



[12] 发明专利申请公开说明书

[21] 申请号 02812492.8

[43] 公开日 2004 年 8 月 11 日

[11] 公开号 CN 1520537A

[22] 申请日 2002.4.17 [21] 申请号 02812492.8

[30] 优先权

[32] 2001. 5. 10 [33] US [31] 09/852,942

[32] 2001. 5. 10 [33] US [31] 09/852,372

[32] 2001. 5. 11 [33] US [31] 09/853,226

[86] 国际申请 PCT/US2002/011935 2002.4.17

[87] 国际公布 WO2002/093336 英 2002.11.21

[85] 进入国家阶段日期 2003.12.22

[71] 申请人 先进微装置公司

地址 美国加利福尼亚州

[72] 发明人 G·S·斯特劳金

[74] 专利代理机构 北京纪凯知识产权代理有限公司

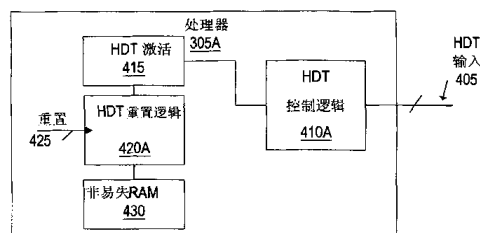
代理人 戈 泊 程 伟

权利要求书 5 页 说明书 11 页 附图 9 页

[54] 发明名称 用于在个人计算机系统中关闭后门访问的机制

[57] 摘要

本发明提供一种可以关闭后门访问机制的方法、装置和系统。处理器包括第一寄存器，该寄存器设定成可储存一个或多个硬件除错测试(hardware - debug - test, HDT)激活位；相连的第一控制逻辑以便接收多个 HDT 输入信号；以及和第一寄存器相连的第二控制逻辑。第一控制逻辑的连接方式使其可以访问第一寄存器，而第二控制逻辑设置成在第一寄存器内储存一个或多个缺省值以响应该处理器的重置。另外一个处理器则包括相连的第一控制逻辑用以接收多个微码输入；与第一控制逻辑相连的第一寄存器；以及与第一寄存器相连的第二控制逻辑。第一寄存器设定成使其可以储存一个或多个微码装载器激活位，而第二控制逻辑设定成储存一个或多个缺省值在第一寄存器内，以响应处理器的重置。



1. 一种处理器，包括：

5 第一寄存器，该第一寄存器被设置成可储存一个或多个HDT激活位；

相连的第一控制逻辑以便接收多个HDT输入信号，其中该第一控制逻辑的连接方式使其可以访问第一寄存器；以及

与该第一寄存器相连的第二控制逻辑，其中该第二控制逻辑被设置成当处理器重置时，可以在该第一寄存器内储存一个或多个缺省值。

10

2. 根据权利要求1所述的处理器，其中该第一控制逻辑进一步被设定成可接收一个进入HDT模式的请求，其中该第一控制逻辑进一步被设置成响应对进入HDT模式的请求，读取储存在该第一寄存器内的一个或多个HDT激活位的选取内含值，而其中该第一控制逻辑进一步设定成
15 根据该一个或多个HDT激活位的选取内含值而允许或拒绝该进入HDT模式的请求。

3. 根据权利要求1所述的处理器，进一步包括：

20 一个或多个非易失性存储器，该存储器被设定成可以为一个或多个HDT激活位储存一个或多个缺省值，其中该第二控制逻辑进一步连接成从该一个或多个非易失性存储器为一个或多个HDT激活位读取一个或多个缺省值，以及为一个或多个HDT激活位将一个或多个缺省值写至该第一寄存器，以响应处理器的重置。

25 4. 根据权利要求1所述的处理器，其中该第二控制逻辑进一步连接成为一个或多个HDT激活位接收一个或多个缺省值的信号指示，并将一个或多个HDT激活位的一个或多个缺省值写至第一寄存器内，以响应处理器的重置。

30 5. 根据权利要求1所述的处理器，进一步包括：

第三寄存器，该第三寄存器被设置成可以储存一个或多个微码装

载器激活位；

第三控制逻辑，该第三控制逻辑的连接方式使其可以接收多个微码输入，而其中该第三控制逻辑的连接方式使其可以访问该第三寄存器；以及

- 5 第四控制逻辑，该第四控制逻辑和该第三控制逻辑相耦接，其中该第四控制逻辑被设置成使其可以在响应处理器重置时，在该第三寄存器内储存一个或多个缺省值。

6. 根据权利要求5所述的处理器，其中该第三控制逻辑进一步设定成可以接收一个修改微码的请求，其中该第三控制逻辑进一步设定成可以响应修改微码的请求，而读取储存在该第三寄存器内的一个或多个微码装载器激活位的选取内含值，且其中该第三控制逻辑进一步设定成可以根据一个或多个微码装载器激活位的选取内含值而决定允许或拒绝该修改微码的请求。

15

7. 根据权利要求1所述的处理器，进一步包括：

第二寄存器，该第二寄存器和该第一控制逻辑相连接，其中该第二寄存器被设置成可以储存一个或多个HDT激活锁码位。

- 20 8. 根据权利要求7所述的处理器，其中该第一控制逻辑进一步设定成可以接收修改HDT模式状态的请求，其中该第一控制逻辑被设置成使其可以响应修改HDT模式状态的请求，而读取储存在第二寄存器内的一个或多个HDT激活锁码位的选取内含值，而且其中该第一控制逻辑进一步设定成可以根据在一个或多个HDT激活锁码位的选取内含值而决定允许或拒绝该修改HDT模式状态的请求。

25

9. 一种处理器，包括：

相连的第一控制逻辑，用以接收多个微码输入；

- 30 与该第一控制逻辑相连的第一寄存器，其中该第一寄存器设定成可以储存一个或多个微码装载器激活位；以及

与该第一寄存器相连的第二控制逻辑，其中该第二控制逻辑被设

置成可以响应该处理器重置时，储存一个或多个缺省值在第一寄存器内。

10.根据权利要求9所述的处理器，其中该第一控制逻辑进一步设定成可以接收修改微码的请求，其中该第一控制逻辑进一步设定成可以响应修改微码的请求，而读取储存在该第一寄存器内的一个或多个微码装载机激活位的选取内含值，以及其中该第一控制逻辑进一步设定成可以根据一个或多个微码装载机激活位的选取内含值而决定允许或拒绝该修改微码的请求。

10

11.根据权利要求9所述的处理器，进一步包括：

一个或多个非易失性存储器，该存储器设定成可以为一个或多个微码装载机激活位储存一个或多个缺省值，其中该第二控制逻辑进一步连接成可以从该一个或多个非易失性存储器为一个或多个微码装载机激活位读取一个或多个缺省值，而且可以响应处理器在重置时，为一个或多个微码装载机激活位将一个或多个缺省值写至微码装载机寄存器。

15

12.根据权利要求9所述的处理器，其中该第二控制逻辑进一步连接成可以接收指示性信号，即在响应处理器在重置时，用以指示一个或多个微码装载机激活位的一个或多个缺省值，然后将一个或多个微码装载机激活位的一个或多个缺省值写至该第一寄存器内。

20

13.根据权利要求9所述的处理器，进一步包括：

第二寄存器，该第二寄存器和该第一控制逻辑相连接，其中该第二寄存器被设置成可以储存一个或多个微码装载机激活锁码位。

25

14.根据权利要求13所述的处理器，其中该第一控制逻辑进一步设定成可以接收一个修改微码装载机锁码状态的请求，其中该第一控制逻辑进一步设定成可以响应修改微码装载机锁码状态的请求，而读取储存在该第二寄存器内一个或多个微码装载机激活锁码位的选取内含值，

