[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 9/44 (2006.01)

[12] 发明专利申请公开说明书

[21] 申请号 200610106056.0

[43] 公开日 2006 年 12 月 13 日

[11] 公开号 CN 1877525A

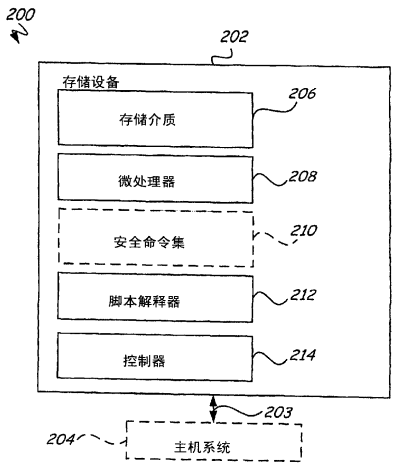
[22] 申请日 2006.5.10
[21] 申请号 200610106056.0
[30] 优先权
[32] 2005.5.10 [33] US [31] 11/125,861
[71] 申请人 西加特技术有限责任公司
地址 美国加利福尼亚州
[72] 发明人 R·H·泰巴德奥
[74] 专利代理机构 上海专利商标事务所有限公司
代理人 钱慰民
权利要求书 3 页 说明书 12 页 附图 7 页

[54] 发明名称

用于在嵌入式系统中安全执行的协议脚本语言

[57] 摘要

用于安全执行脚本的具有加固安全特性的存储设备具有存储介质、接口、安全命令集和脚本解释器。所述接口适合于可通信地将存储设备连接至主机系统以及在主机系统和存储设备之间传送接口命令和信息。所述安全命令集被存储在存储介质上。所述脚本解释器被设置在存储设备内用于根据一个或多个所接收的接口命令处理脚本。所述脚本解释器适合于实施严格序列的脚本执行以及在接口命令违反严格序列启用脚本的情况下终止脚本执行并重新运行与脚本执行有关的任何改变。



1. 一种用于安全执行脚本的具有加固安全特性的存储设备，所述存储设备包括：

存储介质；

接口，适合于可通信地将存储设备连接至主机系统并在主机系统和存储设备之间传送接口命令和信息；

存储在存储介质上的安全命令集；以及

位于存储设备内部的用于根据一个或多个所接收的接口命令处理脚本的脚本解释器，所述脚本解释器适合于实施按严格次序的脚本执行以及在接口命令违反严格次序启用脚本的情况下终止脚本执行并重新运行与脚本执行有关的任何改变。

2. 根据权利要求 1 所述的存储设备，其中所述安全命令集包括：

接口命令，所述接口命令包括从接口接收的命令或者将数据返回到接口的命令，并具有公知的输出和输入。

3. 根据权利要求 1 所述的存储设备，其中所述安全命令集还包括：

内部命令和函数，所述内部命令和函数包括数据处理操作以及不能直接从接口命令调用并且不返回信息到接口的逻辑函数。

4. 根据权利要求 1 所述的存储设备，其中所述脚本解释器包括：

由控制器使用的软件应用程序，用于处理存储设备内的脚本。

5. 根据权利要求 1 所述的存储设备，其中所述脚本解释器适合于在存储设备内实施按严格局部次序的安全命令的执行。

6. 根据权利要求 1 所述的存储设备，还包括：

控制器，位于存储设备内部，适合于控制至存储介质的访问；以及

在存储介质上的安全分区，对安全分区的存取是由控制器严格控制的，其中所述安全命令集被存储在安全分区内并且其中控制器禁止对安全命令集未授权情况下的访问。

7. 一种用于在位于存储子系统内的处理器上处理脚本的方法，所述方法包括：

在存储子系统的存储介质上提供安全命令集，所述安全命令集包括存储子

系统所适合于响应的脚本和接口命令；

利用存储子系统内的脚本解释器检验所接收的命令是否同安全命令集中的至少一个命令相匹配；以及

如果接收的命令同命令集内的任一命令均不匹配，那么利用脚本解释器终止所接收的命令的执行。

8. 根据权利要求 7 所述的方法，其中如果所接收的命令与命令集中的一个命令相匹配，那么所述方法还包括：

从存储介质上的存储单元加载与所接收的命令有关的脚本；以及利用脚本解释器处理所加载的脚本。

9. 根据权利要求 8 所述的方法，其中所述加载步骤还包括：

根据所加载的脚本通过要求一个或多个输入参数的方式对所接收的命令施加期望；

如果所接收的命令的输入参数与一个或多个所需的输入参数不相匹配，那么利用脚本解释器终止脚本的执行；以及

如果所接收的命令的输入参数不相匹配，那么重新运行与脚本执行有关的任何改变至在前脚本执行状态。

10. 根据权利要求 7 所述的方法，还包括:

在事件记录中存储关于失败命令执行的记录。

11. 根据权利要求 8 所述的方法,其中所述处理步骤包括：

根据所加载的脚本执行数据运行和逻辑函数。

12. 根据权利要求 7 所述的方法，其中所述脚本解释器适合于实施由与所接收的命令相关的脚本所定义的严格命令序列。

13. 根据权利要求 12 所述的方法，其中如果所接收的命令同命令集中的一个命令相匹配，那么所述方法还包括：

在检测到对严格命令序列的违反的情况下终止命令的执行或者与命令有关的脚本的执行。

14. 一种设置在存储设备上用于处理的脚本处理系统，所述脚本处理系统包括：

存储介质；

适合于往返于存储介质读写数据的读/写机制；

定义严格的输入命令期望集的允许命令集，所述允许命令集包括接口命令和内部函数；以及

脚本解释器，适合于执行脚本并根据允许命令集实施严格的输入命令期望集，所述脚本解释器适合于在违反输入命令期望的情况下终止脚本的执行。

15. 根据权利要求 14 所述的脚本处理系统，其中所述脚本解释器适合于倒转与终止的脚本的脚本执行有关的任何改变。

16. 根据权利要求 14 所述的脚本处理系统，其中所述脚本解释器包括：

脚本解释器软件应用程序，适合于处理存储设备内的脚本；以及

微处理器，适合于加载脚本解释器软件应用程序以及通过处理脚本的每个命令的方式阻止可允许命令集；

其中如果所接收的命令与可允许命令集中的任一命令均不相匹配，那么脚本解释器使脚本失败。

17. 根据权利要求 14 所述的脚本处理系统，其中所述存储介质包括数据区域以及一个或多个安全分区，其中对安全分区的存取是由存储设备内的控制器严格控制的，其中允许命令集被存储在一个或多个安全分区中的至少一个安全分区内的存储介质上。

18. 根据权利要求 14 所述的脚本处理系统，其中允许权利集还包括：

仅在脚本环境内执行的用于启动脚本执行内的逻辑操作的命令的子集。

19. 根据权利要求 18 所述的脚本处理系统，其中所述命令的子集包括用于根据两个或更多同时的输入做出判断的条件转移命令。

20. 根据权利要求 14 所述的脚本处理系统，其中允许命令集包括具有公知的输入和输出并用于执行公知确定的操作的命令。

用于在嵌入式系统中安全执行的协议脚本语言

相关申请的交叉引用

无。

技术领域

本发明涉及在存储设备上执行的脚本，具体而言，涉及一种用于在嵌入到存储设备的微处理器中控制脚本或者程序设计语言的执行的方法。

背景技术

包括闪速存储器设备和可移去存储器设备在内的磁盘驱动器及其他存储子系统典型地在其嵌入式电子仪器中具有处理器（有时称作微处理器）。处理器可用于隐藏计算结果以及存储的数据。在许多安全申请中，以及在许多申请提供者会希望防止申请或者内容的窃取使用的申请中，也许需要以隐藏的方式执行某些计算。这包括隐藏密钥，所述密钥可以被采用以经由密码证据或者经由对共享秘密的简单认识来开启资产。以隐藏方式执行计算也可以包括隐藏某些计算，其可以被采用以允许在存储设备以外的软件或者内容可以以希望的方式运行。不同于常规的读/写命令，安全命令典型地被提供在协议环境内，所述协议规定在存储设备和主机之间必须根据严格定义的局部规则约束来交换特殊序列的消息。数字权利管理及其他内容保护协议具有该形式。读/写命令本身是一个整体，而只有当严格序列完成且没有错误或者妨碍的情况下协议才是一个整体。

