[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 11/30 (2006.01)
G06F 12/14 (2006.01)
H04L 9/32 (2006.01)

[12] 发明专利申请公开说明书

[21] 申请号 200480018214.6

[43] 公开日 2006 年 8 月 2 日

[11] 公开号 CN 1813244A

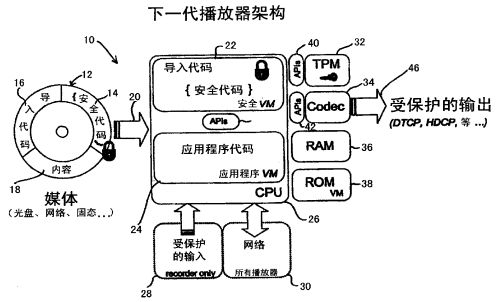
[22] 申请日 2004.6.28
[21] 申请号 200480018214.6
[30] 优先权
[32] 2003. 6.27 [33] US [31] 60/481,034
[32] 2003. 7. 7 [33] US [31] 60/481,066
[32] 2003. 8. 5 [33] US [31] 60/493,072
[86] 国际申请 PCT/US2004/021048 2004.6.28
[87] 国际公布 WO2005/001666 英 2005.1.6
[85] 进入国家阶段日期 2005.12.27
[71] 申请人 迪斯尼实业公司
地址 美国加利福尼亚州
[72] 发明人 斯科特·沃森
[74] 专利代理机构 中科专利商标代理有限责任公司
代理人 朱进桂
权利要求书 2 页 说明书 6 页 附图 2 页

[54] 发明名称

下一代媒体播放器的双虚拟机以及信任平台

[57] 摘要

一种基于运算环境的软件，用于提供从网络下载或从媒体播放器载入的媒体的安全认证，所述软件包括两个对等模式操作的虚拟机。 低级虚拟机提供解码和解密功能，而高级虚拟机提供应用级功能，例如用户接口、输入/输出。



1、一种驻留在运算环境中的双虚拟机架构，所述架构包括：

低级虚拟机，用于执行至少解码媒体或提供安全功能之一；以及

高级虚拟机，用于执行应用级功能；

其中，低级虚拟机与高级虚拟机具有对等关系。

2、根据权利要求1所述的双虚拟机架构，其中，高级虚拟机包括用于与用户接口的应用程序。

3、根据权利要求1所述的双虚拟机架构，其中，应用级功能至少包括提供与用户的接口或与具有媒体的网络进行通信之一。

4、根据权利要求1所述的双虚拟机架构，其中，由高级虚拟管理器将来自媒体的安全代码发送到低级虚拟管理器，用于解密。

5、根据权利要求4所述的双虚拟机架构，其中，媒体至少从DVD、光盘、网络或固态设备之一是可用的。

6、根据权利要求1所述的双虚拟机架构，还包括处理模块，处理模块包括至少一个解密密钥。

7、根据权利要求6所述的双虚拟机架构，其中，处理模块执行安全密码运算。

8、根据权利要求6所述的双虚拟机架构，其中，处理模块被用于监控运算环境。

9、根据权利要求1所述的双虚拟机架构，其中，运算环境包括中央处理单元（CPU）。

10、一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，该方法包括步骤：

设置低级虚拟机，用于执行安全功能；

设置高级虚拟机，用于执行用户接口和应用级功能；

其中，低级虚拟机与高级虚拟机具有对等关系。

11、根据权利要求10所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，还包括高级虚拟机具有用于与用户接口的应用程序。

12、根据权利要求10所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，其中，应用级功能至少包括提供与用户的接口或与具有媒体的网络进行通信之一。

13、根据权利要求10所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，还包括由高级虚拟管理器将来自媒体的安全代码发送到低级虚拟管理器，用于解密。

14、根据权利要求13所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，其中，媒体至少从DVD、光盘、网络或固态设备之一是可用的。

15、根据权利要求10所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，还包括处理模块，处理模块提供至少一个解密密钥。

16、根据权利要求15所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，其中，处理模块执行安全密码运算。

17、根据权利要求15所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，还包括由处理模块监控运算环境。

