚拟总线 113，连接到高速缓冲存储器/MMU（存储器管理部件）111。虚拟总线 113 包括许多操作信号，这些信号定义核心 109 的操作。高速缓冲存储器和 MMU 通常为独立系统，但为了简洁，以组合形式表示它们。MMU 为选件，并且在嵌入处理器 107 的特定实施方式中，可以不提供此选件。本发明设想了嵌入处理器的许多不同变更和实施方式，并且不受此类变更的限制。

虚拟总线 113 包括一条虚拟地址总线和一条双向虚拟数据总线，虚拟地址总线包括由核心 109 认定的到达高速缓冲存储器/MMU 111 的地址信号，而双向虚拟数据总线位于核心 109 和高速缓冲存储器/MMU 111 之间。术语“虚拟”表示虚拟总线 113 是不可见的，或者相反，是不向嵌入处理器 107 外部提供到达嵌入系统 103 之其他组件

的。物理总线 115 类似于虚拟总线 113，并包括类似的地址数据和控制信号，但是这两条总线在运行期间的操作不同。可以禁用高速缓冲存储器/MMU 111 内的高速缓冲存储器，此时虚拟总线 113 的运行方式与物理总线 115 的运行方式相同。然而，如果启用高速缓冲存储器，则物理总线 115 以不同方式运行，并且不能真实表示核心 109 的操作。希望监视虚拟总线 113，以便监视核心 109 的操作，或者调试由核心 109 正在执行的应用程序软件的校正操作，或者相反，执行这种校正操作。

向连接到核心 109 的虚拟总线选通逻辑 117，提供虚拟总线 113。虚拟总线选通逻辑 117 可以插入到核心 109 内，或者被设计为来自核心 109 的外部。经由更新数据信号、更新地址信号、调试数据总线和调试地址总线，将虚拟总线选通逻辑 117，连接到实时调试接口 105。调试地址总线可包括将在下文中描述的属性信号。更新数据信号为数据调试更新信号，该信号有选择地启动或者在其他情况下启用调试数据总线上的信号；而更新地址信号为地址调试更新信号，该信号有选择地启用或启动调试地址总线。向实时调试接口 105 提供由核心 109 生成的处理器状态信号，以确定核心 109 的某些状态事件，这将在下文中进一步说明。处理器状态信号包括至少一个核心事务状态输出信号，该信号指出核心 109 的事务状态。

辅助总线 102 是一个可伸缩的输出调试端口，由嵌入系统 103 的实时调试接口 105 控制该端口。在一种实施方式中，由 GEPDIS 定义辅助总线 102，以使开发系统 101 监视核心 109。向外部硬件接口 119 提供辅助总线 102，以向开发系统 101 内的开发工作站 121 提供调试信号，其中接口 119 包含缓冲器固件、处理能力等。工作站 121 安装有任意固件、硬件或软件，并且能够以个人计算机之类的方式实现，以控制调试操作或应用程序软件的开发。操作员在工作站 121 与调试软件进行交互，以便控制实时调试接口 105 的操作，包括监视虚拟总线 113 的地址和数据信号。

嵌入系统 113 可以具有几种操作方式，包括用于正常操作的正常方式和调试方式。实时调试接口 105 使调试方式更加容易，并且可以

具有几种可选方式作为调试方式的子集。例如，实时调试接口包括程序跟踪方式和数据跟踪方式，其中数据跟踪方式还有读和写调试方式。

图 2 为根据本发明之实施方式的虚拟地址总线选通逻辑 200 的示意图。在虚拟总线选通逻辑 117 内，提供虚拟地址总线选通逻辑 200，以便将虚拟总线 113 的虚拟地址总线连接到调试地址总线。在此处所示的实施方式中，虚拟地址总线包括 32 个独立地址信号，标记为虚拟地址[31:0]。然而，可以理解，本发明并不限于任意特定数目的地址信号。将各虚拟地址总线信号提供给一连串锁存器 201 的相应输入，锁存器可以为任意形式的合适锁存器，如 D 型锁存器。将锁存器 201 的各锁存器的 Q 输出，提供给一连串缓冲器 203 的各输入。缓冲器 203 的输出认定调试地址总线的各个信号，包括各信号调试地址[31:0]。嵌入处理器 107 包括一个内部主时钟信号，称为 CLK1，将该信号提供给具有两个输入的与门 205 的一个输入。CLK1 信号为至少一个主时钟信号，将该信号提供给核心 109，以用于控制和定时。与门 205 的另一输入接收更新地址信号。与门 205 的输出认定信号调试地址时钟，将后者提供给锁存器 201 之各锁存器的时钟输入。

