

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/445 (2006.01)

H04Q 7/32 (2006.01)



[12] 发明专利说明书

专利号 ZL 02814831.2

[45] 授权公告日 2006 年 9 月 13 日

[11] 授权公告号 CN 1275149C

[22] 申请日 2002.7.22 [21] 申请号 02814831.2

[30] 优先权

[32] 2001. 7. 26 [33] US [31] 09/917,026

[32] 2001. 7. 26 [33] US [31] 09/916,460

[32] 2001. 7. 26 [33] US [31] 09/916,900

[32] 2001. 8. 10 [33] US [31] 09/927,131

[32] 2001.10. 2 [33] US [31] 09/969,305

[86] 国际申请 PCT/IB2002/002875 2002. 7. 22

[87] 国际公布 WO2003/010663 英 2003.2. 6

[85] 进入国家阶段日期 2004.1.29

[71] 专利权人 基奥赛拉无线公司

地址 美国加利福尼亚州

[72] 发明人 高里·拉贾拉姆 保罗·泽肯多夫

迭戈·卡普兰

审查员 崔志鹏

[74] 专利代理机构 北京英赛嘉华知识产权代理有限公司

代理人 葛强 方挺

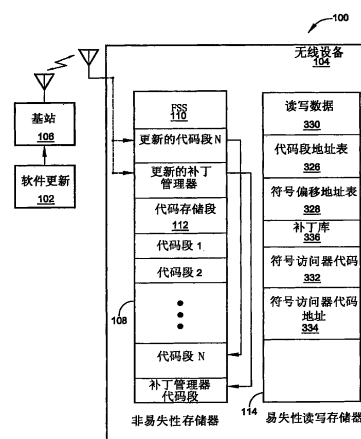
权利要求书 6 页 说明书 26 页 附图 24 页

[54] 发明名称

用于管理无线通信设备系统软件现场下载的系统和方法

[57] 摘要

本发明公开了一种管理无线通信设备中的软件更新操作的方法，所述方法包括：通过空中链路接口接收软件更新，所述软件更新包括新的代码段和指令集；解析所述软件更新，以获得所述指令集；启动通过配置以处理所述指令集的运行时期指令集引擎；基于所述指令集，识别要被替换的当前代码段；以及用新的代码段替换所述当前的代码段。可选择的管理功能包括恢复状态监测、备份、压缩和升级排序。



1. 一种管理无线通信设备中的软件更新操作的方法，所述方法包括：
通过空中链路接口接收软件更新，所述软件更新包括新的代码段和

5 指令集；

解析所述软件更新，以获得所述指令集；

启动配置成处理所述指令集的运行时期指令集引擎；

基于所述指令集，识别要被替换的当前代码段；以及

用新的代码段替换所述当前的代码段。

10

2. 如权利要求 1 所述的方法，其中，所述替换的步骤包括：

将所述新代码段存储在一非易失性存储器的文件系统段中。

3. 如权利要求 1 所述的方法，其中，所述指令集包括恢复状态指令
15 集，所述方法进一步包括执行步骤，所述执行步骤包括：：

将恢复状态表载入一非易失性存储器中，所述状态表具有用于识别
新的代码段的识别段，和用于识别所述新的代码段的软件更新状态的相
关状态段；

在连续用所述新代码段代替所述当前代码段后，更新用于所述新代
20 码段的状态表；

将所述恢复状态表存入非易失性存储器中。

4. 如权利要求 3 所述的方法，其中，所述恢复状态表包括用于识别
与用所述新代码段替换所述当前代码段的操作相关的风险的风险段。

25

5. 如权利要求 4 所述的方法，还包括：

在所述无线通信设备上电后，从所述非易失性存储器中读取所述恢
复状态表；

识别没有被一第二新代码段替换的一第二当前代码段；以及

30 用所述第二新代码段替换所述第二当前代码段。

6. 如权利要求 5 所述的方法, 还包括确定用所述第二新代码段替换所述第二当前代码段的风险。

5 7. 如权利要求 6 所述的方法, 其中, 如果所述新第二代码段包括新引导代码段, 则所述风险是高的。

8. 如权利要求 6 所述的方法, 其中, 如果所述新第二代码段包括新补丁管理器代码段, 则所述风险是高的。

10

9. 如权利要求 6 所述的方法, 其中, 如果所述新第二代码段包括新代码段的地址表, 则所述风险是高的。

10. 如权利要求 6 所述的方法, 其中, 如果所述新第二代码段包括新
15 符号偏移地址表, 则所述风险是高的。

11. 如权利要求 6 所述的方法, 其中, 如果所述新第二代码段包括新符号补充代码地址, 则所述风险是高的。

20 12. 如权利要求 6 所述的方法, 其中, 如果所述新第二代码段包括新读写数据, 则所述风险是高的。

13. 如权利要求 6 所述的方法, 还包括将所述电池功率电平与预定的
25 电池功率电平比较, 以确定是否可以得到充足的电池功率来替换所述当前代码段。

14. 如权利要求 5 所述的方法, 还包括在所述替换的步骤中, 防止用户对所述无线通信设备断电。

30 15. 如权利要求 1 所述的方法, 其中, 所述指令包括数据备份指令,

所述替换步骤还包括:

将非易失性存储器中的代码段存储中的所述当前代码段复制到所述非易失性存储器中的文件系统存储中; 以及

将所述新代码段安装到所述代码段存储中。

5

16. 如权利要求 15 所述的方法, 还包括以下步骤:

为所述新代码段执行代码段测试;