18、根据权利要求10所述的一种与运算环境无关的、为媒体播放器提供程序拷贝保护的方法，其中，运算环境包括中央处理单元(CPU)。

下一代媒体播放器的双虚拟机以及信任平台

技术领域

本发明涉及开发新的安全系统和方法，包括对可拆卸媒体播放器的拷贝保护。

背景技术

虚拟机（VM）是用于说明充当编译器和微处理器（或“硬件平台“）之间的接口、并实际执行程序指令的软件的术语。编译器是处理按照特定编程语言写出的陈述并且将其变为二进制机器语言的特殊程序或计算机处理器所使用的“代码”。

Java 编程语言和运行环境的开发者，Sun Microsystems，因为其 Java 虚拟机的开发而著名。Java 虚拟机为计算机处理器（或“硬件平台”）解释编译的 Java 二进制代码（称为字节代码），以便其可以执行 Java 程序指令。

Java 被设计为允许构建应用程序，应用程序可以在任意平台上运行而不需要程序员针对每一个独立平台重写或重新编译。当给平台提供 Java 虚拟机时，任何 Java 程序可以在该平台上运行。Java 虚拟机使这成为可能，因为它考虑到特定指令长度和平台的其它特性。

虚拟机是一种抽象的运算机器。与真实的运算机器一样，它具有指令设置并且在运行时操作各种存储区域。使用虚拟机来实现编程语言是相当普遍的;最著名的虚拟机也许是 UCSD Pascal 的 P-Code 机器。

另外，更一般地，可以将虚拟机描述为操作系统或运行计算机的任何程序。

在下一代媒体播放器中，例如 DVD 或 CD 播放器，长期需要发展拷贝保护的改进方法。

一个公知的用于 DVD 的内容安全系统是内容加扰系统（CSS），其中 DVD 上的数据被加密。然后当读盘时，DVD 播放器使用 40 比特解密密钥来解密数据。然而，CSS 的严重缺陷在于其密钥和算法是固定的。

反向设计加密算法，并且播放现有 DVD 盘的每一个可能解密密钥是可用的。当泄漏秘密时，系统将受到永远威胁，这是因为没有办法更新安全算法或密钥。现在，存在多个消费者可用的程序，其利用单个“点击”从 DVD 内容中去除所有安全性。

内容所有者不希望再次发生这种情况，尤其随着内容的逼真度增加。因此，下一代内容安全系统不应该这样易受攻击。

软件零售商也面临盗版问题，然而由于计算机的本质，他们采取不同于 DVD 娱乐公司所使用的方式。历史上，打包软件程序（例如计算机游戏）制造商利用“程序安全”来保护其内容。即，不存在用于保护程序的固定的预定方法，而是每一个软件厂商编写或获得“安全代码”来保护其内容。这种程序安全代码在复杂度和技术上根据程序不同而变化，但是最重要地，因为每一个程序具有不同的安全软件实施方式，不可能写出通用“去除安全”程序，就像写出用于攻击 DVD 安全的程序。

另一种公知的拷贝保护的方法是写硬件特殊指令。该方法的问题在于它是极其有限的。利用这种方法，对于每一个硬件配置，必须呈现不同组的指令。这有点不切实际。

因此，需要一种向例如媒体播放器（不是专用硬件）的硬件提供拷贝保护的方法。

发明内容

因此，向媒体播放器提供了一种与平台无关的程序拷贝保护的系统和方法。本公开提出一种解决方法，其中给下一代媒体播放器提供双虚拟机架构。本公开还提出使用基于硬件的嵌入式安全子系统，例如信任平台模块（TPM），来连接虚拟机架构的特定方面。

根据本公开的双虚拟机架构由高级虚拟机和低级虚拟机组成。低级虚拟机被设计用于支持低级媒体解密和解码功能，而高级虚拟机被设计用于处理应用层的行为。因此，这种架构从应用软件中分离出安全软件。