30

而且其中该第一控制逻辑进一步设定成可以根据在一个或多个微码装载机激活锁码位的选取内含值而决定允许或拒绝该修改微码装载机锁码状态的请求。

5 15.一种判定HDT模式激活状态的方法，该方法包括：

接收起动HDT模式的请求；

判定HDT模式激活状态；以及

如果该HDT模式激活状态设定成激活，则起动HDT模式。

10 16.根据权利要求15所述的方法，其中判定HDT模式激活状态包含从寄存器中读取与一个或多个HDT激活位相对应的一个或多个的内含值。

17.一种修改微码的方法，该方法包括：

接收修改微码的请求；

15 判定微码装载机激活状态；以及

如果该微码装载机激活状态设定成激活，则修改微码。

18.根据权利要求17所述的方法，其中判定微码装载机

20 激活状态包含从寄存器中读取和一个或多个微码装载机激活位相对应的一个或多个的内含值。

19.一种改变HDT模式状态的方法，该方法包括：

接收改变HDT模式状态的请求；

判定HDT模式激活锁码状态；以及

25 如果该HDT模式激活锁码状态设定成解除锁码，则修改HDT模式状态。

20.根据权利要求19所述的方法，其中判定HDT模式激活

30 锁码状态包含从寄存器中读取与一个或多个HDT激活锁码位相对应的一个或多个的内含值。

21.根据权利要求19所述的方法，其中修改HDT模式状态包含将与一个

或多个HDT激活位相对应的一个或多个的内含值写至寄存器中。

22.一种改变微码装载机激活状态的方法，该方法包括：

接收改变微码装载机激活状态的请求；

5 判定微码装载机激活锁码状态；以及

如果该微码装载机激活锁码状态设定成解除锁码，则修改微码装载机激活状态。

23.根据权利要求22所述的方法，其中决定微码装载机激活锁码状态包
10 含从寄存器中读取与一个或多个微码装载机激活锁码位相对应的一个或多个的内含值。

24.根据权利要求22所述的方法，其中修改微码装载机激活状态包含将
15 与一个或多个微码装载机激活位相对应的一个或多个的内含值写至寄存器中。

25.一种操作处理器的方法，该方法包括：

获取一个或多个缺省值，其中获取该一个或多个缺省值是从下列的组集中选取出来，该组集包含：

20 从一个或多个非易失性存储器中读取一个或多个缺省值，以及
 经由一个拉上或拉下式的电阻器接收一个或多个缺省值做为限定值；以及

当响应处理器在重置时，在一个或多个寄存器内写入一个或多个的缺省值以做为其一个或多个的不同内含值，其中该一个或多个的不同内含值是从下列的组集中选取出来，该组集包含：
25

一个或多个的HDT激活位；

一个或多个的HDT激活锁码位；

一个或多个的微码装载机激活位；以及

一个或多个的微码装载机激活锁码位。

用于在个人计算机系统中关闭后门访问的机制

5 技术领域

广义来说，本发明涉及于计算机系统，具体讲，涉及保护个人计算机系统免于遭受未经授权的后门访问侵入的方法和装置。

背景技术

10 本申请案是审查中相关美国专利申请编号09/852,372的部分连续申请案，该申请专利的标题是Secure Execution Box and Method，乃于2001年5月10日送件，而发明者为Dale E. Gulick 和Geoffrey S. Strongin。本申请案也是审查中相关美国专利申请编号09/852,942的部份连续申请案，而该申请专利的标题是Computer System Architecture for
15 Enhanced Security and Manageability，该申请专利于2001年5月10日送件，而发明人是Geoffrey S. Strongin和Dale E. Gulick。

图1显示示例性计算机系统100，该计算机系统100包括处理器102、北桥104、内存106、先进绘图埠(Advanced Graphics Port, AGP)存储器108、外围装置内接(Peripheral Component Interconnect, PCI)总线110、
20 南桥112、电池113、AT附接(AT Attachment ATA)接口114(一般更常知的名称为整合驱动电子(Integrated Drive Electronics, IDE)接口)、共同串联总线(universal serial bus(USB))接口116、低位接脚计数(Low Pin Count, LPC)总线118、输入/输出控制芯片(超I/OTM)120、以及BIOS存储器122。注意北桥104和南桥112也可以只包括一颗芯片，或一组的芯片，即所
25 谓的集合名辞「芯片组」。同时也可以注意到，其它类型的总线、装置、及/或子系统也可以依照需要而包含在计算机系统100之内，例如高速缓存(caches)、调制解调器(modems)、平行(parallel)或序列(serial)接口、SCSI接口、网络适配卡等等。【超I/O是美国加州Santa Clara国家半导体公司(National Semiconductor Corporation)的注册商标】

30 处理器102和北桥104连接在一起。北桥104可以作为处理器102、内存106、AGP内存108、以及PCI总线之间的接口，而南桥112则作为

PCI总线110和连接到IDE接口、USB接口116和LPC总线118等的接口设备、装置及子系统之间的接口。图中的电池113和南桥112相连，超I/OTM芯片120则和LPC总线118连接在一起。

北桥104的作用是提供一种处理器102、内存106、AGP内存108、
5 连接到PCI总线110的装置、以及连接到南桥112的装置与子系统之间的连络访问管道。一般而言，可移除式的外围装置系插入PCI沟槽内(此部份未于图中显示出来)，而这些沟槽和PCI总线110相连，因此也和计算机系统 100相通。另一方式则将装置放在主机板上，因此可以直接和PCI总线110相连接。

10 南桥112的作用是提供一种 PCI总线110和各式装置及子系统之间可用的接口，所谓的各式装置及子系统如调制解调器、打印机、键盘、鼠标等，这些通常通过LPC总线118(或是它的前代产品，如X-总线或ISA总线)而和计算机系统100相连接。南桥112包括用做装置和计算机系统100其它部份之间接口的逻辑，而它们的连络方式可以通过IDE接口
15 114、USB接口116、及LPC总线118。

从软件的观点而言，x86操作环境在使用者隐私方面只能提供一点点的保护，同样在企业隐密和资产的安全性、或内容信息提供者的拥有权上也只能提供有限的保护。所有这些希求的目标，隐私(privacy)、安全(security)、和拥有权(ownership)(合称PSO)在网络互连计算机的时
20 代都变得非常关键，而原始个人计算机的设计并没将PSO的需求列入考虑。

从硬件的观点而言，x86操作环境对PSO而言是相当欠缺的，它让硬件可以经由软件，或者就由掀开个人计算机的机壳，而轻易直接地侵入，因此会让入侵者或窃取者破坏大部份的安全软件和装置。为了
25 方便使用，个人计算机事实上增加了PSO上的问题。

发明内容

根据本发明，提供一种处理器，该处理器包括第一寄存器(register)，而该寄存器被设定成可储存一个或多个硬件除错测试
30 (hardware-debug-test, HDT)激活位，该处理器还包括一个相连的第一控制逻辑以便接收多个HDT输入信号，以及包括和第一寄存器相连的第

二控制逻辑。第一控制逻辑的连接方式使其可以访问第一寄存器，而第二控制逻辑被设置成使其当该处理器在重置(reset)时，可以在第一寄存器内储存一个或多个缺省值。

根据本发明，提供一种确定HDT模式(mode)激活状态的方法，该方法包括接收一个请求，然后启动HDT模式，确定HDT模式激活状态，
5 以及如果HDT模式状态设定是激活时，则启动该HDT模式。

根据本发明，提供一种修改微码的方法，该方法包括接收一个请求以修改微码，决定微码装载机激活状态，以及如果该微码装载机激活状态设定为激活时，则修改微码。

10 根据本发明，提供一种改变HDT模式状态的方法，该方法包括接收一个请求以改变HDT模式状态，决定HDT模式激活锁码状态，以及如果该HDT模式激活锁码状态设定成解除时，则修改该HDT模式状态。

根据本发明，提供一种改变微码装载机激活状态的方法，该方法包括接收一个请求以改变微码装载机激活状态，决定微码装载机激活锁码状态，
15 以及如果该微码装载机激活锁码状态设定成解除时，则修改该微码装载机激活状态。

根据本发明，提供一种操作处理器的方法，该方法包括从下述组中选取一个或多个缺省值，所述组包含从一个或多个非易失性存储器中所读取的一个或多个缺省值，该方法又包括从拉上(pull-up)或拉下
20 (pull-down)式的电阻器(resistor)中接收一个或多个缺省值以做为限定值(strapped value)。该方法又包括回应处理器的重置而将一个或多个缺省值写入一个或多个寄存器内，以做为这些寄存器内一个或多个不同的内含值。这些一个或多个不同的内含值从含有一个或多个HDT激活位、一个或多个HDT激活锁码位、一个或多个微码装载机激活位、以及一个或多个微码装载机激活锁码位等的组集中挑选出来。
25