记录所述代码段测试的结果; 以及

10 将所述代码段测试的结果与预定的值进行比较, 以确定所述新代码段是否是有效的。

17. 如权利要求 16 所述的方法, 其中, 所述代码段测试包含在所述软件更新中。

15 18. 如权利要求 16 所述的方法, 其中, 所述代码段测试是所述指令集的一部分。

19. 如权利要求 16 所述的方法, 其中, 所述代码段测试是所述新代码段的一部分。

20

20. 如权利要求 15 所述的方法, 还包括: 用所述非易失性存储器中的文件系统存储中的当前代码段替换所述非易失性存储器中的代码段存储中的新代码段。

25 21. 如权利要求 1 所述的方法, 其中, 所述软件更新包括多个新代码段, 所述指令集包括更新顺序指令集, 所述方法包括:

确定用多个第二新代码段替换多个第二当前代码段的多个风险;

以及基于所述风险, 区分所述多个新代码段的优先安装顺序。

30 22. 如权利要求 21 所述的方法, 其中, 所述安装顺序是首先处理最

低的风险，最后处理最高的风险。

23. 一种管理无线通信设备中的软件更新操作的系统，所述系统包括：

5 空中链路接口，能够通信地将所述无线通信设备耦合到无线通信网络，所述空中链路接口被配置成接收包括新的代码段和指令集的软件更新；以及，

 运行时期引擎，通过配置从所述空中链路接口接收所述软件更新，识别要被新代码段替换的当前代码段，以及用新代码段替换所述当前的
10 代码段。

24. 如权利要求 23 所述的系统，包括：

 非易失性存储器；

 存储在所述非易失性存储器中的恢复状态表，所述恢复状态表包括：
15 用于识别新代码段识别段，和
 用于识别所述新代码段的软件更新状态的相关状态段。

25. 如权利要求 24 所述的系统，其中，所述恢复状态表还包括用于识别与连同所述软件更新一起安装所述新代码段相关的风险的风险段。

20

26. 如权利要求 25 所述的系统，还包括恢复状态管理器，被配置成在所述无线通信设备上电后，读取所述恢复状态表，并将具有不完整状态的第二新代码段安装到所述非易失性存储器的代码段中。

25 27. 如权利要求 26 所述的系统，其中，所述恢复状态管理器被进一步配置成确定与各个具有不完整状态的新代码段相关的风险。

28. 如权利要求 27 所述的系统，其中，如果所述软件更新包括新引导代码段，则所述风险是高的。

30

29. 如权利要求 27 所述的系统，其中，如果所述软件更新包括新补丁管理器代码段，则所述风险是高的。

30. 如权利要求 27 所述的系统，其中，如果所述软件更新包括新代码段的地址表，则所述风险是高的。

31. 如权利要求 27 所述的系统，其中，如果所述软件更新包括新符号偏移地址表，则所述风险是高的。

32. 如权利要求 27 所述的系统其中，如果所述软件更新包括新符号补充代码地址，则所述风险是高的。

33. 如权利要求 27 所述的系统，其中，如果所述软件更新包括新读写数据，则所述风险是高的。

34. 如权利要求 27 所述的系统，还包括将所述电池功率电平与预定的电池功率电平比较，以确定是否可以得到充足的电池功率来替换所述当前代码段。

35. 如权利要求 27 所述的系统，还包括在所述替换操作的过程中，防止用户对所述无线通信设备断电。

36. 如权利要求 27 所述的系统，其中，所述软件更新还包括代码段测试，所述运行时期引擎被配置成执行所述代码测试，并将所述代码段测试的结果与预定的值进行比较，以确定所述新代码段是否是有效的。

37 如权利要求 36 所述的系统，其中，所述代码段测试是所述新代码段的一部分。

38. 如权利要求 36 所述的系统，其中，所述代码段测试是所述指令

集的一部分。

39. 如权利要求 36 所述的系统，其中，所述代码段测试是单独分开的代码段。

5

40. 如权利要求 36 所述的系统，还包括位于文件系统存储中的备用补丁库，所述备用补丁库具有可由所述运行时期引擎执行的指令，用于将所述新代码段从所述文件系统存储中安装到所述代码存储中。

用于管理无线通信设备系统软件现场下载的系统和方法

5 技术领域

本申请要求 2001 年 8 月 10 日提交的、名称为"用于执行无线通信设备动态指令集的系统和方法"的序列号为 09/927, 131 的美国专利申请的优先权, 并与 2001 年 7 月 26 日提交的、名称为"用于现场下载无线通信设备软件代码段的系统和方法"的序列号为 09/916, 900 的美国专利申请, 以及 2001 年 7 月 26 日提交的名称为"用于压缩现场可升级无线通信设备软件代码段的系统和方法"的序列号为 09/9169, 460 的美国专利申请相关。它们都被结合于本文中, 以供参考。

本发明一般涉及无线通信设备, 尤其涉及利用动态指令集管理多种系统软件现场下载的管理功能(例如恢复状态监测、备份、压缩和升级排序)的系统和方法。

技术背景

对已经使用的电话的软件不进行更新是罕见的。一旦电话被制造并销售给公众, 更新可能涉及在软件中被发现的问题。一些更新可能涉及电话的新特性的使用, 或服务供给者提供的服务。其他的更新可能涉及地区问题, 或与特定载波信号相关的问题。例如, 在特定的区域, 载波信号的网络分布可能影响手机的空中链路接口状态, 导致手机产生不期望的行为, 例如不正确的频道搜索, 不正确的呼叫结束, 不正确的音频等。

传统的更新方法是把通信设备(此处是指无线设备, 电话, 电话耳机或手机)送回最近的电信零售/服务市场(outlet), 或制造商来处理这种改变。这种更新所涉及的成本是昂贵的, 并超过底线。进一步, 消费者不方便并可能被激怒。在实践中的方案通常是给消费者新的电话。