一种方便的和通用的用于执行这种隐藏的且定义明确的运算和计算的方式是通过在存储设备处理器上执行的脚本或者编程语言。允许外力程序设计存储设备处理器的潜在缺点是所述外力能因此存心不良地使用该程序设计能力。或者出于恶意，或者出于意外，外力可以削弱存储设备的运行而产生出乎意外的后果。本发明的实施例提供了这些及其他问题的解决方案，并且提供了相对于现有技术的其他优点。

发明内容

一种用于脚本的安全执行的具有加固安全特性的存储设备具有存储介质、接口、安全命令集以及脚本解释器。所述接口适合于可通信地将所述存储设备

与一主机系统相连接以及在主机系统和存储设备之间传送接口命令和信息。在存储介质上存储所述安全命令集。所述脚本解释器被设置在存储设备内用于根据一个或多个所接收的接口命令处理脚本。所述脚本解释器适合于实施按严格次序的脚本执行并且在接口命令违反严格次序启用脚本的情况下终止脚本执行并重新运行(roll back)与脚本执行有关的任何改变。

在一个实施方式中，描述了一种用于在设置在存储子系统内的处理器上处理脚本的方法。根据经由将存储子系统连接至主机系统的接口接收的命令处理每个脚本。存储子系统所响应的安全命令集被存储在存储子系统内的存储介质上。存储子系统内的控制器检验每个所接收的命令是否同安全命令集内的至少一个命令相匹配。如果所接收的命令同命令集中的任一命令均不匹配，那么终止所接收的命令的执行。

在另一个实施方式中，一种脚本处理系统被设置在存储设备上。所述存储设备具有存储介质和适合于往返于存储介质读和写数据的电路，并且所述存储设备经由接口连接主机系统。允许命令集定义严格的输入命令期望集。允许命令集包括接口命令和内部函数。脚本解释器适合于执行脚本并根据允许命令集实施严格的输入命令期望集。脚本解释器适合于在违反输入命令期望的情况下终止脚本的执行。

通过读取下列详细说明并参照对相关附图的评述，表征本发明的实施方式的其他特征和优势将变得显而易见。

附图说明

图 1 是磁盘驱动器的立体图。

图 2 是根据本发明的实施方式实现的系统的简化方框图。

图 3 是根据本发明的实施方式的存储子系统的简化方框图。

图 4 是根据本发明的实施方式的用于处理在位于主系统和存储设备之间的接口上方接收的命令的方法的简化流程。

图 5 是根据本发明的实施方式的用于在存储设备上处理由接收的命令启用的脚本的方法的简化流程。

图 6 是根据本发明的实施方式实现的系统的简化方框图。

图 7 是根据本发明的实施方式的用于利用存储子系统的脚本解释器处理脚本的方法的简化流程。

具体实施方式

图 1 是其中本发明的实施方式是有益的磁盘驱动器 100 的立体图。磁盘驱动器 100 包括具有基座 102 和顶盖（未显示）的外壳。磁盘驱动器 100 还包括磁盘组 106，其通过磁盘夹 108 安装在主轴电动机（未显示）上。磁盘组 106 包括多个专用磁盘，这些专用磁盘是以围绕中心轴 109 同轴旋转的方式安装的。每个磁盘表面具有相关的磁盘磁头浮动块 110，其被安装到磁盘驱动器 100 用于同磁盘表面进行通信。

在图 1 所示的实例中，浮动块 110 由支架（suspensions）112 支持，所述支架 112 又附着于激励器 116 的磁道存取臂 114 上。图 1 所示的激励器具有通常所说的旋转电液转换器的类型并包括直线电机（VCM），在图中通常以 118 来标示。直线电机 118 利用它的附装磁头 110 围绕枢轴 120 旋转激励器 116，以便沿在磁盘内径 124 和磁盘外径 126 之间的拱式路径 122 在期望的数据磁道上方定位位于浮动块 110 上的磁头。直线电机 118 由伺服电子仪器 130 根据由位于滑动块 110 和接口 103 上方的主机系统 101 的磁头生成的信号驱动。

对本领域的技术人员而言应该理解的是：图 1 所示的实施方式是在其上可以实现本发明的一种可能的存储系统。用于控制驱动器侧脚本执行的本发明也可以在任何具有微处理器的存储子系统上实现。在此使用的术语“脚本”指的是命令和有关信息的有序集，其可以改变数据，存取外围设备，或者执行功能。驱动器侧或设备侧脚本是利用存储设备的嵌入式的电子仪器执行的脚本。

通常，本发明提供了一种手段，所述手段允许程序员经由存储设备中的微处理器编写由存储设备处理的脚本，同时提供对大型病毒的保护措施。本发明利用在存储设备内的脚本解释器。脚本解释器被配置成或者适合于接受经接口接收的预定命令。假定这种命令是公知的并且这种命令是完全受约束的，那么可以假设所述命令是安全的。在此使用的术语“命令”指的是由存储设备经由接口接收的“等到结束”、ATA 或者 SCSI 类型的命令，其中所述接口是本领域公知的。

对脚本而言有效的命令仅仅是经由接口的调用有效的。唯一的接受命令是跨越接口处理的函数调用，以及任何其他类型的命令被脚本解释器拒绝。特征函数根本不能被调用除非其在存储设备上的“安全”命令中已经预先被定义。此外，特征函数不能以出乎意外的次序被调用，或者脚本解释器将拒绝函数调用

并重新运行起源于脚本执行的任何改变至它的初始状态。

脚本（命令和信息的有序序列）定义严格的部分有序集。如果曾经经由接口传送违反严格部分有序的命令，那么整个脚本被终止并且系统重新运行与脚本有关的任何改变以将存储设备返回至它的以前的状态。换句话说，系统重新运行改变以便就好像脚本从未被启动一样。

在严格部分有序的外部命令以外，允许命令也可以包括与条件检验有关的命令，以便存储设备可以根据它自己的内部检验生成条件分支。这些检验包括像代替外部命令的输出，检验以查看输出是否等于或者大于某些值，那么根据那个输出生成条件分支至部分有序之类的东西。用这样的方式，脚本可以被利用以便完全内部地回答例如"是/否"类型的问题，并按结果进行转移。

图 2 是根据本发明的实施方式实现的系统 200 的简化方框图。系统 200 包括存储设备 202 和主机系统 204，其通过接口 203 可通信地相互连接。存储系统 202 可以是磁盘驱动器、可移动或者固定存储器子系统、闪速存储器磁盘、MRAM、EEPROM 或者任何其他的存储系统。主机系统 204 可以是具有操作系统和用于经由接口 203 同存储系统 202 连接的 SCSI 或者 ATA 总线的计算机。

通常，存储设备 202 包括存储介质 206、微处理器 208、命令集 210、脚本解释器 212（运行时间或者命令集执行规则）和控制器 214。通常，微处理器 208 适合于执行计算和经由脚本解释器 212（或者运行时间）处理脚本和程序，与由主机系统 204 执行的任何处理无关。命令集 210 是之前定义的命令，从所述命令具有公知的输出而且其以公知被确定的方式来处理这个意义上而言所述命令的特征是安全的。命令集 210 可以被存储在控制器 214 的固件中、存储在存储介质 206 上、或者存储在任何其他的为控制器 214 能达到的并且没有越权存取危险的存储单元中。控制器 214 管理在存储介质 206 和外部空间（诸如来自主机系统 204）之间的通信。