由于本发明可以有不同的应用形式和变化，在本文中所叙述的特殊实施例虽然以图解的方式加以详细说明，但是应该可以认知的，本文中所叙述的特殊实施例并不欲用来限定本发明的包含范围，相反地，
30 本发明应该包括所有可能修改、等效和其它的形式，而这些都在附属的专利保护范围声明的发明精神和范围中。

附图说明

由参考下列说明并配合所附图式而可了解本发明，其中相同的参考数字系表示相同的组件，而在所附图式中：

图1展示先前计算机系统采用技术的方块图；

5 图2展示根据本发明的一个概念，于附有安全执行盒(secure execution box)的计算机系统中，所建构的数据和命令的实施例的流程图；

图3展示根据本发明的一个概念，建构附有SMM MSRs的计算机系统的实施例的方块图；

10 第4A至4D图展示根据本发明的各不同概念，建构包含锁码寄存器和逻辑的处理器实施例的方块图；

图5展示根据本发明的一个概念，用以启动HDT 模式的方法实施例的流程图；

15 图6展示根据本发明的一个概念，用以改变HDT激活状态的方法实施例的流程图；

图7展示根据本发明的一个概念，用以启动微码装载器的方法实施例的流程图；以及

图8展示根据本发明的一个概念，用以改变微码装载器激活状态的方法实施例的流程图。

20

具体实施方式

本发明的实施例的详细说明现叙述于下。为了简单明了的缘故，并非所有真实建构体的细节都在本文的特例中完整的叙述。当然可以认知的，于类似真实的实施例在发展过程当中，都会有许多实现特殊的状况需要加以判断，以达成原来发展者的特殊目的。譬如说，符合系统相关和商业相关的限制条件，诸如此类，都会使实施例转成实体时会因时因地而异。再者，也应该可以认知的，像这样发展时所做的投入可能会相当复杂和费时，但对本领域的专业人士，无论如何都会变成一项例常的操作，而本文所描述的内容对他们也会有很大的帮助。

25 相关联于参考号码所使用的字母是欲用来表示连接该参考号码的该项目的一种替代的实施例。

30

在内建有节约能源的计算机系统中，系统管理模式(System Management Mode, SMM)是计算机系统的一种运行模式。SMM是因应第四代x86处理器而产生的。之后所出现更新代的x86处理器，SMM就已经变成必备而不需操作系统的特别处理，也就是说，计算机系统的
5 进出SMM已经对操作系统只有些微甚或完全没有干扰。

图2展示根据本发明的一个概念，在附有安全执行盒260的计算机系统中，所建构的数据和命令的实施例的流程方块图。使用者的输入和输出(I/O)数据以及/或者命令205可以馈入一个或多个应用程序210，也可以从这些应用程序中撷取使用者的输入和输出(I/O)数据以及/或者
10 命令。应用程序210在计算机系统内和密码服务供应软件(cryptography service providers)215交换数据和命令，而所谓的计算机系统如计算机系统100或其它的计算机系统。密码服务供应软件215可以利用应用程序化接口(API)调用220与驱动程序(drivers)225交互，而驱动程序可以提供到硬件230的访问。

15 根据本发明的一个概念，驱动程序225和硬件230是安全执行盒的一部份，而安全执行盒设定在安全执行模式(secure execution mode, SEM)260的状态下操作。被信任的隐私、安全和拥有权(PSO)操作，也简单称为安全操作，可以当计算机系统在SEM 260时展开操作。从使用者I/O205和/或应用程序210传送而来的软件调用可以在SMM260中，
20 然后放置于安全执行盒内。软件调用可以进入SEM 260中的安全执行盒，然后进到不同的安全硬件资源。

图3展示根据本发明的各不同概念，所建构附有SMM MSRs的计算机系统300的实施例的方块图。第3 图包括处理器305、北桥310、内存306、以及南桥330。该处理器包括一个或多个SMM机器特定寄存器
25 (MSRs)307。北桥310包括内存控制器315。北桥310连接在处理器305和南桥330之间，连接处理器305是通过本地总线308，而连接南桥330则是通过PCI总线110。北桥310的连接方式是要从处理器305处接收SMI $\overline{\text{ACT}}$ #信号。

在图3的实施例中，计算机系统300会利用标准处理器信号(例如，
30 给北桥310的信号是SMI $\overline{\text{ACT}}$ #)和/或在本地总线308及PCI总线110上的总线循环(bus cycles)而发出信号，以表示该处理器305是在SMM状态。

在本发明的不同实施例中，我们假定SMM的安全机制。我们注意到有一种或多种后门(backdoor)可用以破坏SMM的安全机制。即将要考虑的问题有处理器305的硬件除错测试(HDT)模式的误用，以及处理器305装载和置换微码的能力。第4A至4D图展示处理器305不同的实施
5 例，305A、305B、305C、305D，每一种实施例包括用以对抗一种或多种后门攻击的不同安全防护。

在图4A中，处理器305A包括HDT控制逻辑410A、HDT重置逻辑420A、以及一个或多个寄存器，该寄存器包括一个HDT激活寄存器415和非易失性随机访问内存(NVRAM)430。如图所示，HDT控制逻辑410A
10 的连接方式是可以通过多个HDT接脚405接收多个输入信号。HDT控制逻辑410A又连接到HDT激活寄存器415。HDT重置逻辑420A的连接方式使其可以在线425上接收一个RESET信号，以及访问(即读和写)HDT激活寄存器415和NVRAM430。

在图4B中，处理器305B包括HDT控制逻辑410B、HDT重置逻辑
15 420B、以及两个寄存器，包括有HDT激活寄存器415和HDT激活锁码寄存器435。如图所示，HDT控制逻辑410B的连接方式是可以通过多个HDT接脚405接收多个输入信号。HDT控制逻辑410B又连接到HDT激活寄存器415和HDT激活锁码寄存器435。HDT重置逻辑420B的连接方式使其可以在线425上接收一个RESET信号，以及经由拉上(或拉下)式的
20 电阻器445而在线440上接收信号。

在图4C中，处理器305C包括微码控制逻辑455、微码装载器激活重置逻辑465、以及一个或多个寄存器，该寄存器包括微码装载器激活寄存器460。如图所示，微码控制逻辑455的连接方式是可以通过多个HDT
25 接脚405接收多个输入信号。微码控制逻辑455又连接到微码装载器激活寄存器460。微码装载器激活重置逻辑465的连接方式，使其可以接收重置(RESET)信号，也可以访问微码装载器激活寄存器460。

在图4D中，处理器305D包括和微码控制逻辑455整合的HDT控制逻辑410、HDT重置逻辑420、以及用以形成控制/重置逻辑475的MLE重置逻辑465。HDT激活寄存器415和微码装载器激活寄存器460整合至
30 一个多位锁码寄存器480内。图中显示多个送入控制/重置逻辑475的输入470。该多个输入470可以包含HDT输入405、微码输入450、和/或重

置信号平均值。其它的实施例(未于这里展示)只整合了HDT控制逻辑410和微码控制逻辑455, 或者只有HDT重置逻辑420和MLE重置逻辑465。

根据本发明中所提出的不同的实施例, 寄存器415、435、460、以及NVRAM 430等, 包括用以储存一个或多个位的储存空间。在其中的一个实施例中, 每一个寄存器被设置成可以储存一个信号位。注意激活寄存器415和460 可以整合至一单独的锁码寄存器, 而HDT激活锁码寄存器435可以用来当成一个微码激活锁码寄存器。目前业界正考虑将寄存器415、435、460、和/或480包含在SMM MSRs 307内。

在不同的实施例中, HDT激活寄存器415设定成储存一个或多个HDT激活位, 该位可以发出信号以表示是否HDT模式是处于激活或失效(disabled)的状态。HDT重置逻辑420 被设置成使其在处理器305的重置下, 可以设定一个或多个的HDT激活位至预设状态。

控制HDT模式的多样实施例也可予以考虑, 如图4A和图4B所示。在其中的一个实施例中, HDT模式在非量产的处理器305上激活成预设的模式, 这里的非量产的处理器指的是供工程和测试用的处理器。HDT模式也可以在标准量产的处理器305上预设成失效的模式。在图4A所示的另一个实施例中, 预设的状态可以储存在NVRAM 430当中, 也可以从当中读取出来。在此实施例中, 预设状态是可以改变的, 但在这里用以说明的实施例中, 其缺省值设成失效。如图4B所示的另一个实施例, 其预设状态是利用吊带(strapping)选项而设定。提供给HDT重置逻辑420B的缺省值是经由拉上(或拉下)电阻器445而进入的。