无线设备在多种环境中使用，为大量的不同消费者提供不同的用户服务，因此，即使无线设备的软件能够被升级以改善服务，为所有使用者提供统一改进的升级也是不可能的。

如果无线通信设备软件可以便宜地升级，并且不造成消费者的不便，
5 那么就是有利的。

如果无线通信设备软件能被升级，并且消费者在相当长的时间里仍可使用他们的电话，那么就是有利的。

如果无线通信设备软件能被更新，技术服务时间最少，或不需要将设备送到服务机构，那么就是有利的。

10 如果能在代码段区分无线设备系统软件，那么在更新系统软件时，只需替换系统软件的代码段，这将是有益的。如果这些代码段能通过空中链路接口与无线设备通信，那么将是有益的。

如果无线设备能被用于现场更新系统软件的动态加载指令集组操作，将是有益的。

15 如果动态指令集通过监测升级状态、备份关键代码段、完成存储器压缩、及排序更新存储区过程以保护无线装置避免更新错误，那么将是有益的。

发明内容

20 无线通信设备的软件更新材料可向用户提供最适合的产品和用户体验。回收手机而更新软件在商业上是昂贵的。手机制成后，这些更新对于向用户提供增值服务或解决电话使用中发现问题可能是必须的。本发明使得通过空中链路接口现场升级手机在实践中成为可能。尤其是本发明可使无线通信设备执行动态指令集。这些动态指令集可使无线设备
25 “智能化地”或有条件地更新系统软件或系统数据。另外，动态指令集允许无线设备确定是否更新过程已经成功地完成。动态指令集允许关键代码段被存储，以防止升级的代码段是不可操作的。动态指令集也执行存储器压缩和存储区排序。

因而，提供了一种管理无线通信设备中的软件更新操作的方法，所
30 述方法包括：通过空中链路接口接收软件更新，所述软件更新包括新的

代码段和指令集；解析所述软件更新，以获得所述指令集；启动通过配置以处理所述指令集的运行时期指令集引擎；基于所述指令集，识别要被替换的当前代码段；以及用新的代码段替换所述当前的代码段。可选择的管理功能包括恢复状态监测、备份、压缩和升级排序。

- 5 下面将提供上述恢复状态监测管理功能、备份管理功能、压缩管理功能和升级排序管理功能，以及用于管理无线通信设备中系统软件下载操作的系统的详细说明。

附图简要说明

- 10 图 1 是整个无线设备软件维护系统的示意方框图。
- 图 2 是软件维护系统的示意方框图，突出表明通过空中链路接口对指令集的安装。
- 图 3 是本发明的用于在无线通信设备中执行动态指令集的系统的示意方框图。
- 15 图 4 是无线设备存储器的示意方框图。
- 图 5 是图 3 所示的代码段地址表。
- 图 6 是图 3 所示的符号库的符号的详细描述。
- 图 7 是图 3 所示的符号偏移地址表。
- 图 8a 和图 8b 是对由运行时期引擎所存取的操作码(op-code)的说明。
- 20 图 9 是包括图 1-8b 所示特征的表明本发明的在无线通信设备中管理系统软件下载操作的系统的示意方框图。
- 图 10 是图 9 所示的动态指令集中的动态指令集恢复状态监测管理器的示意方框图。
- 图 11 是图 9 所示的恢复状态表的一个示例的表示。
- 25 图 12 是图 9 所示的动态指令集中的动态指令备份管理器的示意方框图。
- 图 13 是图 9 所示的动态指令集中的动态指令压缩管理器的示意方框图。
- 图 14 是图 9 所示的动态指令集中的动态指令升级排序管理器的示意方框图。
- 30

图 15a和 15b是本发明的用于执行无线通信设备中的动态指令集的方法的流程图。

图 16 是动态指令集操作的一个示例的流程图。

图 17 是动态指令集操作的另一示例的流程图。

5 图 18 是动态指令集操作的第三示例的流程图。

图 19 是动态指令集操作的第四示例的流程图。

图 20 是动态指令集操作的第五示例的操作流程图。

图 21 是本发明的用于管理无线通信设备中的系统软件下载操作的方法的流程图。

10 图 22 是图 21 的其它细节的流程图,着重说明了本发明的恢复状态监测功能。

图 23 是图 21 的其它细节的流程图,着重说明了本发明的备份功能。

图 24 是图 21 的其它细节的流程图,着重说明了本发明的压缩功能。

15 图 25 是图 21 的其它细节的流程图,着重说明了本发明的升级排序功能。

优选实施例的详细描述

下面详细描述中的一些部分将用程序、步骤、逻辑块、代码、过程和其他表示在无线设备微处理器或存储器中进行数据位操作的符号表达
20 术语来说明。

这些描述和表达是数据处理领域的技术人员所使用的、与该领域的其他技术人员进行最高效的工作交流的工具。在这里,程序、微处理器执行步骤、应用程序、逻辑块,过程等等一般是所构思的前后一致的、通向期望结果的指令或步骤序列。步骤是那些物理量的所需要的物理操
25 作。虽然不是必须的,通常这些量采用电或磁信号的形式,能被存储、传输、组合、比较以及能在基于无线设备微处理器中被进行其他的操作。主要是由于普遍使用的缘故,把这些信号作为比特、值、单元(element)、符号、字符、项(term)、数等来说明有时被证明是方便的。在物理设备(例如存储器)被提及时,他们是通过总线或其他电连接与其他物理设
30 备相连接的。这些物理设备可被认为是与逻辑处理器或应用程序互相作

用的，因此，是"连接"到逻辑运算。例如，存储器能存储或访问代码，进而，连接运算或应用程序能从存储器调用代码段用于执行。

然而应该意识到，这些以及相似的术语是与合适的物理量联系在一起的，仅仅是这些物理量的便利的标签。除非特别指出，应该意识到，
5 在以下对整个发明的讨论中，使用术语例如"处理"、"连接"、"转换"、"显示"、"提示"、"决定"、"显示"、"认可"等进行的讨论涉及无线设备微处理器系统中的动作和处理，即把计算机系统的寄存器和存储器内的代表物理(电子的)量的数据处理和转换为其他代表在无线设备存储器或寄存器或其他的信息存储、传送或显示设备内的物理量的数据。