应当理解，存储设备 202 也可以作为独立的存储设备。主机 204 以虚线表示用于指示在某些环境中主机 204 不是必需的。例如，如果存储设备 202 是独立的网络设备，那么存储设备 202 可以经由标准接头（诸如双绞线以太网连接、共轴网络连接、拨号网络连接等等）直接连接至网络。如果存储设备 202 是独立的网络设备，那么存储设备 202 可以包括用于认证用户和设备的认证算法以控制对存储在存储设备 202 上的数据的访问。

通常，命令集 210 表示存储设备 202 所响应的容许或者允许命令。此外，命令集 210 可以包括定义脚本解释器 212 可以执行经由接口 203 接收的指令所采用的次序的可允许命令次序。程序员可以编写脚本用于以严格控制的方式经由接口 203 存取命令集 210。

在存储设备 202 的微处理器 208 上用于处理的脚本主要是由允许命令集 210 组成的。脚本中的每个命令可以作为期待输入，作为内部执行的指令，或者作为用于提供输出的装置。命令可以被设计成提供输入和输出，仅仅提供输入，或者仅仅提供输出。

通常，从每一安全命令具有公知的输出并以公知确定的方式执行这个意义上而言，存储设备所响应的安全命令集的特征在于其安全性。脚本主要是由下述三种角色的安全命令组成：1)作为期待的输入，2)作为内部执行的指令，或者 3)作为输出生成命令。脚本本身标识期待的输入命令。在嵌入式平台上的脚本的执行要求脚本内的命令以下述的次序出现：所述命令可以被脚本所代替并且没有任何插入的命令。如果由脚本支配的按照严格顺序的命令是违反，那么整个脚本被终止，并且由脚本所代替的动作被“重新运行”。用这种方式，脚本可以被看作根据多个输入执行单一的计算。在优选实施方式种，每个命令表示式返回一可分配的值。

除了设备通过外部指令所响应的命令之外，还存在仅在脚本环境中执行的命令集。一种该命令是条件 IF-THEN-ELSEIF-..-ENDIF 命令。其他命令或者运算符包括不等式、等式、逻辑和算术命令或者运算符。诸如类型之类的其他程序设计语言抽象概念被设置为“安全”。脚本不允许直接存储器操作。另外，循环和递归具有作为默认设置的上限。

应当理解的是，允许外营力编程存储设备处理器的缺点是外营力能因此使用程序设计能力从而存心不良或者意外地削弱运行并可能产生出乎意外的结果。本发明介绍了一种用于将这种编程限制为允许命令的严格可控程序，被在脚本环境内执行所述命令。如果由脚本启用的命令被越出期待次序范围外或者以不同于由脚本所支配的序列被执行，那么处理器 208 重新运行脚本所代替的全部动作。假如这种编程符合允许命令的严格可控序列，在某种程度上，因为命令执行是根据以受控制的次序执行的允许命令集，因此，由外营力对存储设备处理器的编程是被允许并安全的。在某种程度上，因为脚本执行是确定的，

所以通过这种方法，可以保证脚本的安全执行。

在一个实施方式中，脚本可以被概念化为受管理的密码，其是利用语言编译程序被发展为把运行时间作为目标的代码。管理代码典型地提供"元数据"以便允许运行时间来设置在组合模块中代码化的方法，从而存储并检索保密信息，处理异常等等。用于管理代码的执行的管理代码以运行时间为目标，提供诸如存储器管理、线索管理等等之类的核心服务。在替换实施方式中，微处理器208利用执行规则集或者脚本解释器212执行脚本语言。为了处理管理代码块起见，脚本解释器212也可以是运行时间。

如上所述，本发明的方法考虑了条件命令，其仅在脚本环境内被执行。这种条件命令提供了用于根据通过存储设备内的脚本执行的检验结果进行分支的装置。这种检验可以出于各种目的搜索并且检索状态或者其他信息的类型并根据检索信息产生输出而不显示该信息。其一个实例可以是在医学装置内的数据库。在医学环境中，数据库可以包括有关病人的出生日期、社会保险号及其他保密信息的介绍。通常需要保护这种信息避免越权存取和检验，并且防止这种信息以通过接口执行命令的方式被暴露。允许接口命令集可以存取局部函数及其他命令以检验某些类型的信息是否是真实的，但是其不显示任何数据给用户。例如，护士可以检验病人记录是否存在。护士可以促使经由接口发送命令以检验病人的记录。命令存取局部函数，所述局部函数允许系统内部地检验记录。取决于该调用的函数，在内部检验以后，系统可以向护士展示用于编辑与病人概貌相关的信息的页而不暴露保密资料给该护士。

本发明的一个主要优点是复数运算完全可以在存储设备内执行。例如，涉及加密／解密对称密钥或者公共专用密钥对的操作可以查询转发区，其中在所述转发区中提供密钥至授权的设备。如果密钥是未加密的，那么这可能潜在地暴露密钥至未被授权的计算机窃贼，如果计算机窃贼正在监视事务处理。利用本发明经由网络发出的用于获取密钥的命令（诸如"Get Key"命令）被脚本解释器捕获，并且代替返回未加密的密钥至用户，所述密钥可以被设备的内部处理器在存储设备内检索并加密，而不会暴露密钥。密钥或者密钥对可以被经由接口返回到请求设备中，例如，包封PKES密钥的RSA。以这种方法，存储设备可以被制造得利用智能卡来运行（其需要密钥），而无需暴露密钥该"特洛伊木马"或者特工人员。

另外，应当理解的是：条件命令允许期望两个或更多输入命令同时发生。在这一情况下，任何一个命令可以继续而无需另一个继续。虽然如此，还有脚本解释器 212 将接受的命令的严格部分有序。脚本本身可以被吸入到存储介质 206 上的存储位置（或者任何其他的存储位置），并且可以通过启用处于该存储单元中的内容的方式被启用。脚本的输出可以包括各种各样的设备输出，并且可以包括对存储在存储介质 206 上的数据的转换。

图 3 是根据本发明的实施方式的存储子系统 300 的简化方框图。对本领域的技术人员而言应当理解的是：为了简化论述起见省略了大量细节，诸如读/写机制之类的细节，所述读/写机制包括马达、锭子、滑动块等等。该子系统 300 包括接口 302、控制器 304、微处理器 306、安全（或者允许）命令集 308 以及存储介质 310。在存储介质 310 上提供脚本解释器应用程序 312，并且根据需要由微处理器 306 加载以处理存储设备侧脚本。该接口 302 可以是用于直接同主机系统通信的主机接口。做为选择，接口 302 可以是适合于同网络连接并且通过该接口连接接收命令以及期望数据的网卡。

如果命令包含在安全命令集中，那么当经由接口 302 接收命令时，控制器 304 存取与命令有关的存储单元。命令被通过存取存储介质 310 的存储单元的方式被启用，其中在所述存储单元中存储用于执行命令的脚本。微处理器 306 启用脚本解释器 312（例如加载脚本解释器 312 到存储器里）。微处理器 306 运用脚本解释器处理脚本。第一脚本解释器 312 将每个命令同安全（允许）命令集 308 中的命令进行比较。然后，脚本解释器 312 处理脚本。如果命令不是安全命令集 308 中的命令，那么不包括期待的信息，或者如果命令以违反期待次序的方式被接收，那么脚本执行被终止并且在脚本执行期间生成的改变被重新运行至之前的命令状态。