在运行中，更新地址信号用作启用信号，以将虚拟地址总线的各信号，传送到与 CLK1 信号之认定同步的调试地址总线上。这样，如果认定了更新地址信号，则 CLK1 信号用时钟同步锁存器 201，以将虚拟地址[31:0]信号，锁存到调试地址[31:0]信号的各个信号中。当对更新地址信号取反或该信号为逻辑零时，锁存器 210 是静态的，从而调试地址[31:0]信号保持稳定。请注意，转变嵌入处理器 107 的任意总线，包括调试地址总线，其功耗是很大的。正如下文中进一步说明的那样，使用更新地址信号来降低或排除调试地址总线的多余转变，是非常省电的。

图 3 为根据本发明之实施方式实现的虚拟数据总线选通逻辑 300 的示意图。也将虚拟数据总线选通逻辑 300，插入到虚拟选通逻辑 117 中，以便有选择地将虚拟总线 113 的各数据信号，传送到调试数据总线。在所示实施方式中，虚拟总线 113 的虚拟数据总线包括 32 个信号，

虚拟数据[31:0]。调试数据总线也包括 32 个信号，调试数据[31:0]。然而，可以理解，本发明并不限于任何特定数目的数据信号。将更新数据信号，提供给许多具有两个输入的与非门 301 的各与非门的一个输入端。与非门 301 的各与非门的另一输入端，接收虚拟数据[31:0]信号的各信号。例如，第一与非门 301 接收虚拟数据[0]信号，第二与非门 301 接收虚拟数据[1]信号，等等。将各与非门 301 的输出，提供给许多反相器 303 的各反相器的输入端。反相器 303 的各反相器在其各自输出端，认定调试数据[31:0]信号的各个信号的相应信号。

除了无需与 CLK1 信号同步之外，虚拟数据总线选通逻辑 300 的操作，与虚拟地址总线选通逻辑 200 的操作类似。然而，可以理解，能够以与虚拟数据总线选通逻辑 300 相似的方式，实现虚拟地址总线选通逻辑 200，反之亦然。在此处所描述的特定实施方式中，虚拟地址总线选通逻辑 200 稍微延迟其地址，以确保周期真实有效并能适当锁存。其他优点在于，调试地址总线和信号的定时并不很关键，因此，可以使用较小、较低的功耗设备实现缓冲器 203。当更新数据信号取反为低时，调试数据总线保持静态不变。当认定更新数据信号为高时，在经过逻辑门 301、303 的轻微延迟后，调试数据总线逻辑跟随虚拟总线 113 的虚拟数据总线。特别是，当认定更新数据信号为高时，调试数据[0]信号跟随虚拟数据[0]信号，调试数据[1]信号跟随虚拟数据[1]信号，等等。

应该理解，虚拟地址总线选通逻辑 200 和虚拟数据总线选通逻辑 300，分别经由更新地址信号和更新数据信号，有选择地控制并启动相应的调试地址总线和调试数据总线。这样，可以利用更新地址和更新数据信号，来控制调试地址和调试数据总线的功耗。同样，能够在调试、测试或校正操作期间，控制嵌入系统 103 的功耗。特别是在诸如寻呼机和移动电话之类的电池供电的便携设备中，在正常操作方式中禁用调试方式。将更新地址和更新数据信号强制为低，以确保调试地址和调试数据总线不转变。这可确保附加的调试方式逻辑或电路不消耗大量电源，从而不影响电池寿命。

当需要对调试数据总线和调试地址总线的调试信号进行采样时，实时调试接口 105 认定更新数据和更新地址信号。同时，从核心 109 向实时调试接口 105 提供各种属性信号，其中用处理器状态表示该信号。由开发系统 101 控制的实时调试接口 105，确定何时认定更新数据和更新地址信号，以启动或启用调试数据和调试地址信号，供实时调试接口 105 进行监视。在特定实施方式中，实时调试接口 105 可以启动或启用程序跟踪，以监视在流改变（COF）事件期间由核心 109 执行的指令提取序列。处理器状态信号表示 COF 状态，而虚拟总线 113 的虚拟地址总线认定 COF 地址。如处理器状态信号所表示的那样，在 COF 状态或事件期间，实时调试接口 105 认定更新地址信号，以启动调试地址总线并对它取样。同步消息也可以为程序跟踪操作的一部分，其中实时调试接口 105 认定更新地址信号。