10 图 1 是整个无线设备软件维护系统 100 的示意方框图。在对软件维护系统 100 进行总的说明之后，将在下面详细说明本发明的系统软件组织。系统 100 描述了传送系统软件更新材料和指令集组(程序)以及在无线设备内安装被传送的软件的过程。手机制造商开发了系统软件更新材料和补丁管理器运行时期指令(PMRTI)，其作为指令集组或动态指令集，更
15 为公知。系统软件被分进符号库。符号库被排列到代码段。在符号库被更新时，软件更新材料 102 作为一个或多个代码段被传送。软件更新材料被广播到无线通信设备 104 所代表的现场无线设备，或使用公知传统的空中链路 (air) 以及数据或消息传送协议，从基站 106 在个别通信中传输。由于能很容易地调整无线通信设备以处理任何可用的空中
20 (over-the-air) 传送协议从而接收系统软件和 PMRTI 更新材料，因而本发明不限于任何特定的传送格式。

系统软件可被看作是不同的子系统的集合。代码对象能紧紧地结合进这些抽象子系统中一个，结果集合被标注为符号库。这样逻辑分解了源代码 (code base)，软件补丁和修正软件能与这些符号库的一个相关。在
25 最通常的情况下，单一的更新材料与一个，至多二个符号库相关。其余代码数据库、符号库保持不变。

符号库的概念提供了处理代码和常数的机制。另一方面，读写(RW)数据安装到唯一的单独的 RW 库中，该库包含了用于所有库的基于 RAM 的数据。

一旦被无线设备 104 接收，所传送的代码段必须被处理。该无线设备重写非易失性存储器 108 的特定代码段。非易失性存储器 108 包括文件系统段(FSS)110 和代码存储段 112。通常在被传送前，代码段被压缩，以使对 FSS110 的占用最小。通常更新代码段带有它自己的 RW 数据，RW 数据是另一种符号库，包含用于每个符号库的所有 RW 数据。虽然在系统软件运行时，RW 数据装载在随机存取可读写易失性存储器 114 中，但它经常需要存储在非易失性存储器 108 中，因而每次无线设备复位时，它经常被装入随机存取可读写易失性存储器 114 中。这包括 RW 数据第一次装入到随机存取可读写易失性存储器。正如下面更详细解释的那样，RW 数据通常排列在补丁管理器代码段中。

系统 100 包括虚拟表的概念。利用这种表，在一个代码段内的符号库能被修补（更换），而不会破坏(替换)系统软件的其他部分(其他的代码段)。为提高效率，虚拟表在随机存取可读写易失性存储器 114 中执行。代码段地址表和符号偏移地址表都是虚拟表。

更新代码段被无线设备 104 接收，并存储在 FSS110 中。通常无线设备用户接口(UI)将通知使用者新软件可用，使用者对 UI 提示作出响应，确认该通知，发出进行修补或更新操作的信号。可选地，更新操作是自动完成的。在更新过程中，无线设备可能不能进行标准的通信作业。补丁管理器代码段包括非易失性可读写驱动器符号库，该非易失性可读写驱动器符号库也被装入随机存取可读写易失性存储器 114。该非易失性可读写驱动器符号库可使代码段被更新的代码段重写。补丁管理器代码段包括读-写数据，代码段地址表和符号偏移地址表，以及符号访问器代码和符号访问器代码地址(在下面讨论)。在引进更新的代码段时，这些数据中的一部分是无效的；更新的补丁管理器代码段包括对更新的代码段有效的读写数据、代码段地址表和符号偏移地址表。一旦更新的代码段被装入代码存储段 112，无线设备复位。在复位操作之后，无线设备能执行更新的系统软件。也应该理解，补丁管理器代码段可能包括其他上面未讨论过的符号库。这些其他符号库不需要装入可读写易失性存储器 114。

图 2 是软件维护系统 100 的示意方框图，突出显示了经空中链路接口对指令集的安装。除了更新系统软件代码段，维护系统 100 能下载和

安装在本文被称为补丁管理器运行时期指令集(PMRTI)的动态指令集、程序、或补丁管理器指令集(PMIS)。PMRTI 代码段 200 以与上述系统软件代码段同样的方式被传送到无线设备 104。PMRTI 代码段最初存储在 FSS110 中。PMRTI 代码段通常是二进制文件，可作为编译的指令显示在手机上。PMRTI 代码段是足够详尽的，可用于基本数学运算和条件运算。例如，RF 标定的 PMRTI 可执行下列各项操作：

IF RF CAL ITEM IS LESS THAN X (如果 RF CAL 项小于 X)

EXECUTE INSTRUCTION (执行指令)

ELSE (否则)

EXECUTE INSTRUCTION (执行指令)

PMRTI 能支持基本数学运算，例如：加、减、乘、除。与系统软件代码段一样，PMRTI 代码段可被加载以响应 UI 提示，在 PMRTI 被装入代码存储段 112 之后，无线设备必须被重起。随后可以执行 PMRTI 代码段。如果 PMRTI 代码段与任何虚拟表或读-写数据有关，更新的补丁管理器代码段与 PMRTI 同时将被传送，以在代码存储段 112 内安装 PMRTI。可选地，PMRTI 能从 FSS110 中被保持和处理。在手机 104 已执行了 PMRTI 内的全部指令之后，可以从 FSS 110 中删除 PMRTI 段。可选地，保留 PMRTI 以便将来的操作。例如，每次无线设备开机时，可以执行 PMRTI。