通常，脚本可以包含被期待跨越接口的命令（“接口命令”）以及其他的命令（所述命令和在此所称的“局部函数”或者“内部函数”有关）。通常，“局部函数”或者“内部函数”可以包括数据处理命令或者操作以及不能直接经由接口存取的而且不能返回数据至接口的逻辑函数。例如，微处理器 306 根据经由接口 302 接收的命令将脚本加载到脚本解释器 312 里。该脚本由一系列按照次序执行的函数（Fnl,Fn2, Fn3,..., FnN）组成。每个函数可以包括经由接口接收的期待的接口命令，但是一些函数可以包括局部函数以及同时包括命令。局部命令优选地

包含于安全命令集308中。如果Fnl表示函数或者逻辑操作，诸如x=l+b，其中b是接口命令，那么设置期望以使得b跨越接口302。如果b是局部函数或者局部变量，那么不期望设置b。重要的是，因此，理解了在接口命令和脚本运行以及局部函数之间的差异。

通常应当理解的是：存储在存储器中的脚本不必非要由控制器进行检验直到试图脚本的执行为止。因此，可以经由接口302加载脚本至存储介质310上。控制器304根据经由接口302接收的命令干预脚本的执行。因此，当刚一在存储介质310上存取脚本的存储单元，控制器304就干预以确信在允许进行脚本执行以前执行的函数和命令是安全的。如果通过将其同之前确定的安全命令集308进行比较后确定脚本函数或者命令不是安全的，那么脚本执行被停止并且与执行有关的任何改变被撤消（重新运行至之前的脚本执行状态，所述状态是临脚本被存取时数据和/或存储设备300的设置的状态）。

图4是根据本发明的实施方式的用于控制经由在主机系统和存储设备之间的接口接收的命令的执行的处理的简化流程图。经由接口从设备接收命令（步骤400）。存储设备的控制器将所接收的命令同包含在安全命令集中的命令进行比较（步骤402），其中所述安全命令集被存储在存储设备中。如果所接收的命令与安全命令集中的命令不匹配（步骤404），那么脚本执行被终止（步骤406），由脚本执行所引起的任何改变被返回或者重新运行至初始状态（步骤408），并且关于事件的记录被存储在安全分区（partition）（SP）记录中（步骤410）。

相反，如果所接收的命令同安全命令集中的命令匹配（步骤404），那么命令被检查期待的参数（步骤412）。例如，如果所接收的命令启用用于察看"Bob是否21岁"的处理，那么命令的期待参数典型地将包括人名和当前的日期。如果所接收的命令参数缺乏或者与期待的参数不相匹配（步骤414），那么脚本执行被终止（步骤406），由脚本执行所引起的任何改变被返回或者重新运行至初始状态（步骤408），并且事件的记录被存储在安全分区（SP）记录中（步骤410）。如果参数同期待的参数匹配（步骤414），那么执行命令（步骤416）。

典型地，由存储设备内的微处理器通过存取与命令有关的存储单元的方式执行命令。与命令有关的存储单元可以包含脚本，然后由微处理器结合脚本解释器处理所述脚本（如图5所示）。

应当理解的是：对经过接口接收的每个命令重复上述描述的处理，并且在

脚本内对每个接口命令而言都遇到上述处理。接口命令是经由接口接收的命令或者直接从存储设备将信息经由接口发送的命令。通过在执行以前检验每个命令，控制器检验每个命令是否是安全命令，并且命令执行可以被严格限于之前确定列表的安全命令。通过重新运行来自失败脚本执行的全部改变，系统确保仅仅紧随命令式语言程序步骤的安全脚本被执行。

图 5 显示了根据本发明的实施方式的处理脚本的方法的简化流程图。在该特殊的实施方式中，假定经由在主机系统和存储设备之间的接口已经接收命令，并且所述命令被包容在安全命令集中。另外，因该流程图起见，脚本解释器被认为已经被存储设备的微处理器加载，并且以下的处理被理解在存储设备内完全发生。

启用安全接口命令的命令输入被接收（块 500）。在这种情况下，特殊的命令是用于核对某个人的年龄以检验他或她是否是 21 岁的命令。在伪码种，命令是“CheckAge (Name,Today)”，其包括名字参数和当今日期。

脚本解释器开始存储在与所接收的命令“CheckAge()”有关的存储单元中的脚本的执行（步骤 502）。脚本解释器根据姓名参数选择关于人名字的文件（步骤 504）。脚本解释器对所选择的文件执行更新二元运算（步骤 506）。脚本解释器然后读取关于所选择文件的日期内容（例如生日）的二进制（步骤 508）。脚本解释器对所述日期执行更新二元运算（步骤 510）。

脚本解释器然后处理条件函数，所述条件函数建立两种可能的期望：或者该人 21 岁或者 21 岁以上，或者该人 21 岁以下。如果该日期大于或等于今天减 21 年（步骤 512），那么返回"是"至该接口（步骤 514）。如果该日期少于 21 年份，然后返回“不”至该接口（步骤 516）。

在该实例中，如果“CheckAge”命令被经由接口发送而不带有期待的“Name”和“Today”（今天的日期）参数，那么刚一发现未提供该期待的参数就终止命令的执行（图 4 中的步骤 412）。应当理解的是：脚本可以包含用于返回数据至接口或者执行来自安全命令集的接口命令的函数；然而，这种函数必须以满足该参数期望的次序和以期待的次序执行。另外，脚本执行失败并且改变被重新运行至之前的执行状态。

图 6 举例说明了根据本发明的实施方式实现的系统 600。系统 600 包括主机系统 602，所述主机系统 602 可以是个人计算机或者其他的处理设备。系统

600 还包括可通信地经由接口 603 同主机系统 602 相连接的存储设备 604。存储设备包括控制器 605 和存储介质 610。控制器 605 包括微处理器 606 和固件 608。提供脚本解释器 607，其可以是脚本被写入其中的运行时间应用程序，微处理器 606 实施脚本的步骤或者规则集，或者可以是适合于处理脚本的电路。存储介质 610 具有包含脚本的存储单元 612。存储介质 610 还可以被分成数据区域 614 和一个或多个安全分区 616。在一个实施方式中，安全或者允许命令集（C）被存储在控制器 605 的固件 608 中。在一个替换实施方式中，安全或者允许命令集（C）被存储在存储介质 610 的安全分区 616 中。

通常，安全分区 616 是存储介质 610 上对其的访问被严格控制的区域。在一个实施方式中，仅仅存储设备 604 的控制器 605（或者微处理器 606 或者诸如结合控制器 605 运行的脚本解释器 607 之类的应用程序）可以存取安全分区 616。在安全分区 616 内的权限表中所标识（唯一地被标识或者标识为合理的角色）的被授权用户（或设备）可以被允许经由控制器 605 对存储在分区 616 内的数据的受限存取。以这种方法，安全或者允许命令集（C）可以被保存在存储介质 610 中而不允许对命令集的改变。

在这些实施方式中，存储在存储介质 610 的数据区域 614 中存储单元 612 内的脚本经由命令或者试图存取存储单元 612 的方式通过接口由外部设备来执行。微处理器 606 分析脚本并存取安全分区 616（或者固件 608）中的命令集（C）以便在执行每个命令以前确认每个命令。

脚本解释器 607 可以被存储在固件 608 中或者可以被存储在存储介质 610 的数据区域 614 中，并且根据需要由微处理器 606 加载。脚本解释器 607 通常结合固件 608 和微处理器 606 一同运行，类似于软件应用程序结合随机存取存储器（RAM）和计算机系统内的微处理器一同运行的方式。

如预先指示地那样，命令可以包括条件命令，所述条件命令可以允许两个或更多输入命令同时被预期执行，其中或者命令可以在不需要其他必要地继承的情况下（诸如 IF-THEN-ELSEIF-..ENDIF 命令）继续。虽然如此但仍存在脚本解释器将处理的严格部分有序的命令。