数据跟踪消息包括监视数据访问地址，以及有条件地提供的关联数据。对一个或多个定义地址范围的任意一个地址范围的数据读写，数据跟踪是有用的。当启用数据跟踪消息处理时，实时调试接口 105 监视读写周期中包含的数据提取的处理器状态。处理器状态信号还表示是否正在出现读写周期。对于写消息处理或读消息处理，实时调试接口 105 均认定数据提取期间的更新地址信号，以在适当数据周期期间捕获相应地址。例如，对于读消息处理，如果处理器状态总线表示读周期，则实时调试接口 105 认定更新地址信号。同样，对于写消息处理，如果处理器状态总线表示写周期，则实时调试接口 105 认定更新地址信号。当然，对于读写信号而言，如果同时启用读写消息处理，则实时调试接口 105 认定更新地址信号。同时，当启用数据跟踪时，实时调试接口 105 比较从调试地址总线检索的地址，与读写消息之一或多个定义地址范围内的任意地址。如果发现某个地址与读周期或写周期匹配，则实时调试接口 105 认定更新数据信号，该信号使虚拟总线选通逻辑 117 认定调试数据总线上的相应数据。

图 4 为一定时图，说明根据本发明之某一实施方式的程序跟踪虚拟地址选通操作。以嵌入系统 103 之定时引用的方式，表示核心 109

的主时钟信号 CLK1 和 CLK2。该定时图表示虚拟总线 113 的虚拟地址，后者说明标记为 ADDRESS1、ADDRESS2、COF 地址、ADDRESS4 和 ADDRESS5 的五个地址的认定。该定时图还表示虚拟数据总线，后者分别以 DATA1、DATA2、DATA3、DATA4 和 DATA5 的方式说明五个相应数据信号。该图同时表示处理器状态信号，该信号说明一条指令提取，接着是 COF 事件，然后是数据提取，最后是另一条指令提取。该定时图还表示更新地址信号、调试地址时钟信号以及调试地址总线。

在图 4 的示例中，启用程序跟踪，而禁用数据跟踪。由于启用数据跟踪，所以无需监视虚拟数据总线上的数据。因此，实时调试接口 105 保持更新数据信号的取反为低，从而调试数据总线保持静态待用，而不管虚拟数据总线上的活动。这提供了显著降低功耗的直接优点，因为不需要消耗功率来转换调试数据总线。

实时调试接口 105 无需监视虚拟地址总线上的所有活动。虚拟地址总线上认定的许多地址，为先前认定地址的递增形式。这样，一旦知道一连串顺序地址的初始地址，剩余地址就是已知的或者是可预测的，从而无需将它们传送到调试地址总线上。这对于任意寻址模式均成立，寻址模式包括顺序地址未必递增的模式，只要知道二进制顺序即可。实时调试接口 105 只需例如响应条件或非条件分支或循环指令，抑或任何直接或间接 COF 事件，确定并读取不服从顺序次序的地址。在处理器状态信号上认定的 COF 指令，标识非顺序地址。

如图 4 所示，即使在虚拟地址总线上认定顺序地址 ADDRESS1 和 ADDRESS2，调试地址总线也保持上一 COF 地址不变。具体而言，当实时调试接口 105 在处理器状态信号上检测到一个 COF 指令时，它按照事件箭头 401 所示的方式，认定更新地址信号。更新地址信号的认定启用图 2 所示的锁存器 201。如事件箭头 403 所示，在认定更新地址信号时，在 CLK1 信号的下一上升边上，在以 405 指示的位置认定调试地址时钟信号。调试地址时钟信号的认定启用锁存器 201，从而以事件箭头 407 所示的方式，在调试地址总线信号上驱动虚拟地址