PMRTI 是很强大的运行时期指令引擎。通过 PMRTI 环境，手机能执行任何传来的指令。这种机制可被用于支持 RF 标准。更一般的，作为用户控诉的结果。当厂商或服务供给者意识到软件问题时，PMRTI 能用于远程调试无线设备软件。PMRTI 也能记录诊断软件问题所需要的数据。PMRTI 能启动新下载的系统应用软件，用于数据分析、调试和修正。

PMRTI 能提供用来取代更新的系统软件代码段的基于 RW 数据的更新材料，以针对问题进行分析 and 短期修正。PMRTI 能提供无线设备使用的存储器压缩算法。

在本发明的一些方面中，将系统软件组织为符号库的这种结构可影响执行所需要的易失性存储器 114 和非易失性存储器 108 的大小。这通常是由于代码段比排列在代码段中的符号库更大。这些较大代码段用于

容纳更新的代码段。将系统软件作为库的集合进行组织影响所需的非易失性存储器的大小。对于同样大小的代码而言，由于代码段比排列于其中的符号库更大，因而使用的非易失性存储器的量较大。

一旦软件更新材料被传输到无线设备，软件维护系统 100 就支持存储器压缩。存储器压缩与台式计算机中的磁盘碎片整理程序相似。这种压缩方法可确保存储器的最佳使用并有更多的节余，以用于将来的代码段更新材料，而更新代码段的大小是不可预测的。当系统 100 被修补（更新）时，其可分析代码存储段。系统 100 意图将更新代码段装入由被代替的代码段所占据的存储空间。如果更新的代码段比被代替的代码段更大，那么系统 100 在存储器 112 中压缩代码段。可选地，制造商或服务提供者可计算这种压缩，并且压缩指令可被传输到无线设备 104。

由于算法复杂、数据移动量大，压缩是耗费时间的过程。压缩算法在开始处理之前进行可行性分析。在压缩开始前，可用 UI 提示请求使用者的许可。

在本发明的一些情况中，所有的系统软件代码段能被同时更新。但全部系统软件升级需要更大的 FSS 110。

图 3 是本发明的无线通信设备中的动态指令集的执行示意方框图。系统 300 包括在存储器 108 中的代码存储段 112，该代码存储段包括被分成多个当前代码段的可执行的无线设备系统软件。在图中示出了代码段一(302)、代码段二(304)，代码段 n (306)和补丁管理器代码段 308。然而，本发明不受任何具体的代码段数的限制。进一步，系统 300 进一步包含排列到第二组的代码段中的第一组的符号库。如图示出了排列到代码段一(302)中的符号库一(310)、排列在代码段二(304)中的符号库二(312)和符号库三(314)、以及排列在代码段 n(306)中的符号库 NN(316)。每个库包含具有相关功能的符号。例如，符号库一(310)可能涉及到无线设备液晶显示器(LCD)的操作。那么，该库中的符号可能与显示功能有关。正如下面详细说明的那样，其他符号库被排列在补丁管理器代码段 308。

图 4 是无线设备存储器的示意方框图。如图所示，存储器是图 1 所示的代码存储段 112。存储器是可写的非易失性存储器，如闪存（Flash memory）。应该理解，代码段不必象 FSS 110 那样存储在同一存储器中。

而应该理解，在本发明的系统软件结构中，代码段可以存储在多个相配合的存储器中。代码存储段 112 包括第二组的连续地址的存储器块，其中每个存储器块存储有来自第二组的代码段的相应代码段。因而，代码段一(302)存储在第一存储器块 400 中，代码段二(304)存储在第二存储器块 402 中，代码段 n(306)存储在第 n 存储器块 404 中，以及补丁管理器代码段(308)在第 p 存储器块 406 中存储。

对比图 3 和 4，每个代码段的开始相应地存储在存储器的开始地址处，符号库被排列以在代码段的起始处起始。也就是说，每个符号库在第一地址起始并从第一地址起依次占据 (run through) 一定范围的地址。例如，代码段一(302)在代码存储段存储器 112 的第一起始地址 408 (标号为"S")开始。在图 3 中，符号库一(310)在第一代码段的起始地址 318 处起始。同样地，代码段二(304)开始在第二起始地址 410 (图 4)，符号库二开始在代码段二的开始 320 (图 3)。代码段 n(306)开始在代码存储段存储器 112 (图 4)的第三起始地址 412，符号库 NN (316)开始在代码段 n(322)的开始处(图 3)。补丁管理器代码起始在代码存储段存储器 112 的第 P 起始地址 414，补丁管理器代码段 308 的第一符号库从补丁管理器代码段的开始处 324 开始。因而，符号库一(310)最后存储在存储器块 400 中。如果代码段包括多个符号库，例如代码段二(304)，那么多个符号库存储在相应的存储器块中，存储器块 402 就是这种情况。

在图 3 中，系统 300 进一步包含代码段地址表 326，作为一种类型的符号包括在补丁管理器代码段 308 中排列的符号库中。代码段地址表用存储器中的代码起始地址来交叉引用 (cross-reference) 相应的代码段识别符。

图 5 是图 3 中 326 所示的代码段地址表。查询代码段地址表 326 以查找符号库的代码段起始地址。例如，在要求执行符号库一中的符号时，系统 300 寻找代码段一。为查找代码段一的开始地址，并因而定位符号库一中的符号，代码段地址表 326 被查询 (consult)。符号库在代码段中的排列以及利用表对代码段的跟踪，允许代码段被移动或扩充。在安装被升级的代码段(带有被升级的符号库)时可能需要扩充或移动操作。

回到图 3，应该注意到，并不是每个符号库都必需在代码段的开始处开始。如图所示，排列在代码段二(304)中符号库三(314)就不是排列在代码段的开始地址 320 处。因而，如果要执行符号库三(314)中的符号，系统 300 查询代码段地址表 326 以查找代码段二 (304) 的开始地址。正如
5 以下所解释的，符号偏移地址表允许定位符号库三(314)中符号。这些符号跨越多个库是没有问题的，不过这些库要保持在同一代码段中。