通常，脚本本身应该被写入到存储设备 604 的存储介质 614 上的存储位置中，并且脚本应该通过启用在该位置上的存储器的内容的方式被启用。此外，脚本执行的输出可以包括在存储设备内的存储数据的改变，如果在脚本解释器

结束脚本以前脚本被终止那么当脚本刚一终止就将返回（撤消或者重新运行）所述改变。做为选择，脚本的输出可以经由接口 603 被写入到一个或多个外围设备 618 中。

图 7 是根据本发明的实施方式的用于管理在存储设备上执行脚本的方法的简化流程图。为了在图 7 中显示所述流程图，假定存储设备已经接收对存储在存储设备的存储介质上的存储单元中的内容的输入请求存取，其包含用于执行的脚本。控制器检索存储单元的内容（脚本）并提供内容至微处理器，所述微处理器利用脚本解释器处理所检索的脚本。

脚本解释器根据所接收的命令对脚本执行第一函数（步骤 700）。第一函数可以利用与所接收的接口命令有关的一个或多个参数。假定第一函数是来自安全命令集的内部函数或者安全命令，那么脚本解释器执行从脚本选择的函数或者命令（步骤 702）。如果函数或者命令的所期望的参数不存在（步骤 704），那么脚本解释器终止脚本的执行（步骤 706），重新运行基于脚本的任何改变（步骤 708），并在安全日志中记录失败的脚本事件（步骤 710）。如果函数或者命令的期望的参数存在（步骤 704），那么脚本解释器处理脚本函数至完成（步骤 712）。

如果脚本包括更多函数，那么脚本解释器选择脚本内的下一个函数（步骤 714）。如果下一个选择的脚本函数不满足系统对严格次序的要求，那么终止脚本执行（步骤 706）。重新运行或者撤消与脚本有关的改变（步骤 708），并且失败的脚本执行事件被记录到事件记录中（步骤 710）。如果满足严格的次序（步骤 716），那么重复步骤 702 和序列直到脚本的全部函数都已经被处理为止或者直到脚本失败为止。

应当理解的是：即使连同本发明的各种实施方式的结构和功能的细节在上述的说明中已经阐述了本发明的各种实施方式的许多特征和优点，但是这些公开的内容仅仅是说明性的，并且可以在细节上作出改变，特别是在本发明的原则内对与部件的结构和排列有关的细节作出改变，本发明的原则是由所附权利要求所表示的术语的宽泛广义用意所指示的最大可能范围。例如，在不脱离本发明的范围和精神的情况下，取决于用于脚本处理存储系统的特定应用程序而可以使得所述特殊元件变化，同时基本保持相同的功能。此外，虽然在此描述的优选实施方式给出了用于管理在诸如磁盘驱动器之类的存储子系统的微处理器上执行的脚本的方法或者系统，但是本领域的普通技术人员应当理解的是：