通常，最适于程序安全的虚拟机更近似于实际硬件 CPU 的指令组。即，其支持指针，并且在可执行代码和数据之间存在根本区别。因此，

这种第一类虚拟机被称为“低级 VM”，或“安全 VM”。低级虚拟机被设计为与支持抗篡改（tamper resistant）软件技术的传统 CPU 类似。

类似这种的虚拟机的不利方面在于编程错误或意外的运行条件是严重的。对于安全系统，这可以认为是强项，但是对于应用（更为复杂，并且通常具有强度较低的测试覆盖），这是缺点。

对于应用，“在后台”管理大部分运算细节的“高级 VM”允许开发按照更加可预测和鲁棒方式运转的更可靠的应用程序。“高级”虚拟机的典型范例是 Java。例如，Java 不需要支持“指针”的概念或明确的存储管理（编程错误的一般来源），然而支持“例外处理”，按照可预测方式帮助程序和程序员处理意外运行条件。

高级或应用级虚拟机被设计具有所有特点，并且提供丰富的应用接口。

因此，理想的是组合低级 VM 和高级 VM 的优点，以便提供与其它应用结合进行工作的平台无关安全功能。此外，信任平台模块通过安全地询问和验证执行环境，提供基于硬件的信任基础。

在典型实施例中，本公开被用于 DVD 和 CD 播放器中的可更新安全和拷贝保护的目的。然而，这种架构还支持存储在硬盘、固态存储器中或在网络上发送的媒体的回放。

如上所述，低级虚拟机被设计用于支持低级媒体解密和解码功能。在下一代媒体（NGM）应用中，这种低级虚拟机还负责引导高级 VM。高级 VM 处理应用层的行为，例如高级用户接口、misc、IO 和网络行为。

本公开的双 VM 架构是新型的。双 VM 架构提供“对等”关系而不像传统的“堆叠（stacked）VM”关系。一个 VM 在另一个之上运行的堆叠关系的一个范例是运行窗口仿真程序（x86 仿真程序或 VM）的 PowerPC（如在 Mac 中一样），窗口仿真程序依次执行 JavaVM。

此外，这种双 VM 架构与基于硬件的信任运算模块的结合是创新的。

本公开将程序安全的使用扩展到例如 CD 和 DVD 的媒体。此外，程序安全还允许内容持有者具有比公布的系统更灵活的版权管理。与现有技术固定安全系统（类似 CSS）所提供的简单拷贝保护（CP）相对，

这种灵活性可以被用于实现完备的数字版权管理（DRM）系统。

下面的实施例示出了本公开的特点和优点，从阅读典型实施例的详细说明中，显而易见本发明的上述及其它目的、特点和优点。

附图说明

图1是根据典型实施例在运算环境中的媒体播放器架构的图。

图2是示出了根据典型实施例的低级虚拟管理器和高级虚拟管理器的交互和功能性的方框图。

需要理解，为了演示的简化和清楚，不必按比例绘制图中所示的单元。例如，为了清楚，相对于彼此，放大一些单元的尺寸。此外，适当考虑时，在附图中重复参考数字来表示相应的单元。

具体实施方式

在此公开详细的说明；然而，要理解到，公开的实施例只是本发明的范例，本发明可以按照各种形式实现。因此，在此公开的特殊结构和功能的细节不应该被理解为限制，而只是作为权利要求的基础并作为代表基础，教导本领域技术人员实际按照任意适当详细的结构来不同地使用本公开。详细参考在附图（图 1-2）中所示的本公开。

本公开的系统和方法提供使用媒体播放器的双虚拟机架构。一个 VM 被设计用于支持安全功能，例如媒体解密和解码。在下一代媒体应用中，低级 VM 也许负责引导应用级 VM。高级或应用级 VM 处理应用层的行为，例如高级用户界面、misc、IO 和网络行为。

图 1 和 2 示出了根据典型实施例的运算环境 10 中的媒体播放器架构。具体地，示出了包括媒体数据或内容 18 的媒体源（例如 DVD、光盘、固态设备或网络）、用于允许在媒体播放器上回放媒体的安全代码 12 以及导入代码 16。