如上所述，每个符号库包括与一定功能相关的符号。符号是用于查找和定位例行程序 (routing body)、变量或数据结构的程序员定义的名字。因而，符号可以是地址或值。符号可以是内部的或外部的。内部符号在其所处的代码段范围之外是看图未示的。更具体地，他们不能被在
10 其他的代码段中其他的符号库查找 (seek)。外部符号被跨代码段地使用和调用，并可被在不同代码段中的库查找 (seek)。符号偏移地址表通常包括全部外部符号的列表。

例如，符号库一(310)可在无线设备显示装置上生成字符。在该库中的符号将依次生成电话号码、名字，时间，或其他的显示特性。每个特性都由例行程序生成，例行程序在本文中被称为符号。例如，在符号库一(310)中的一个符号生成显示装置上的电话号码。该符号由"X"表示，是外部符号。当无线设备收到电话呼叫，而且呼叫方 ID 服务是激活的，系统需要执行符号"X"以生成在显示器上显示的号码。因此，系统必须定位
20 符号"X"。

图 6 是对图 3 中的符号库一 (310) 的符号的详细描述。排列符号以从各自的代码段起始地址处偏移。在许多环境中，符号库的开始是代码段的开始，但如果代码段包括多于一个的符号库则不是这样，代码段 (310) 在代码段一的开始处开始(见图 3)。如图 6 所示，符号"X"定位在从符号库开始的偏移(03)处，而"Y"符号被定位在偏移(15)处。符号偏移地址存储在补丁管理器代码段的符号偏移地址表 328 中(见图 3)。
25

图 7 是图 3 中的符号偏移地址表 328。符号偏移地址表 328 利用存储器中的偏移地址和代码段标识符交叉引用相应的符号标识符。因而，在系统搜索执行符号库一中的符号"X"时，符号偏移地址表 328 被查询，以
30 查找该符号在它所排列的代码段中的精确地址。

回到图 3, 第一组的符号库通常都包括在执行这些符号库时必须被查询或设置的读写数据。例如, 符号库可能包括依赖于条件指令的操作。读写数据段被查询以确定完成条件指令所需的状态。本发明将来自全部符号库的读写数据集中到共享读写部分。在本发明的一些方面, 读写数据 330 被排列在补丁管理器代码段 308。可选地(未示出), 读写数据可以排列在不同的代码段, 例如, 代码段 n(306)。

第一组的符号库还包括符号访问器代码, 该符号访问器代码排列在代码段中以计算所搜索的符号的地址。符号访问器代码可以被排列和存储在独立的代码段中, 例如, 代码段二(304)。然而, 如图所示, 符号访问器代码 332 被排列和存储在补丁管理器代码段 308 中。系统 300 进一步包括用于存储符号访问器代码地址的第一位置。第一位置可以是代码存储段 112 中的代码段, 或在无线设备(未示出)的单独的存储器段。第一位置也可以与读写数据一样排列在同一代码段中。如图所示, 第一位置 334 与读写数据 330、符号偏移地址表 328、代码段地址表 326 和符号访问器代码 332 以及补丁库(补丁符号库)336 一起被存储在补丁管理器代码段 308 中。

符号访问器代码访问代码段地址表和符号偏移地址表以计算或查找所寻找符号在存储器中的地址。也就是说, 符号访问器代码用相应的符号标识符和相应的代码段标识符计算所寻找符号的地址。例如, 如果寻找在符号库一中的符号“X”, 调用符号访问器以寻找与符号“X”相应的符号标识符(符号 ID)“X_1”(见图 7), 查询符号偏移地址表以确定“X_1”符号标识符具有从代码段一开始处起的偏移(03)(见图 6), 寻找与代码段一相应的代码段标识符“CS_1”, 查询代码段地址表以确定与代码段标识符(代码段 ID)“CS_1”相关的开始地址。以此方式, 符号访问器代码确定符号标识符“X_1”是从的地址(00100)偏移(03), 或被定位在地址(00103)。

由于符号“X”是实际代码的一部分, 因而它是保留名。换句话说, 它带有与之相关的确定数据。该数据可能是地址或值。符号标识符是为跟踪符号所创建的别名。符号偏移地址表和代码段地址表两者都用标识符工作, 以避免与保留的符号和代码段名相混。同一符号名被许多符号库使用也是可能的。标识符的使用可防止在符号之间产生混乱。

回到图 1，系统 300 进一步包含可读写易失性存储器 114，通常是随机存储器(RAM)。读写数据 330、代码段地址表 326、符号偏移地址表 328、符号访问器代码 332、和符号访问器代码地址 334 从补丁管理器代码段装入可读写易失性存储器 114 以在系统软件执行期间被访问。同样是众所周知的，访问存储在 RAM 中的代码的时间比访问非易失性存储器（如闪存）的时间要短。

回到图 3，虽然被划分的存储器块的大小正好可容纳存储于其内的相应的代码段，但应该注意到，符号库不必装满他们所排列的代码段。可选地，第二组的代码段中的每一个都具有用来容纳符号库的用字节表示的空间（size），每一个地址连续的存储器块都带有容纳相应代码段的以字节表示的空间。例如，代码段一(302)可能是 100 字节的段，以容纳长度为 100 字节的符号库。为与代码段一的字节大小相匹配，第一存储器块将是 100 字节。然而，装入代码段 1 的符号库可能比 100 字节小。如图 3 所示，由于符号库一(310)小于 100 字节，代码段一(302)带有未使用的段 340。因而，第二组的代码段中的每一个都可能比排列符号库所需的大小要大。利用"超大的"代码段，可以容纳更大的更新的符号库。