控制微码装载机模式的多个实施例也可予以考虑, 如图4C和4D所示。在其中的一个的实施例中, 微码更新模式在非量产的处理器305上激活成缺省值, 这里的非量产的处理器指的是供工程和测试用的处理器, 而此部份未以图展示。微码更新模式在标准量产的处理器305上可以预设成失效的模式。在另一个类似图4A所展示的实施例中, 预设的状态可以储存在NVRAM 430当中, 也可以从当中读取出来。在这个实施例中, 预设状态是可以改变的, 但在这里用以说明的实施例中, 其缺省值设成失效。如图4B所示的另一个实施例, 其预设状态是利用吊带选项而设定。提供给MLE重置逻辑465的缺省值系经由拉上(或拉下)

寄存器445而进入的。

转至图5，所展示的是一个用以起动HDT模式的方法500。根据所接收到进入HDT模式(步骤505)的请求，HDT控制逻辑410会检查一个或多个HDT激活位的状态，然后会知道HDT模式是处于激活抑或失效的状态(步骤510)。如果HDT模式是激活状态(步骤515)，那么HDT控制逻辑410会起动HDT模式(步骤520)，而如果HDT模式是失效状态(步骤515)，那么HDT控制逻辑410就不会起动HDT模式。

现在转至图6，所展示的是一个用以改变HDT模式激活状态的方法600，该方法包括如图所示的一个HDT模式锁码。根据所接收到进入HDT模式(步骤605)的请求，HDT控制逻辑410会检查一个或多个HDT激活锁码位的状态，并由此确定HDT锁码模式是否锁码，或者是解除锁码(步骤610)。如果HDT锁码模式是在解除锁码(步骤615)的状态，那么HDT控制逻辑410会起动HDT模式(步骤635)。如果HDT锁码模式是在锁码(步骤615)的状态，那么HDT控制逻辑410会要求授权以改变HDT锁码模式的状态(步骤620)。如果授权了改变动作(步骤625)，则HDT控制逻辑410改变HDT模式锁码位成解除状态(步骤630)。如果改变的动作未予授权，那么HDT控制逻辑410就不会改变HDT模式锁码位。

在不同的实施例，HDT激活状态可以通过设定或者重置一个或多个的HDT激活状态位的方式而改变。例如，HDT模式可以是失效的，但在SMM内部，一个预定进入HDT控制逻辑410的输入可以对HDT控制逻辑410发出信号，使其将HDT模式状态改变成激活的状态。又如图4A的实施例中，一旦信号通知了HDT控制逻辑410，它就会将HDT激活位从失效的状态改变成激活的状态。

回至图4B的实施例，为响应接收的改变HDT模式的请求，HDT控制逻辑410会检查一个或多个HDT激活锁码位的状态，并由此决定HDT锁码模式是否锁码，或者是解除锁码。如果HDT锁码模式是在解除锁码的状态，那么HDT控制逻辑410就可以改变HDT模式状态。反之如果HDT锁码模式是在锁码的状态，那么HDT控制逻辑410就不会改变HDT模式状态。

注意当HDT锁码模式是锁码时(步骤615)，方法600也可以中断而不用要求授权以改变HDT锁码模式的状态(步骤620)。在方法600要求授权

检查之前(步骤620),方法600也可以包括接收改变HDT锁码模式状态的请求(未于图中显示出来)。

现在转至图7,所展示的是起动微码装载器的方法700。根据所接收起动微码更新模式的请求(步骤705),微码控制逻辑455会检查一个或多个微码激活位的状态,并由此知道微码更新模式是在激活或者失效的状态(步骤710)。如果微码更新模式是处于激活的状态(步骤715),那么微码控制逻辑455就会起动微码更新模式(步骤720)。反之如果微码更新模式是在失效的状态(步骤715),那么微码控制逻辑455就不会启动微码更新模式。

10 现转至图8,所展示的是改变微码更新模式激活状态的方法800,该方法包括微码模式锁码。根据所接收进入微码模式的请求(步骤805),微码控制逻辑455会检查一个或多个微码激活锁码位的状态,并由此知道微码模式是在锁码或者解除的状态(步骤810)。如果微码锁码模式是处于解除的状态(步骤815),那么微码控制逻辑455就会起动微码模式(步骤835)。反之如果微码锁码模式是在锁码的状态(步骤815),那么微码控制逻辑455就会要求授权认定以改变微码模式锁码状态(步骤820)。如果改变是经过授权的(步骤825),那么微码控制逻辑455就会改变微码模式锁码位至解开的状态(步骤830)。如果该改变是未经授权的(步骤825),那么微码控制逻辑455就不会改变微码模式锁码位。

20 在不同的实施例中,微码激活状态可以通过设定或重置一个或多个微码激活状态位的方式而改变。例如,微码模式可以是失效的,但在SMM内部,一个预定进入微码控制逻辑455的输入可以对微码控制逻辑455发出信号,使其将微码模式状态改变成激活的状态。如图4C的实施例中,一旦信号通知了微码控制逻辑455,它就会将微码激活位从失效的状态改变成激活的状态。

根据所接收改变微码模式状态的请求,微码控制逻辑455会检查一个或多个微码激活锁码位的状态,并由此判定微码锁码模式是在激活或者失效的状态。如果微码锁码模式是处于失效的状态,那么微码控制逻辑455就可以改变微码模式状态。反之如果微码锁码模式是在失效的状态,那么微码控制逻辑455就不会改变微码模式状态。

30 注意当微码更新锁码模式是锁码时(步骤815),方法800也可以中断

而不用要求授权以改变微码更新锁码模式的状态(步骤820)。在方法800要求授权检查之前(步骤820)，方法800也可以包括接收改变微码更新锁码模式状态的请求(未于图中显示出来)。

5 为了公开的目的而参照ROM的部份也同样适用于闪存和其它实质上是非易失性的存储器型式。注意本发明所公开的方法中有些展示成流程图，而流程图上的各式组成组件也可能视情况而省略，或者以不同的次序而动作。同时也注意本发明所公开的方法也可以在执行时有不同的改变。

如前面公开的本发明，其中的特点可以硬件或软件的方式执行。
10 因此之故，这里所详述的某些部份是以硬件执行程序的方式呈现，而某些细述的部份则以软件执行程序的方式呈现，软件执行程序包括计算机系统或计算机装置的存储器内数据位操作的符号表示。所有这些描述和表示都在本领域的专业人士用来表达硬件和软件的专业术语，硬件和软件的程序和操作都牵涉到物理量上的运用。对软件而言，
15 这些物理量虽然不必然，但也通常会以电、磁、或光的信号的方式而表示，而这些形式的信号可以被储存、转移、合并、比较、以及其它的运用等。为了共通表达的因素，也为了方便的缘故，这些形式的信号有时就以位、数值、组件、符号、字符、项目、数字、或其它之类等来表达。

20 所有这些或类似的项目名称都只是一种方便的记号，用以代表其相关的物理量，除非有特别的指明，否则它们的用法在本发明公开的申请专文中应用相当显明，而这些叙述或是用来陈述一个电子装置的动作和程序，该电子装置可以运用及转变由物理(电、磁、或光)量所代表的物理量，也就是将某些电子装置内储存的物理量所代表的物理量转变成另一种在储存装置、传输或显示器中，而由类似物理量所能代表的
25 数据。这些用以代表的项目名称，举例而言，有程序(processing)、计算(computing)、计算(calculation)、确定(determining)、显示(displaying)等等及其它类似的名称。

要注意的是，本发明中有关软件执行的特点以一般而言是以某种
30 程序储存物体的形式编码，或者是执行在某种传输物体之上。程序储存物体或许是磁性(例如，软盘或硬盘)或光学(例如，只读光盘或CD