根据本公开的媒体回放设备包含能够运行至少一个虚拟机（VM）的中央处理单元 25。在典型实施例中，虚拟机是双虚拟机架构，包括在 CPU26 上运行的低级 VM（例如安全 VM）22 和高级 VM（例如应用 VM）24。在 VM 中运行的程序可以执行并且实施使用规则，以及更新加密算法。运算环境 10 还可以包括应用程序接口（API）40-44，40-44 是用

于允许各种程序互相进行通信的一组例程或协议。

在一个方案中，VM 之一（22 或 24）控制另一个 VM。在另一个方案中，高级和低级虚拟机作为对等虚拟机，按照非等级方式在其之间传递消息。可以以“外部功能调用”来实现这些消息，其中一个虚拟机调用另一个虚拟机中的例程，或者以沿通信信道传递的传统消息来实现。

例如，应用 VM（或高级 VM）24 可以调用安全 VM（或低级 VM）22，以便开始媒体内容 18 的回放（并因此透明解码）。

类似地，安全 VM22 中的代码可以调用应用 VM22，使其了解同步事件或解码问题（例如，安全或许可问题）。

例如，在已经从其原始光学媒体上拷贝媒体的情况下，安全 VM22 通知应用 VM24：需要密钥以便继续播放。响应之，应用 VM24 经由应用级功能 25 显示消息，通知用户他们可以通过用户接口 27“租赁”该电影特定时间长度。如果用户选择这么做，用户必须与工作室服务器进行交易，以便获得包含密钥的“不透明消息”（只由 VM 可理解）。然后，应用 VM24 将包含密钥的消息传递给安全 VM22 以及拷贝保护算法 23，以便认证。

媒体播放器设备还包含处理模块（例如信任处理模块或 TPM）32。TPM 规范是由信任运算组织（TCG）创造的信任运算平台联盟（TCPA）规范的一部分（http://www.trustedcomputinggroup.org）。TPM32 包含解密密钥，并且处理安全密码运算。媒体回放设备还包含 API40、42，允许在虚拟机中运行的任何程序询问设备 IO 硬件和 TPM。这使在 VM 中执行的程序对于使用规则做出智能选择。还提供附加在 CPU26 上的解码模块 34，用于分解解码音频/视频流。

通常，信任平台使实体能够确定该平台中的软件状态或运算环境 10 并且能够将数据密封在该平台的特定软件环境中。实体推断运算环境的状态是否是可接受的并且与平台进行一些交易。如果交易涉及必须存储在该平台上的敏感数据，实体确保按照秘密格式处理该数据，除非该平台中的运算环境的状态对于实体是可接受的。

为了实现该条件，信任平台提供信息，使实体能够推断出信任平台中的软件环境。可靠地测量该信息并且告知实体。同时，信任平台

提供了一种装置，用于加密密码并且描述在解密密钥之前必须进入的软件环境。

“信任测量根基（root）”测量特定平台特征，将测量数据记入测量库，并且将最终结果存储在 TPM（包含用于存储和告知完整测量的信任根基）。因此，TPM 是所有解密密钥的安全存储位置。TPM 还处理大多数密码运算和功能。

此外，媒体回放设备具有安全、受保护的输入和输出 28、与其它播放器 30 联网的能力以及存储设备（例如 RAM36 和 ROM38）。

因此，根据典型实施例，独立虚拟机（VM）运行于包括 CPU 的相同运算环境中。该架构分离两个虚拟机（即高级或应用级 VM 和低级或安全 VM），其中应用和安全虚拟管理器通过标准 API 进行通信。应用虚拟管理器的功能性包括给正在安全虚拟管理器中执行的安全代码提供为了服务，而安全 VM 协调媒体存取和解码功能，以使内容安全对于应用作者是透明的。

对于运算复杂度，安全 VM 对系统资源具有较低的影响，是简单、轻量、低级、并且安全的，并且可以由安全零售商为该 VM 提供适当的软件。应用 VM 具有相对较大的 CPU 以及存储影响，并且负责用户接口和输入/输出功能。

本公开不局限于在例如传统 CD 和 DVD 播放器的媒体播放器中使用，而是可以扩展到在 PC 上运行，或者推广为包括播放可拆卸媒体的功能性的硬件系统。

为了演示和说明，呈现了本公开的优选实施例的上述说明。从阅读下面的附录中，显而易见本公开的其它目的、特点和优点。并不是意欲成为详尽的或者将本公开限制于公开的精确形式。在上述教导下，各种修改和变化是可能的。