地址连续存储器块是指将物理存储器空间分区成的大小可变的逻辑块。当代码段被存储在存储器中时，代码段和存储器块基本上是可互换的术语。代码段的概念被用于识别可能比符号库更大的一段代码，或用于在当移动并处理代码段时，识别代码段中的符号库集合。

如图 3 所示，系统 300 包括在本文也被称为补丁库 336 的补丁符号库，用以在代码存储段中将当前代码段和新代码段排列在一起。新代码段和当前代码段在代码存储段中的排列可形成更新的可执行系统软件。补丁管理器 336 不仅可将新代码段和当前代码段排列在一起，而且可用更新的代码段代替代码段。

回到图 4，存储器 108 的文件系统段 110 接收新代码段，如新代码段 450 和更新的补丁管理器代码段 452。文件系统段还接收初次补丁管理器运行时期指令(PMRTI)454，其包含将当前代码段和新代码段排列在一起的指令。如图 1 所示，空中链路接口 150 可接收新的或更新的代码段及初次 PMRTI。尽管图中的空中链路接口 150 是天线，但是应该理解空中

链路接口也包括 RF 收发机、基带电路及解调电路（未示出）。文件系统段 110 可存储通过空中链路接口 150 接收的代码段。从可读写易失性存储器 114 中执行的补丁库 336 可用新的或更新的代码段 450 代替代码存储段中的第一代码段，例如代码段 n (306)，以响应 PMRTI 454。通常用更新的补丁管理器代码段 452 代替补丁管理器代码段 308。当代码段被代替时，补丁库 336 用文件系统段 110 中的更新代码段例如代码段 450 重写代码存储段 112 中的第一代码段，例如代码段 n (306)。在极端情况下，存储段 112 中的所有代码段都被更新的代码段所代替。即 FSS 110 接收第二组的更新代码段(图未示)，并且补丁库 336 可用第二组的更新代码段代替代码存储段 112 中的第二组的代码段。当然，FSS 110 必须足够大以容纳通过空中链路接口接收的第二组的更新的代码段。

如上所述，接收的更新代码段可包括读写数据代码段、代码段地址表代码段、符号库、符号偏移地址表代码段、符号访问器代码段或带有新补丁库的代码段。所有这些代码段和与它们相关的符号库及符号可被存储成不同和独立的代码段。然后这些代码段中的每一个被唯一的更新的代码段所代替。即所接收的更新的读写代码段将代替代码存储段中的读写代码段。所接收的更新的代码段地址表代码段将代替代码存储段中的代码段地址表代码段。所接收的已更新的符号偏移地址表代码段将代替代码存储段中的符号偏移地址表代码段。被接收的更新的符号访问器代码段将代替代码存储段中的符号访问器代码段。同样地，更新的补丁管理器代码段(带有补丁库)将被接收并代替代码存储段中的补丁管理器代码段。

然而上述的代码段通常与补丁管理器代码段捆绑在一起。这样当补丁管理器代码段 308 被更新的补丁管理器代码段 450 代替时，代码存储段中的读写代码段可被来自文件系统段 110 的读写代码段代替。同样地，当安装更新的补丁管理器代码段 450 时，代码段地址表、符号偏移地址表、符号访问器代码段及补丁库将被取代。新读写数据、新代码段地址表、新符号偏移地址表、新符号访问器代码、新补丁库和更新的补丁管理器代码段 450 及代码存储段中的当前代码段的排列形成更新的可执行系统软件。

当文件系统段 110 接收已更新的符号访问器代码地址时，补丁管理器用更新的符号访问器代码地址代替存储器中第一位置的符号访问器代码地址。如上所述，存储器 334 中的第一位置通常是在补丁管理器代码段中(参见图 3)。

5 如图 3 所示，补丁库 308 还包括压缩器或压缩器符号库 342。压缩器 342 也可以是独立的代码段，然而如上所述，将该与系统软件升级相关的功能捆绑进补丁管理器代码段是有用和有效的。通常压缩器 342 也被称为重设大小代码段，从而新段可与当前代码段一起排列在代码存储段 112 中。

10 就目前建立的本发明的结构、下载和压缩方面而言，下面的讨论将集中在无线通信设备动态指令集执行系统 300。如上面所详细讨论的，系统 300 包括可执行的被分成代码段的系统软件和系统数据。此外，系统 300 包括用于操作系统数据和系统软件并控制系统软件的执行的动态指令集。如图 4 所示，动态指令集 470 被组织进第一 PMRTI 454。如图 3
15 所示，系统还包括用于处理动态指令集的运行时期引擎，被表示为运行时期库 370。与如上所述的压缩器库 342 和补丁库 336 一样，运行时期库 370 通常位于补丁管理器代码段 308 中。然而，可选择地，运行时期库 370 可位于另一个代码段中，例如第一代码段 304 中。

动态指令集是单一或含有条件操作代码的多重指令集组，并且通常
20 包括数据项。运行时期引擎可读取操作代码并决定需要进行何种操作。操作代码可以是条件代码、数学代码、程序代码或逻辑代码。运行时期引擎或运行时期库 370 处理动态指令集以执行诸如数学操作或逻辑操作等操作。即运行时期引擎可根据操作代码读取动态指令集 470 并进行一系列操作。

25 尽管动态指令集不限于任何特定的语言，但由于无线设备存储器空间有限并且执行速度很重要，所以操作代码通常是机器代码的形式。由于操作代码分析数据项并根据分析结果做出决定，因而它被认为是条件代码。运行时期引擎也可决定在数据被分析之前对其进行操作。