总线信号。

这样，作为对实时调试接口 105 认定的更新地址信号的响应，调试地址总线仅从上一 COF 地址转变为新的 COF 地址。如事件箭头 409 所示，在 CLK2 信号的下一上升边上，完成处理器状态信号上的 COF 指令。如事件箭头 411 所示，作为 COF 指令完成的响应，实时调试接口 105 取消认定更新地址信号。作为响应，按事件箭头 413 所示，对调试地址时钟信号取反。由于调试地址时钟信号此后保持取反，所以调试地址总线保持不变，并静态保持新的 COF 地址，从而降低了功耗。

图 5 为一定时图，说明根据本发明之某一实施方式的程序和数据跟踪虚拟地址和数据选通操作。在此示例中，同时启用程序跟踪和数据跟踪。由于启用程序跟踪，所以实时调试接口 105 按上面参照图 4 描述的方式，一旦认定 COF 指令，就监视虚拟地址总线。由于启用了数据跟踪，并且假设同时启用读和写数据跟踪，所以实时调试接口 105 监视虚拟数据总线上出现的有效数据事件。有效数据事件是由工作站 121 定义的，并将该事件传送到实时调试接口 105，以进行监视。例如，可能希望确定写入到特定地址之特定寄存器、或特定地址范围内的许多寄存器或任意其他类型之存储设备中的数据，或从上述设备中读取的数据。可以定义许多此类地址或地址范围。

当启用数据跟踪时，实时调试接口 105 监视所有数据提取指令的处理器状态信号。当检测到数据提取指令时，实时调试接口认定更新地址信号，以便从调试地址总线上的虚拟地址总线中捕获读或写地址。然后，实时调试接口 105 比较从调试地址中检索的地址，与正在监视的任意或所有适用地址或地址范围。如果该地址等于监视的地址或地址范围，则实时调试接口 105 认定更新数据信号，以便在数据事件有效时，将虚拟数据总线上的认定数据，传送到调试数据总线上。请注意，如果正在监视特定地址范围的读和写操作，则认为数据事件有效。否则，仅当正在执行的周期类型与正在监视的周期类型相同时，才认为数据事件有效。

同时，实时调试接口 105 根据处理器状态信号，确定该数据周期

是读周期还是写周期。如果只启用了读数据跟踪，但检测到写周期，或相反，则实时调试接口 105 将不认定更新地址信号，这是由于认为该数据事件无效的缘故。即使启用了读和写数据跟踪，也需要区别特定地址范围的读周期或写周期。例如，可能希望确定某个地址范围的读周期和另一个不同地址范围的写周期。在一种实施方式中，实时调试接口 105 内的比较逻辑，使用周期类型来确定适用的监视地址范围。当然，如果对所有定义的地址范围均启用了读和写操作，则无需区别读周期和写周期。

如图 5 所示，COF 指令出现在处理器状态信号上，由实时调试接口 105 检测该指令。如事件箭头 501 所示，实时调试接口 105 比照地认定更新地址信号。如事件箭头 503 所示，在认定更新地址信号时，CLK1 信号的认定，引起调试地址时钟信号的伴随认定。随后，CLK2 信号的认定，使得处理器状态信号从 COF 指令转变为数据提取指令。实时调试接口 105 检测处理器状态信号上的数据提取指令，由于在此示例中启用了数据跟踪，所以实时调试接口 105 以事件箭头 507 所示的方式，使更新地址信号保持认定。如事件箭头 511 所示，由于更新地址信号保持认定，所以调试地址时钟信号的第二次认定，将虚拟地址总线上出现的数据写地址，锁存到调试地址总线中。由于启用了数据跟踪，所以实时调试接口 105 检索在调试地址总线上认定的数据写地址（数据读写地址），其中实时调试接口 105 比较检索的地址，与此时活动的读消息处理或写消息处理之所有特别定义的地址范围。在图 5 所示的示例中，实时调试接口 105 比较数据写地址与有效地址范围，并确定调试地址的比较匹配，然后按事件箭头 513 所示方式，比照地认定更新数据信号。如事件箭头 515 所示，由实时调试接口 105 进行的更新数据信号的认定，使得虚拟数据总线上的信号经由逻辑门 301 和 303，传送到调试数据总线，导致在调试数据总线上认定的写数据。

可以理解，逻辑门 205、301，锁存器 201 和缓冲器 203、303 提供隔离，而隔离提供降低虚拟总线 113 上之加载的优点。使用降低调试地址和调试数据总线的转换的更新地址和更新数据信号，进一步提