ROM)的,而且可以是只读或随机访问的。同样地,传输物体可以是双绞线、同轴电缆、光纤、或者是其它本领域适合的物质。本发明并不限定这些在实体化时所用的物质或材料。

前面所阐述的特定实施例只是用来说明之用,所以因为本发明的说明而受益的本领域专业人士,可能以不同但实质是等效的方法将其修改和应用。此外,本文所详细说明的结构或设计都非用来限定其形式,其它不同于阐述内容的部份将于后面的专利保护声中说明。因此很明显的,上述所揭露的特定实施例可以变换或修改,而且所有类似的变形都视为本发明的范围和精神之内。因此之故,所要保护部分将在权利要求中以文字说明。

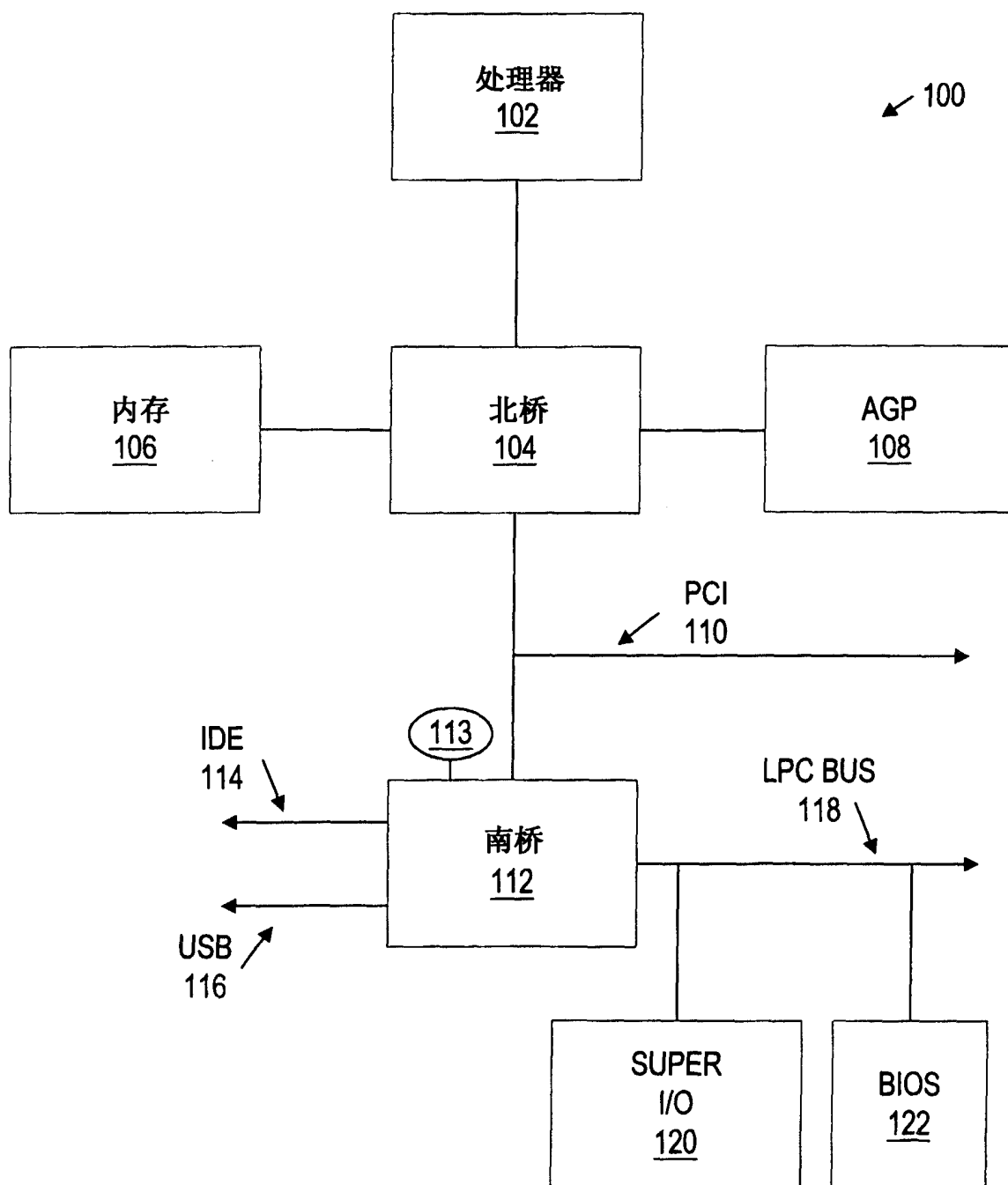


图1
(已知技术)