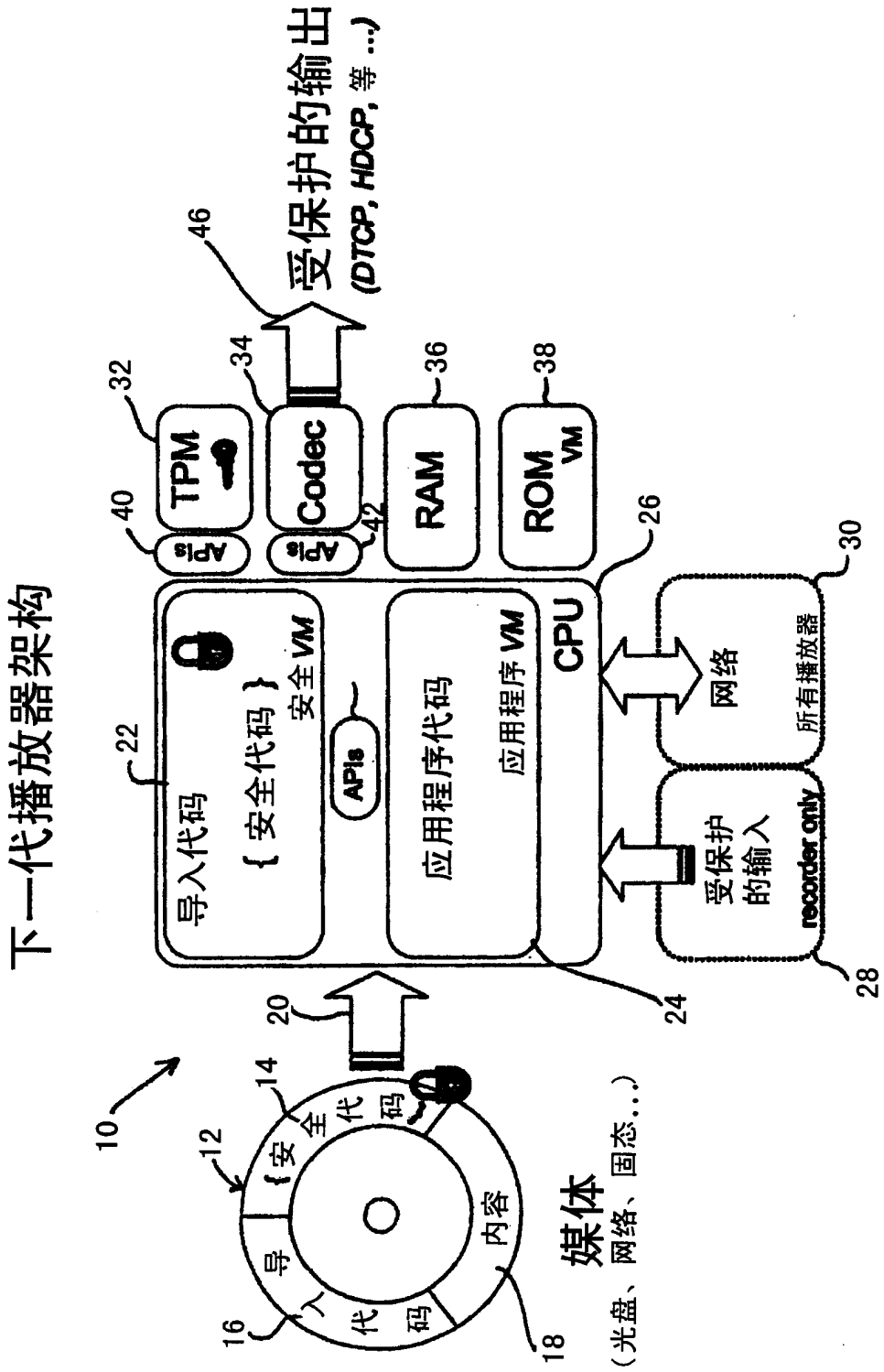

下一代播放器架构
媒体
(光盘、网络、固态...)
导入代码
{安全代码}
安全VM
APIs
应用程序代码
应用程序VM
CPU
受保护的输入
recorder only
网络
所有播放器
TPM
Codec
RAM
ROM
VM
受保护的输出
(DTCP, HDCP, 等...)
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
46