供了降低虚拟总线 113 上之加载的优点，同时明显节省了嵌入处理器 107 的电能，从而节省了嵌入系统 103 的电能。

尽管连同最佳实施方式一起说明了根据本发明的系统和方法，但其意图并不是限制于此处所阐述的特定形式，而是相反，其意图是覆盖包含于权利要求书定义的本发明之实质和范围内的上述选择、修改和等同物。

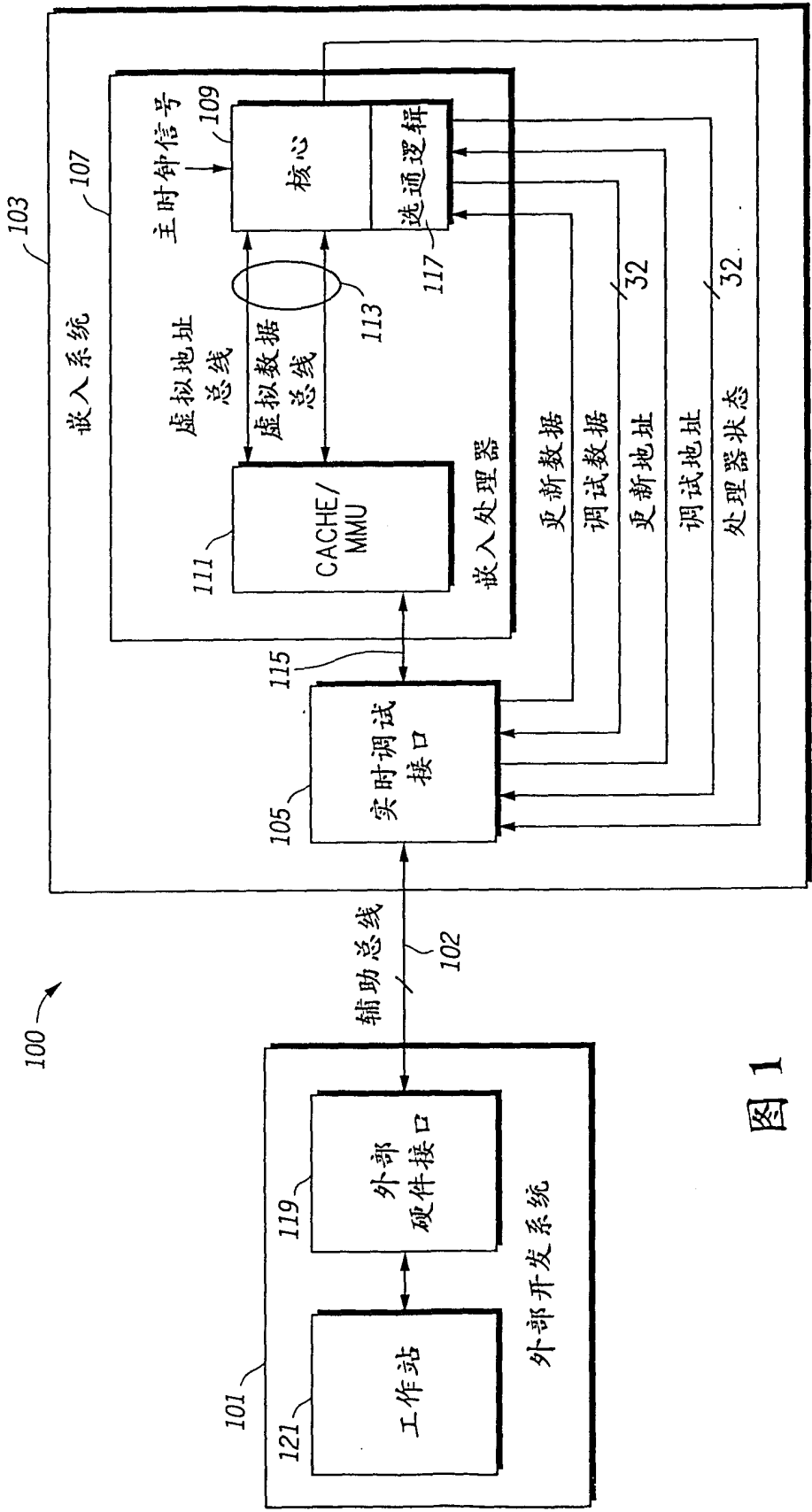