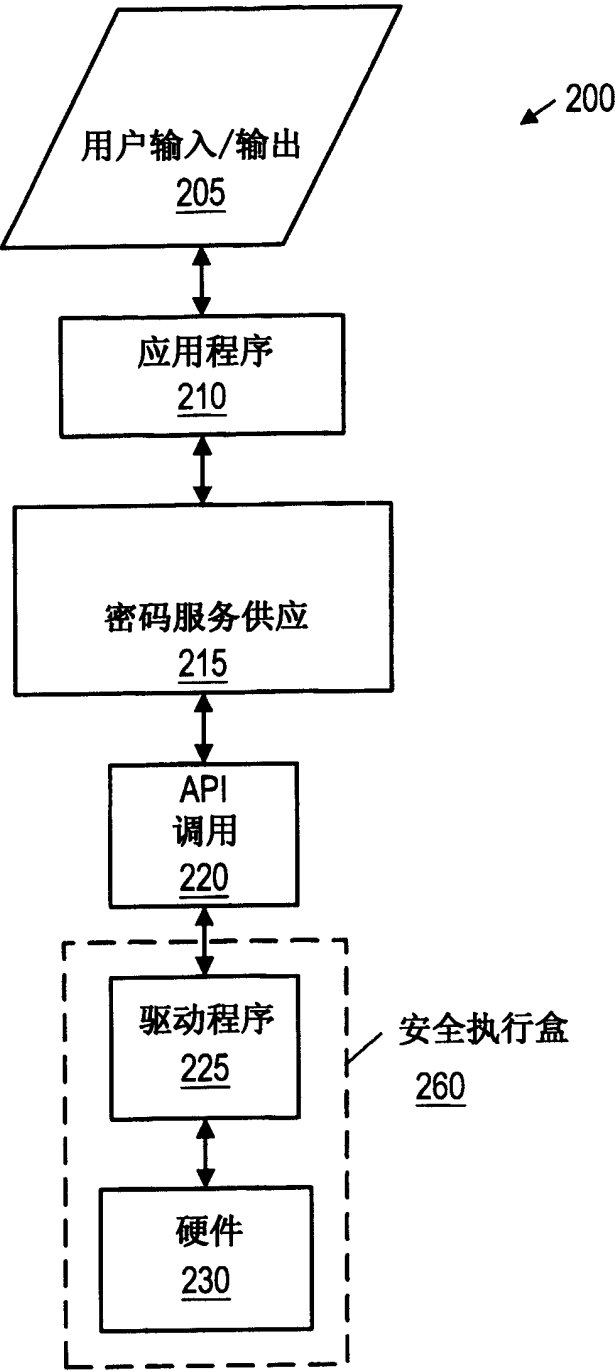


图2

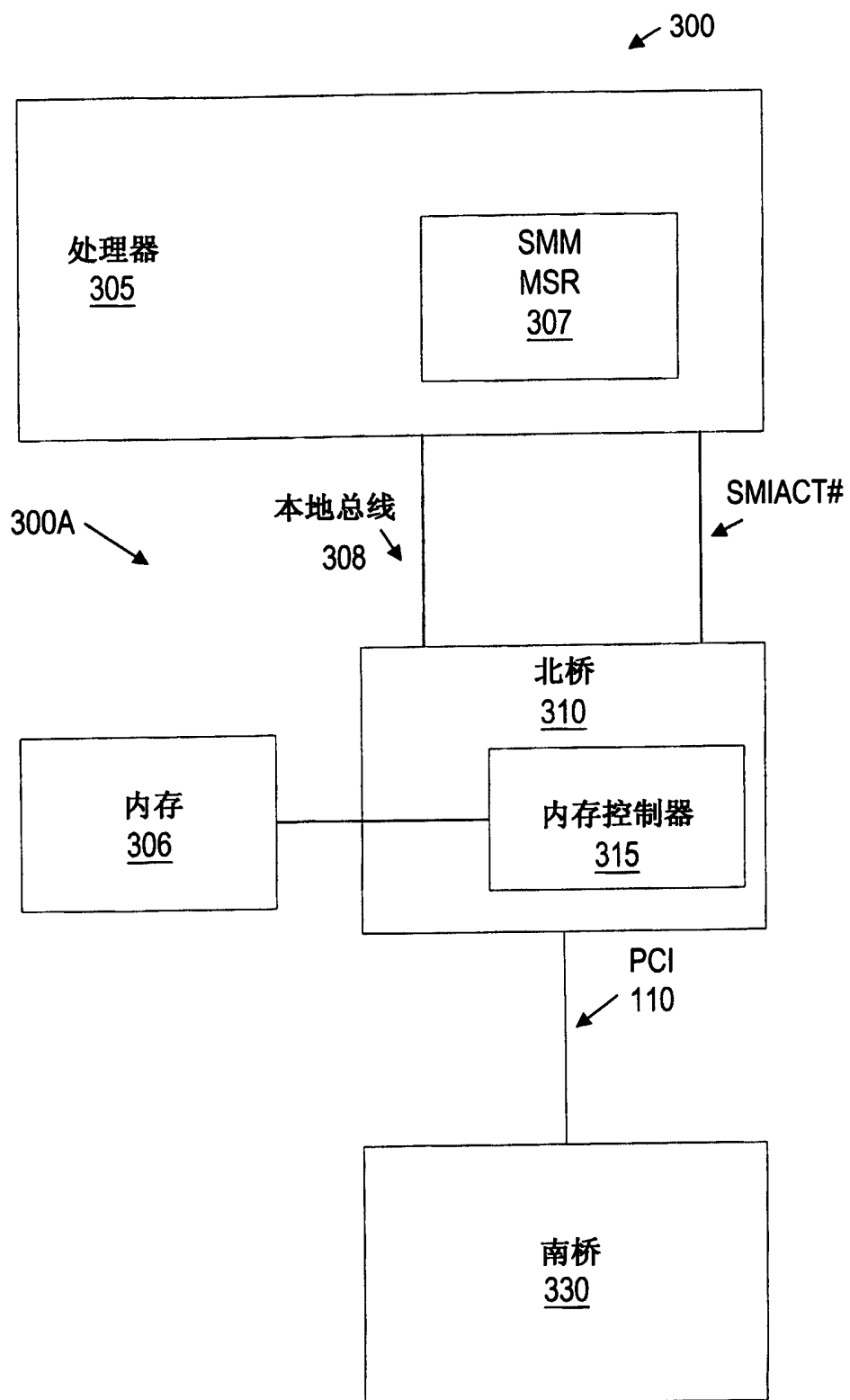


图3

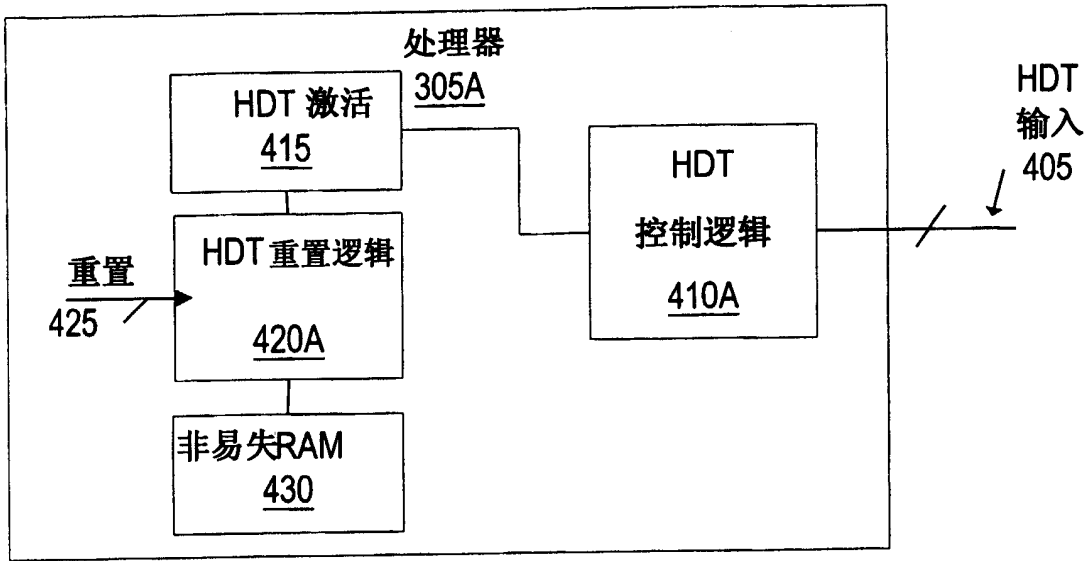


图4A