在不脱离本发明的范围和精神的情况下本发明的教导可以被用于控制在任何类型的存储设备上执行的脚本。

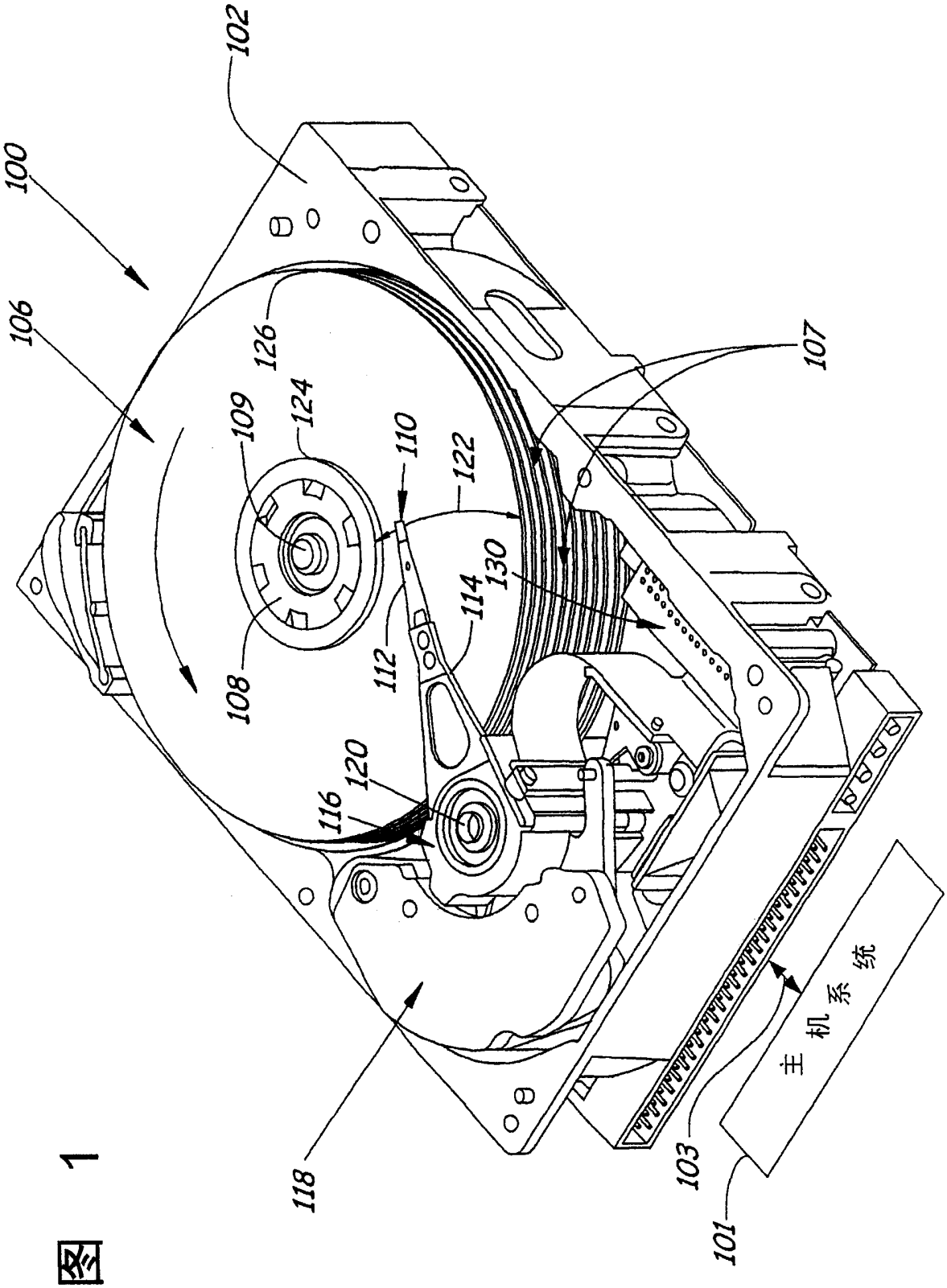

102
100
106
126
107
109
124
110
122
130
114
108
112
120
116
118
103
101
主机系统

图 1

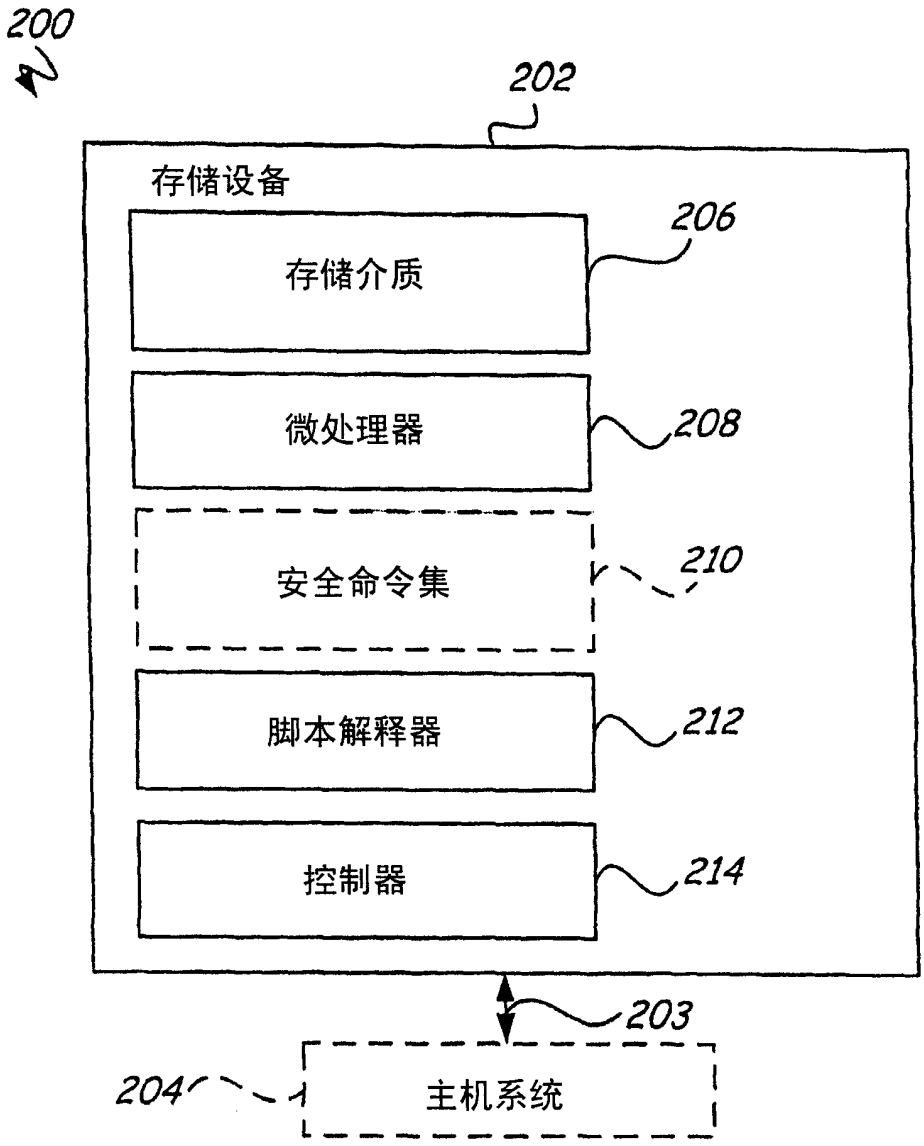

200
202
存储设备
206
存储介质
208
微处理器
210
安全命令集
212
脚本解释器
214
控制器
203
204
主机系统

图 2

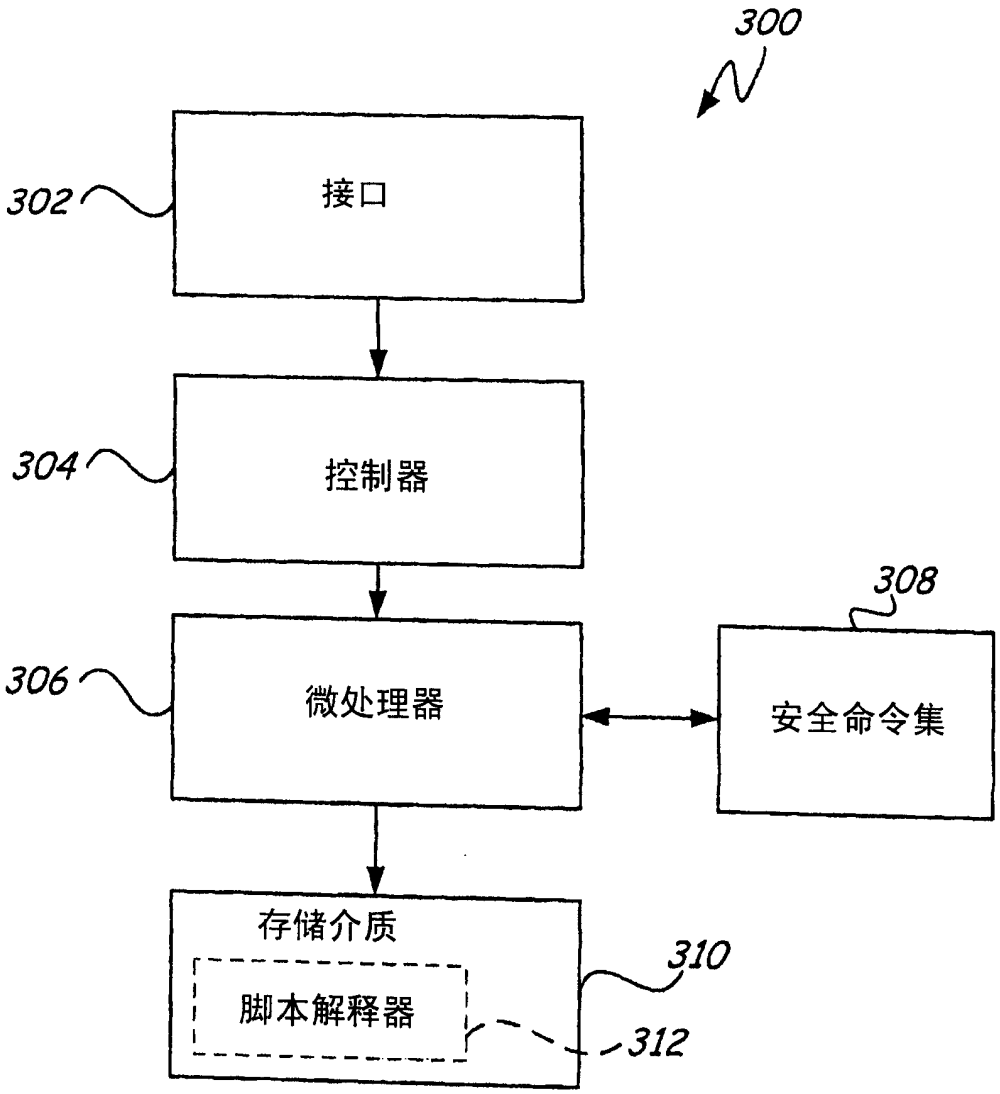


图 3

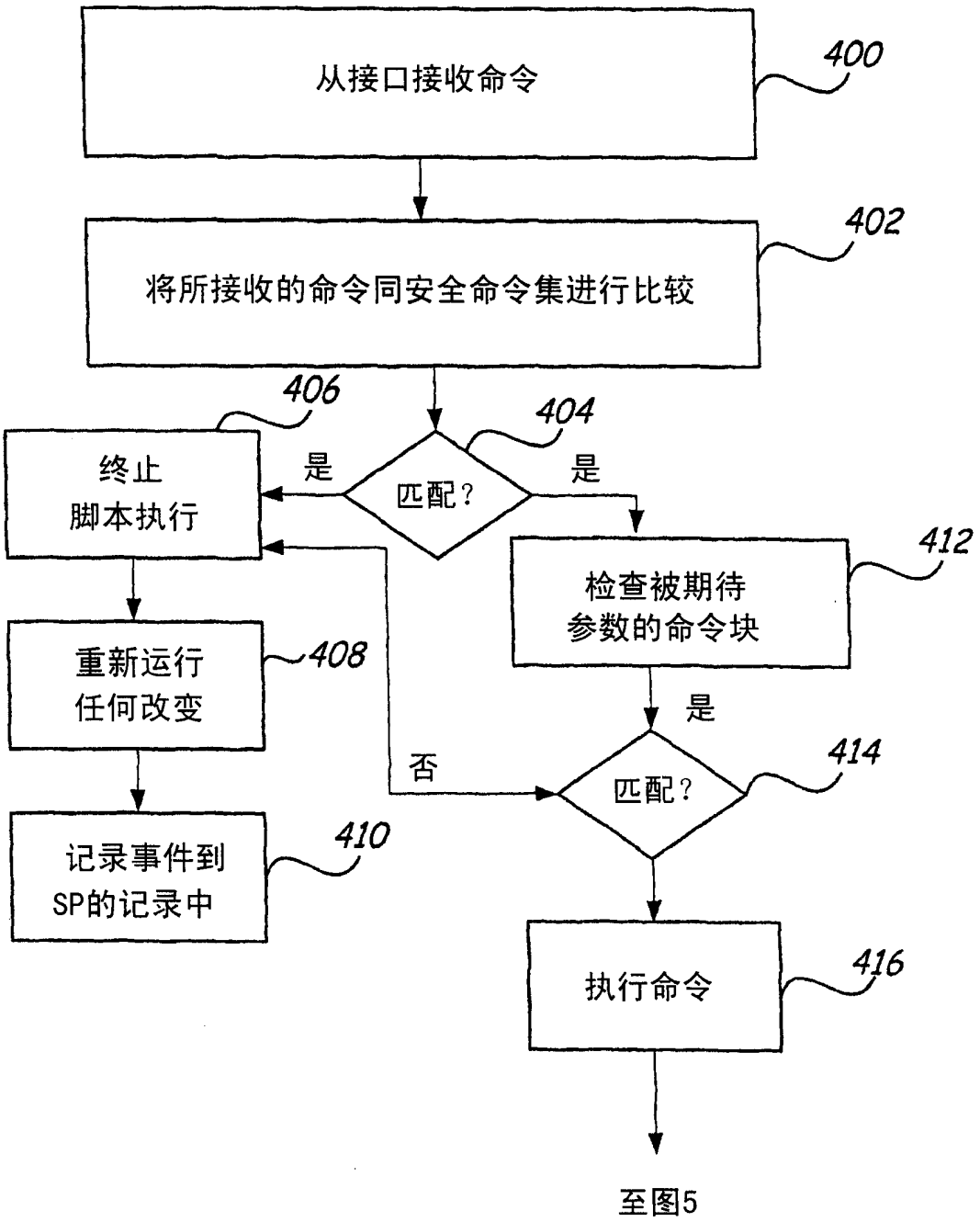

从接口接收命令
400
将所接收的命令同安全命令集进行比较