图 1

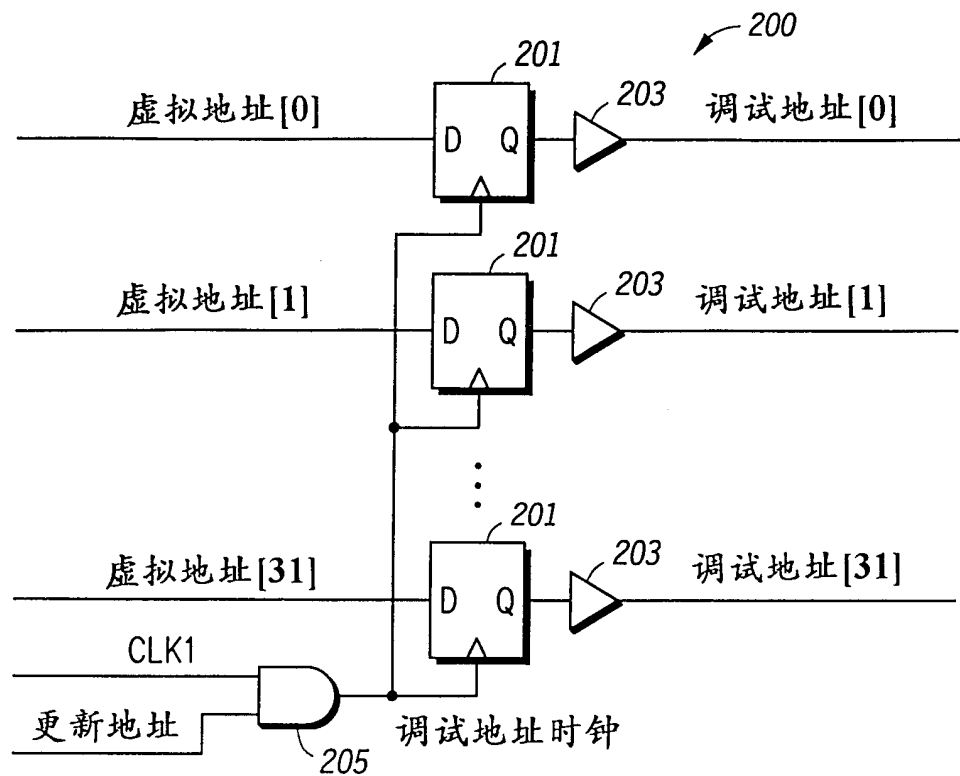


图 2

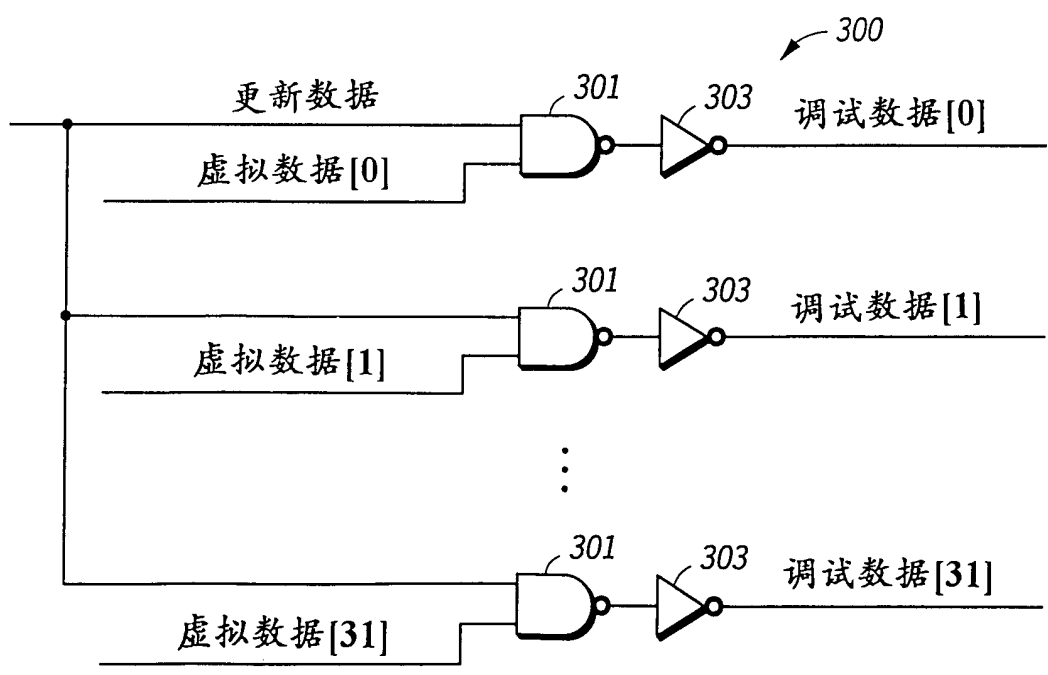