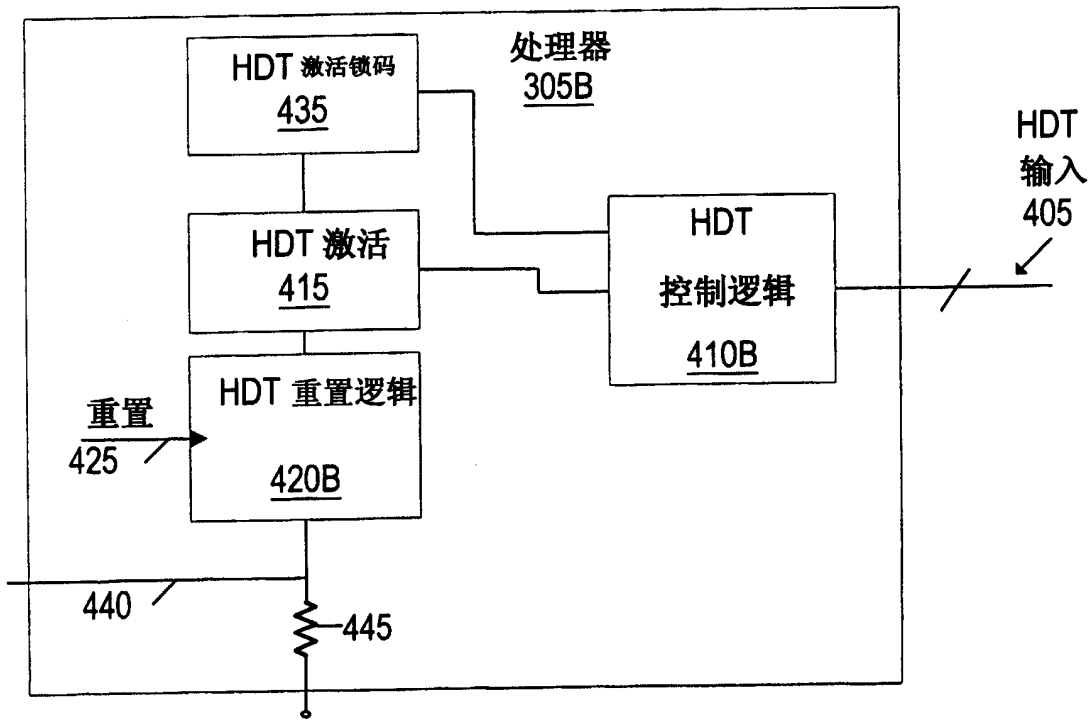


图4B

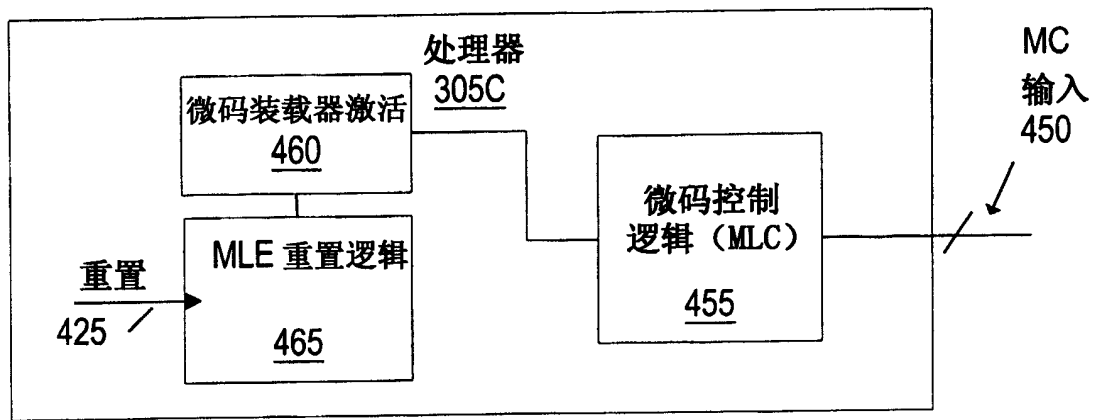


图4C

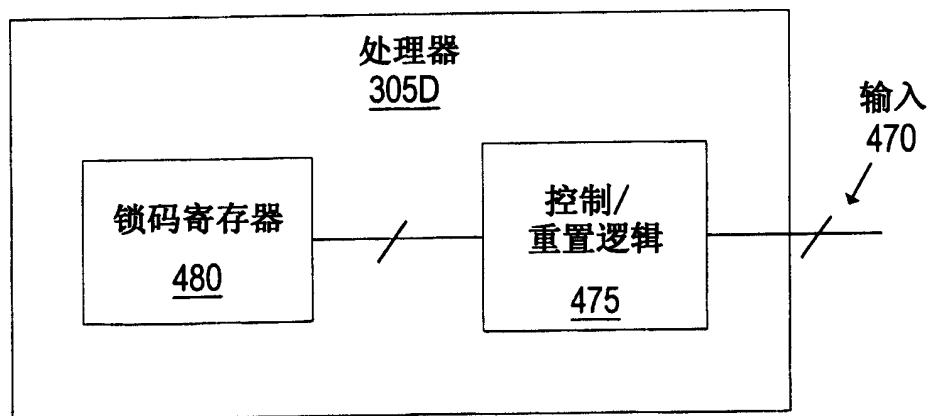


图4D

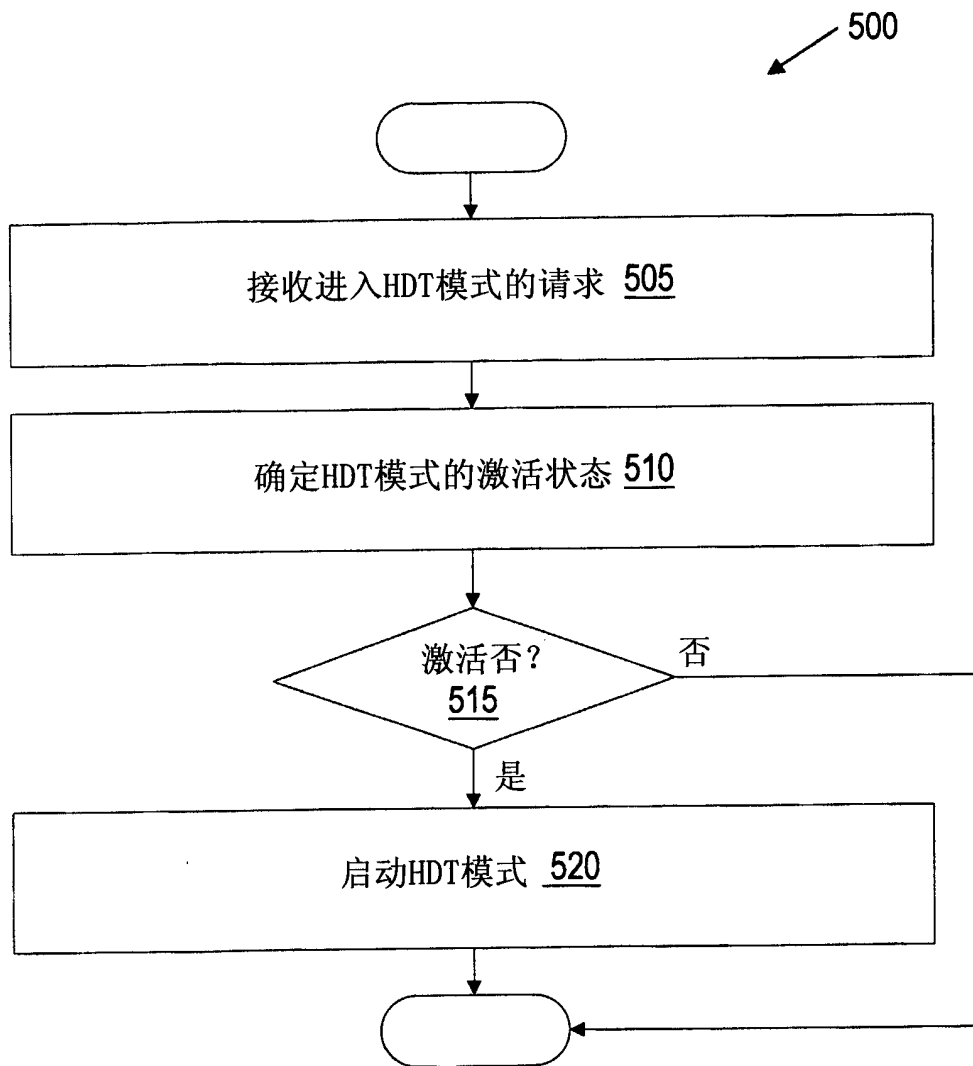