402
406
404
终止
脚本执行
是
匹配？
是
检查被期待
参数的命令块
412
重新运行
任何改变
408
是
否
匹配？
414
记录事件到
SP的记录中
410
执行命令
416
至图5

图 4

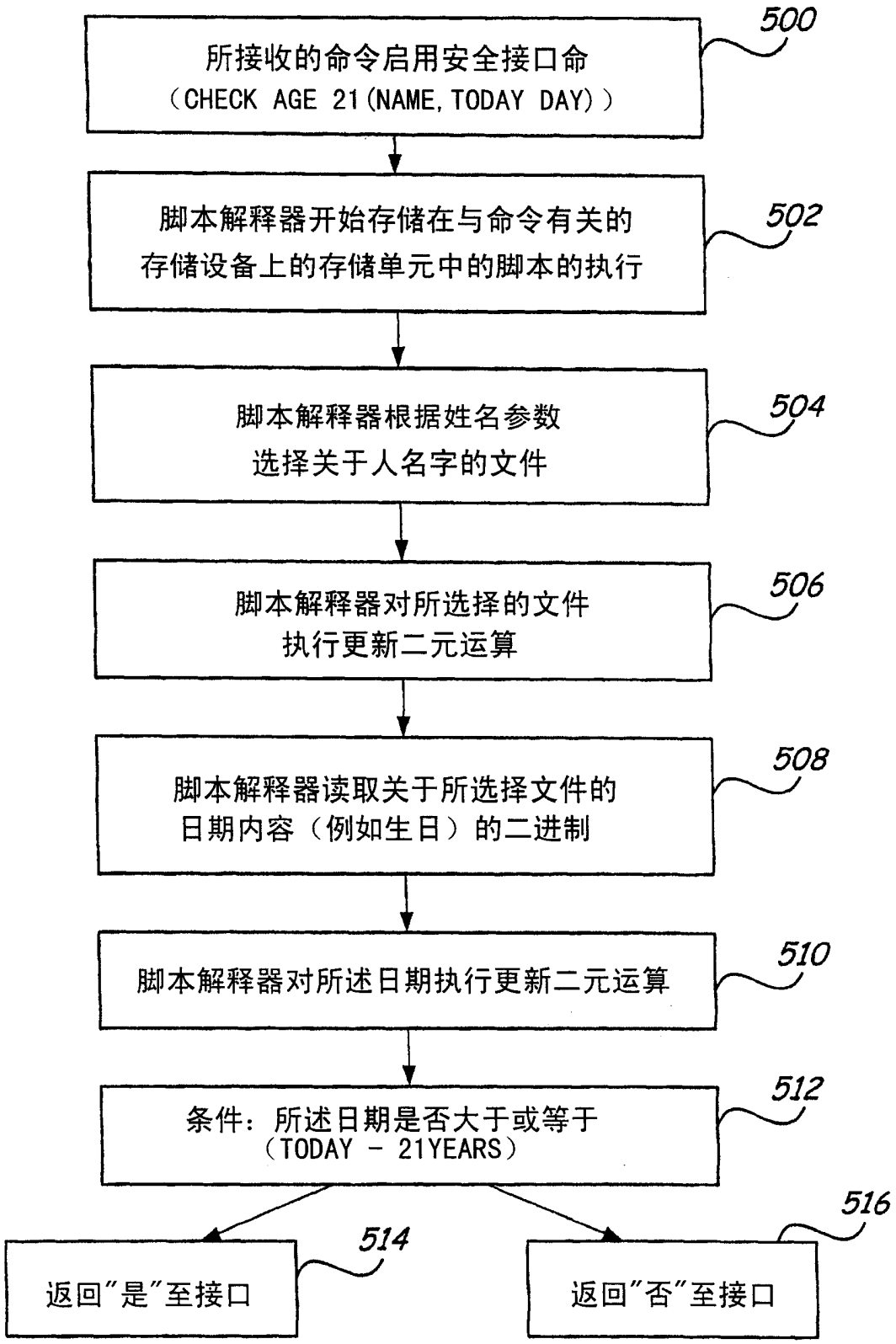

500
所接收的命令启用安全接口命
（CHECK AGE 21(NAME, TODAY DAY)）
502
脚本解释器开始存储在与命令有关的
存储设备上的存储单元中的脚本的执行
504
脚本解释器根据姓名参数
选择关于人名字的文件
506
脚本解释器对所选择的文件
执行更新二元运算
508
脚本解释器读取关于所选择文件的
日期内容（例如生日）的二进制
510
脚本解释器对所述日期执行更新二元运算
512
条件：所述日期是否大于或等于
（TODAY - 21YEARS）
514
返回"是"至接口
516
返回"否"至接口