图 3

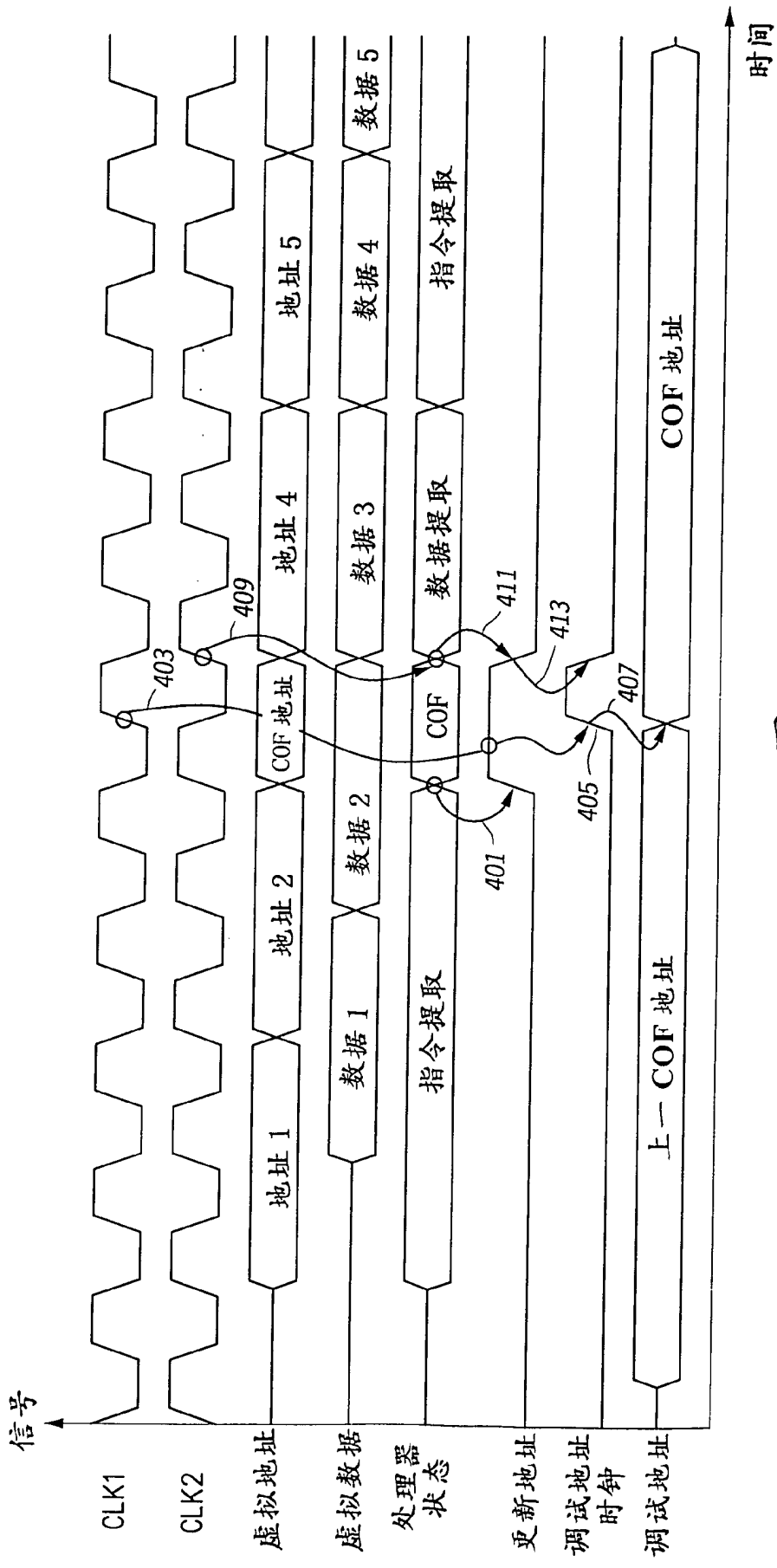


图 4

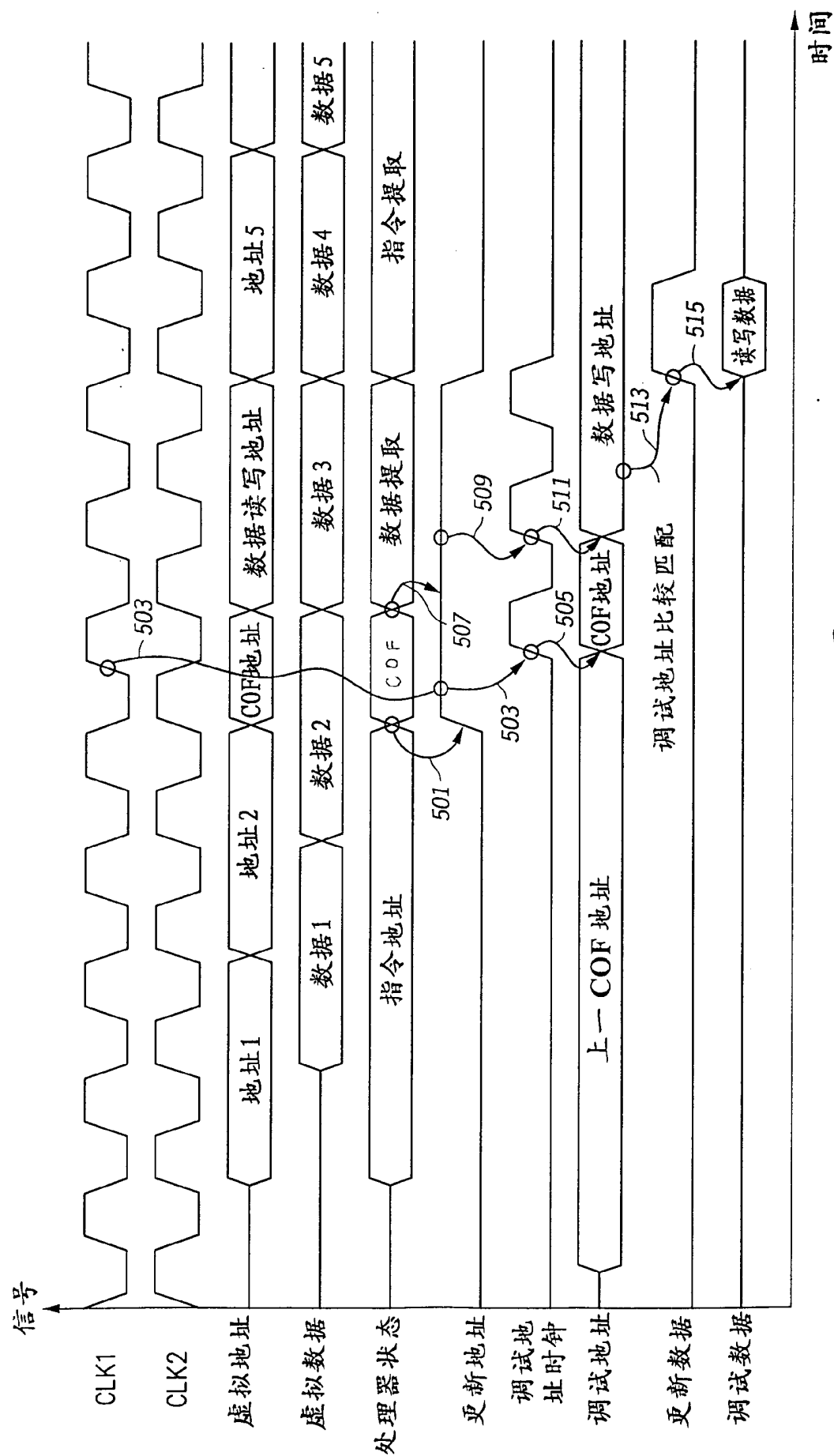


图5