图5

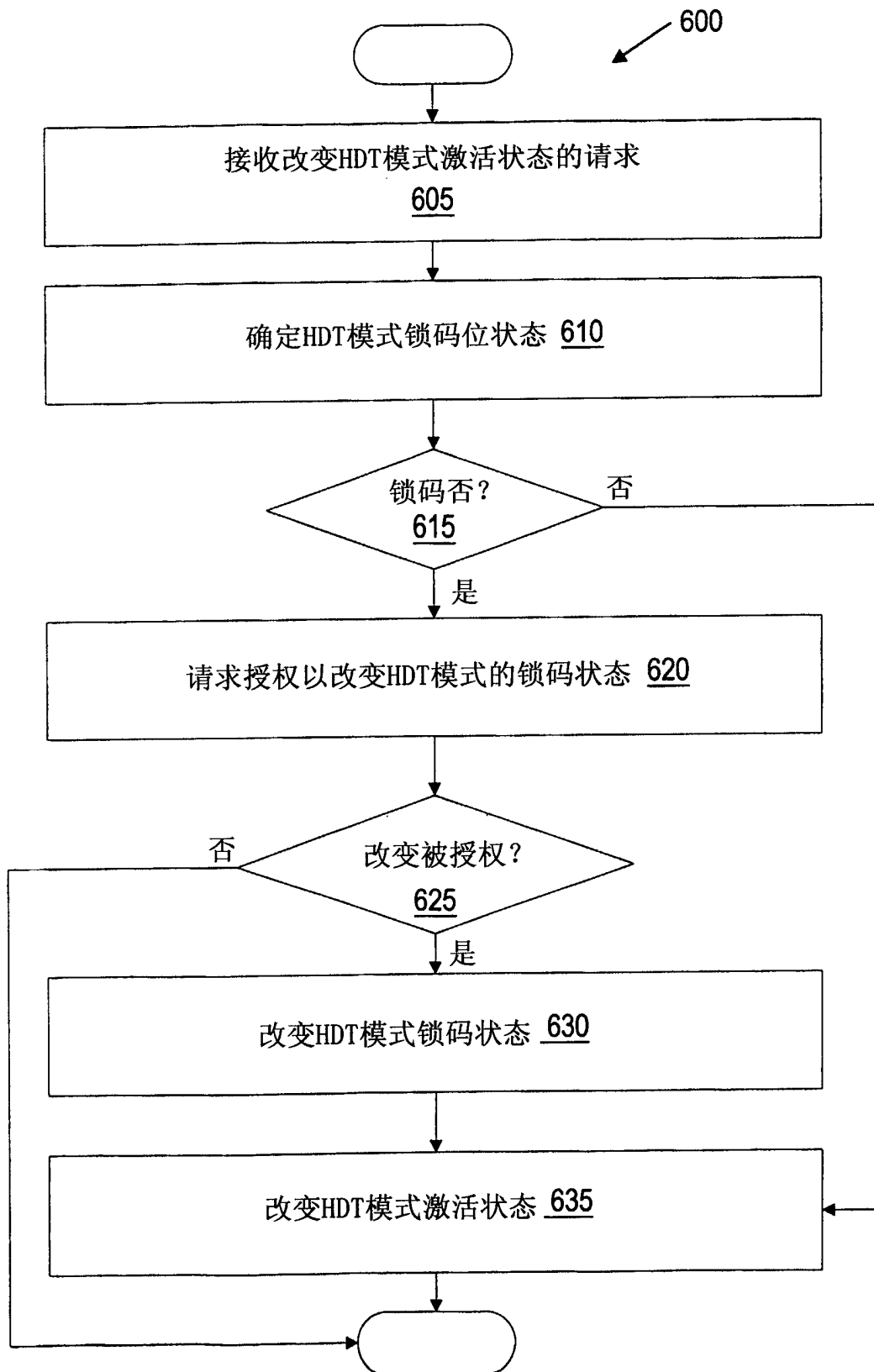


图6

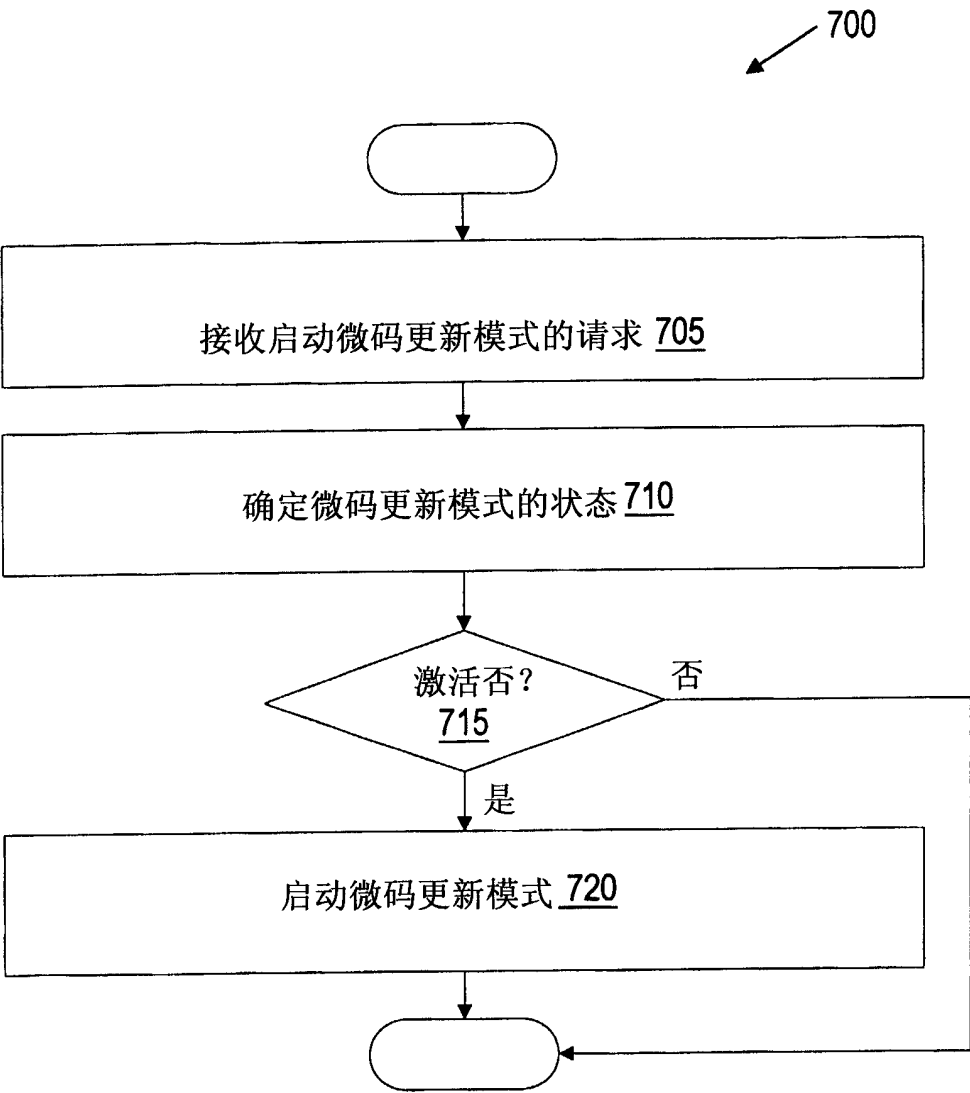


图7

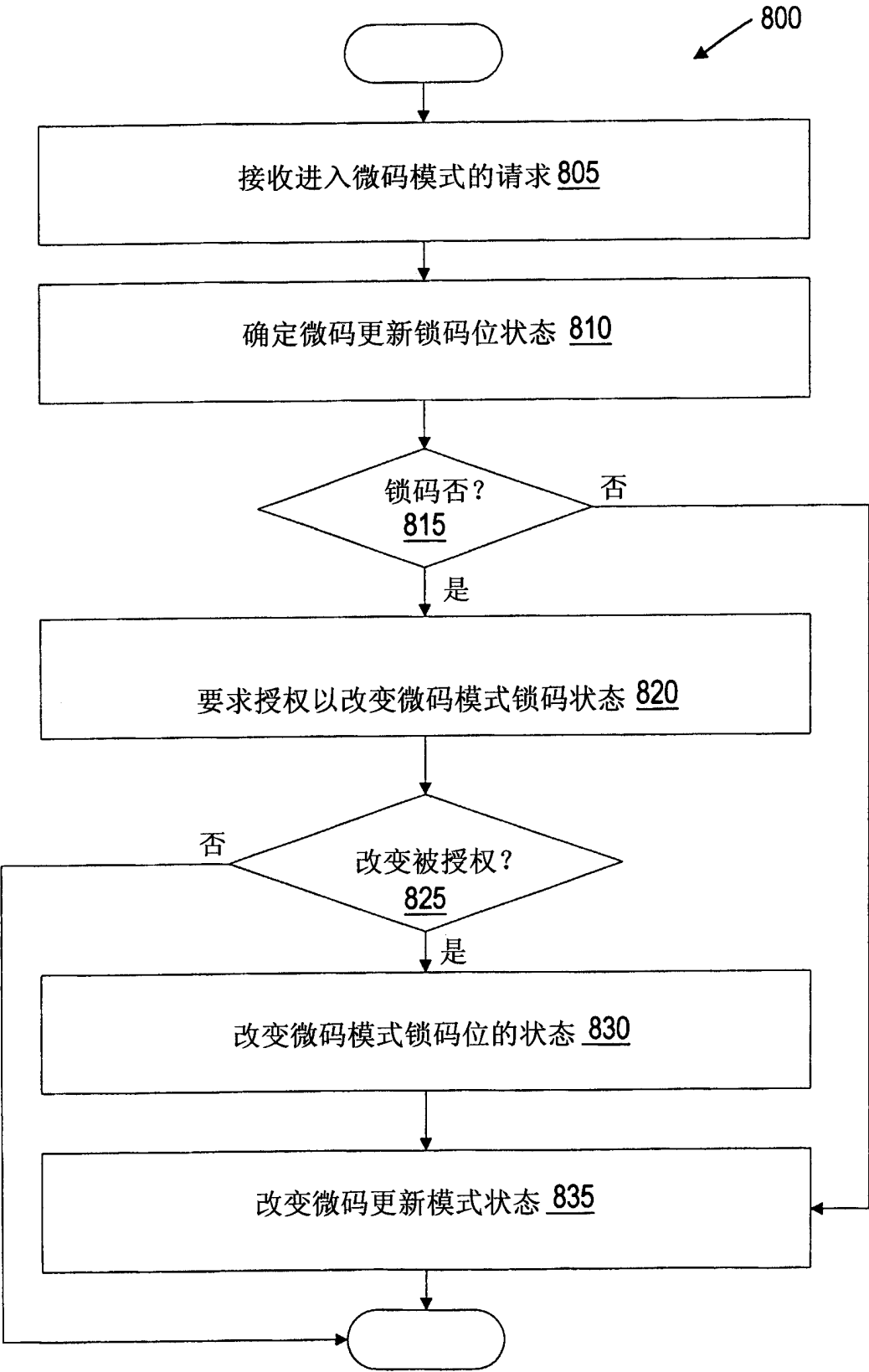


图8