图 5

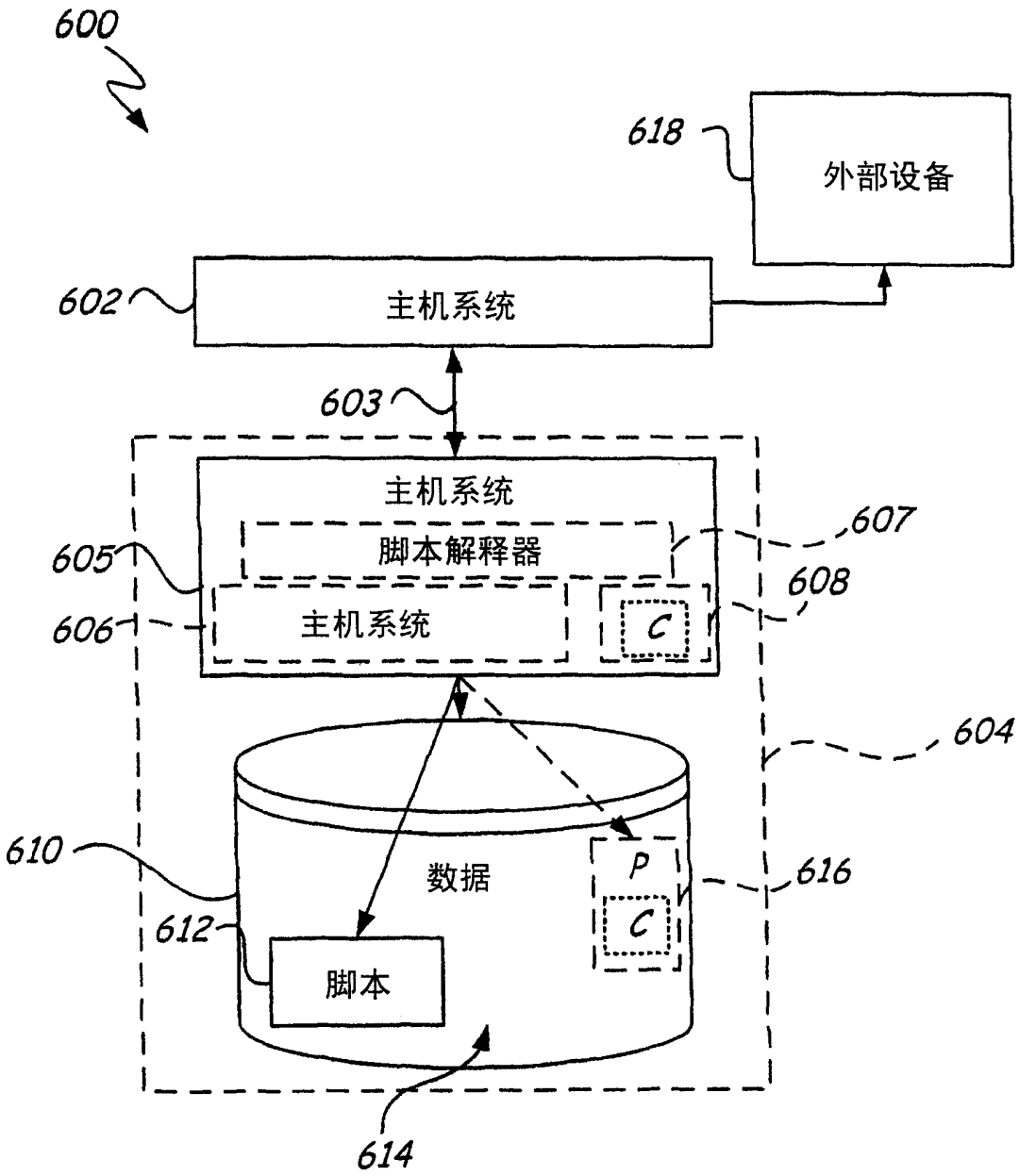

600
618
外部设备
602
主机系统
603
主机系统
脚本解释器
607
605
608
606
主机系统
C
604
610
数据
P
C
616
612
脚本
614

图 6

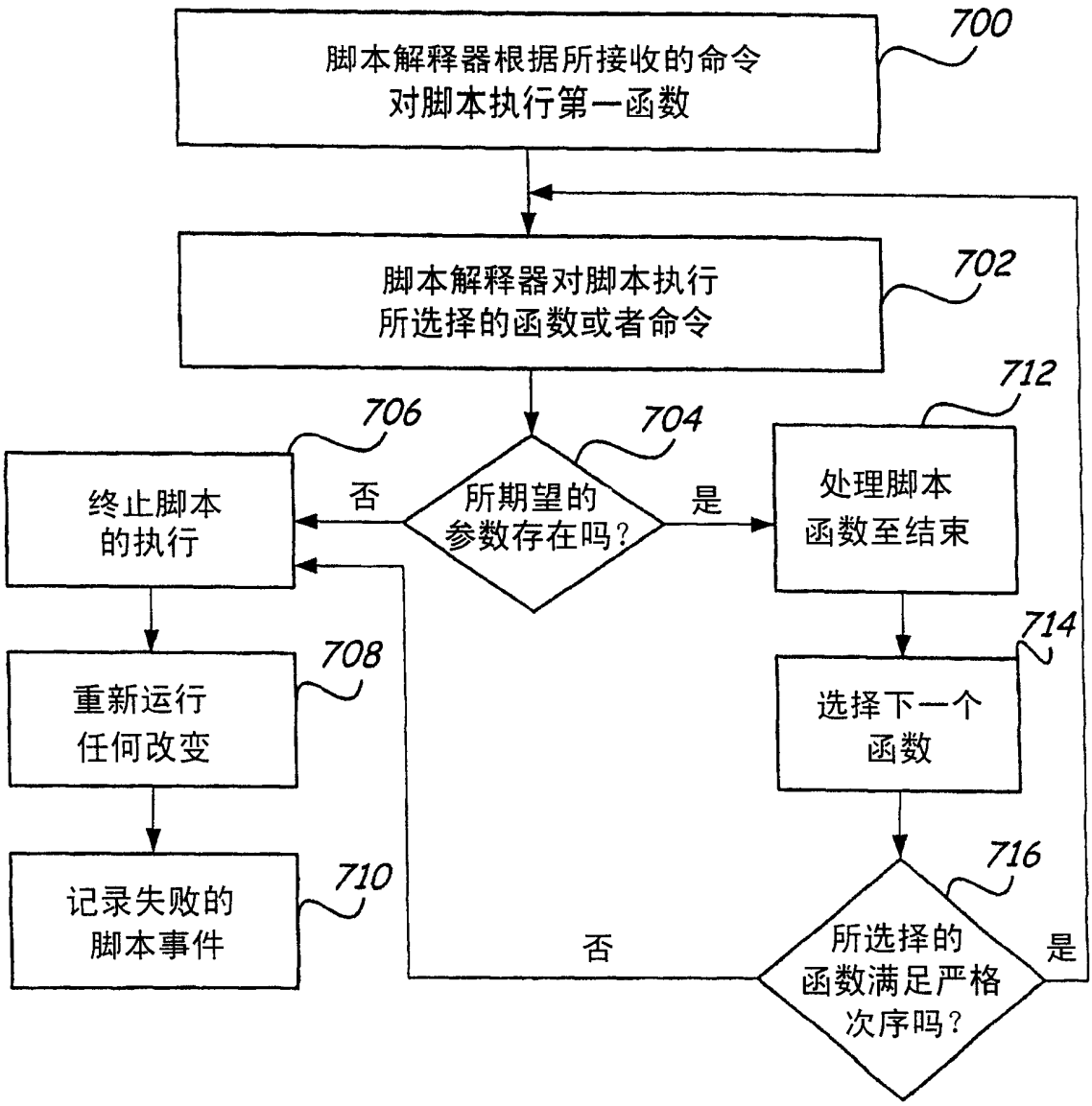

700
脚本解释器根据所接收的命令
对脚本执行第一函数
702
脚本解释器对脚本执行
所选择的函数或者命令
704
所期望的
参数存在吗?
否
是
706
终止脚本
的执行
712
处理脚本
函数至结束
708
重新运行
任何改变
714
选择下一个
函数
710
记录失败的
脚本事件
716
所选择的
函数满足严格
次序吗?
否
是

图 7