图 1

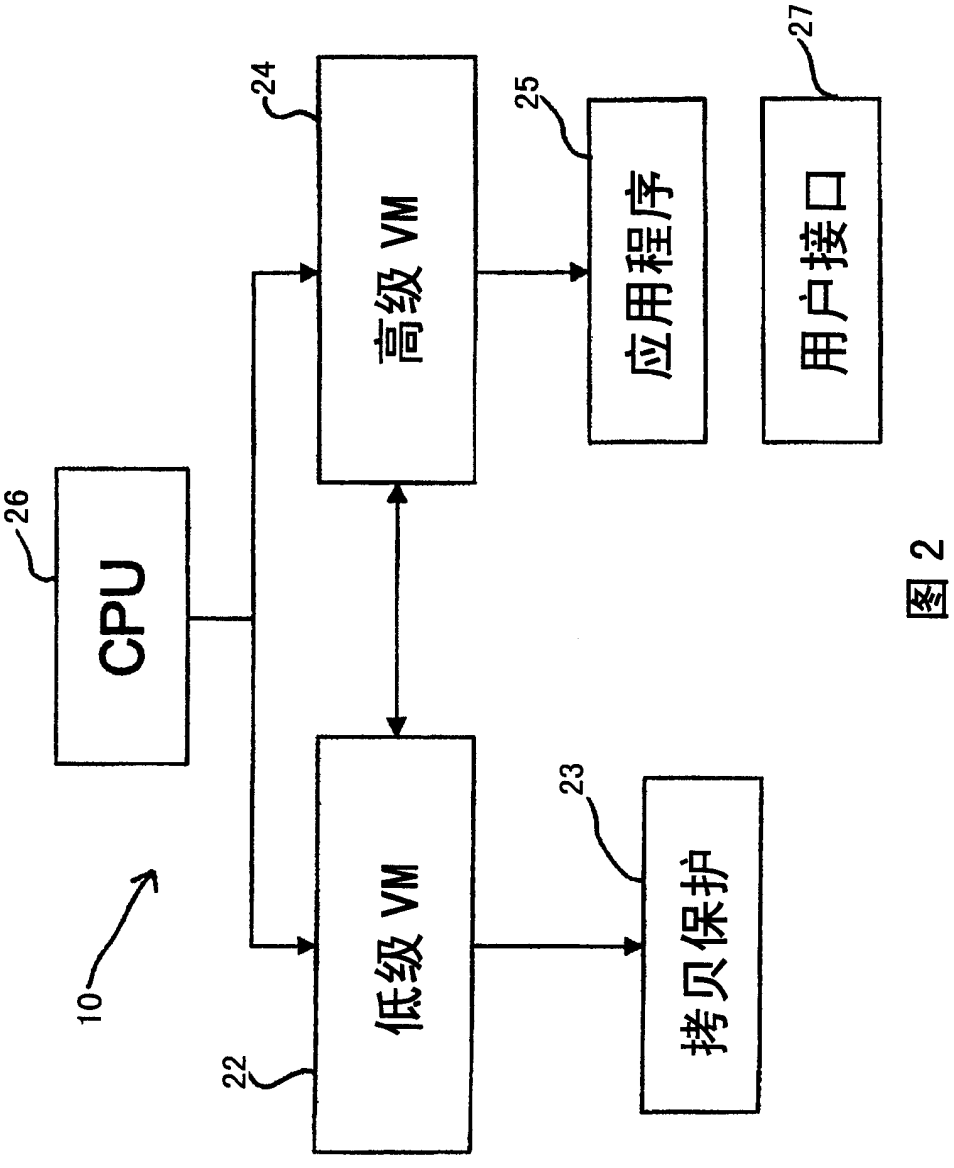


图 2