例如操作代码可指定将从无线设备存储器来的数据项与预定值做比较。
30 较。如果数据项小于预定值，那么不用处理它，如果数据项大于预定值，

那么就用预定值代替它。可选地，在进行上述的比较操作前，操作代码可使从无线设备存储器来的数据项增加第二预定值。

如上所述，非易失性存储器文件系统段 110 可通过接口如空中链路 150 接收动态指令集。如图 1 所示，接口也可以是射频(RF)硬线 160。这样，在没有系统软件正在工作的情况下，如在工厂校准环境中，PMRTI 可被 FSS 110 接收。PMRTI 也可通过逻辑端口接口 162 或可安装的存储器模块 164 来接收。存储器模块 164 可安装在无线设备 104 中，可在初始校准时安装，或现场安装，或在工厂校准时安装。尽管没有特别说明，但 PMRTI 可通过红外接口或蓝牙接口来接收。

图 8a 和 8b 是对被运行时期引擎 370 所访问的指令的说明。在图 8a 中所示的是第一指令 800、第二指令 802 和第 j 指令 804，然而动态指令集不限于任何特定数量的指令。在每条指令中的操作代码是固定的。运行时期引擎 370 可获得指令的长度，即字节或位的数量，来检测指令是否包括数据项。减去操作代码后，剩余指令长度包括数据项。运行时期引擎可从指令中提取数据项。如图所示，第一指令 800 的长度 806 被测量并提取了 808。应注意并不是所有的指令都必须包括将被提取的数据项。在依据指令 800 中的操作代码 810 执行的一系列操作时，运行时期引擎 370 使用提取的数据 808。

图 8b 是图 8a 中的第一指令 800 的更详细说明。以第一指令 800 为例，指令包括操作代码 810 和数据 808。指令，更具体地是指数据项段 808，包括符号标识符，其用于连接无线设备代码段中的符号。如上所详细说明的，符号标识符和代码段地址表 326(参见图 5)和符号偏移地址表 328(参见图 7)一起使用以定位与符号标识符相应的符号。如图所示，在第一指令 800 中示出了符号标识符“X_1”。符号偏移地址表 328 可利用标识符“CS_1”和偏移量“3”在代码段中定位相应符号。代码段地址表 326 可给出代码段一(302)的起始地址。以这种方式可找到符号“X”(参见图 6)。

在运行时期引擎在使用代码段地址表和符号偏移地址表定位了与所接收的符号标识符相应的符号之后，在被定位的符号是数据项时它提取数据。例如，如果符号库一(310)中的符号“X”是数据项时，运行时期引擎就可提取它。可选择地，符号“X”可以是操作代码，并且当其被定位时运

行时期引擎可执行符号“X”。

PMRTI 也可被用于更新系统数据或数据项。在本发明的某些方面中，参见图 4，系统数据存储在文件系统段 110 的代码段中，例如在代码段 472 中。运行时期引擎可访问来自于代码段 472 的系统数据并分析该系统数据。如上所述，运行时期引擎可处理动态指令集中的操作代码以对数据项执行数学或逻辑操作。此操作后，运行时期引擎可处理指令以产生更新的系统数据。应注意到在某些情况下更新的系统数据可包括未改变的数据项。根据操作代码，在第二代码段 472 中的系统数据被更新的系统数据代替。这样，通过运行时期引擎对指令的处理，使用代码段 472 中的已更新的系统数据控制系统软件的执行。以这种方式，系统软件中的特定目标符号可被更新，而不用替换全部的代码段。通过相同的处理，可以替换代码存储段 112 的代码段中的系统数据。例如如果系统数据存储在第三代码段 344 中，运行时期引擎可根据操作代码用更新的系统数据替换第三代码段中的系统数据。

PMRTI 也可被用于更新易失性存储器 114 中的数据项。例如参见图 1，易失性存储器 114 接收读写数据 330。读写数据可从代码存储段 112 和/或 FSS 110 中的一个或多个代码段得来。运行时期引擎可访问读写数据、分析读写数据 330、产生更新的读写数据并用已更新的读写数据代替易失性存储器 114 中的读写数据 330 以响应操作代码。然后使用易失性存储器 114 中的已更新的读写数据来控制系统软件的执行。

在本发明的某些方面中，运行时期引擎可监测系统软件的执行。性能监测被广义地定义，以包括最大数量的无线设备活动。例如，通过一系列操作可导致特定失效条件或性能下降的诸如信道参数、信道特性、系统栈、出错条件等数据或 RAM 中数据项记录可被收集。也可使用动态指令集分析收集到的性能数据、提供已更新的数据变量并再次捕获数据来研究针对问题的可能解决方案。也可使用 PMRTI 处理提供临时性的修正。

更具体地，运行时期引擎可以根据操作代码收集性能数据并将性能数据存储在文件系统段中。然后，通过收集用于评估系统软件性能的数据控制执行系统软件。评估可以作为分析的一种形式被动态指令集操作

代码进行，它也能在无线设备外部进行。在本发明的某些方面中，运行时期引擎可访问从文件系统段收集的性能数据并通过空中链路接口传输性能数据以响应操作代码。从无线设备现场收集性能数据允许制造商全面地分析问题（在本地或全球任何地方），而不用将设备回收。

5 在本发明的某些方面中，文件系统段 110 可接收包括新代码段的管理器运行时期指令。例如图 4 中所示的新代码段 474。可选择地，新代码段可以是与 PMRTI 无关的，如新代码段 n (450)。例如，新代码段 n (450) 可在早期的空中链路通信中被接收或在工厂校准期间被安装。运行时期引擎可将代码段 474 (450) 加到代码存储段中以响应操作代码。在本发明
10 的某些方面中，新代码段代码被加到存储段 112 中的未用块中。可选择地，压缩操作是需要的。然后，使用新代码段 474 (450) 控制执行系统软件。在本发明的其它方面中，PMRTI 454 包括更新的代码段 474。可选择地，新代码段 450 是与 PMRTI 无关的更新的代码段。运行时期引擎可用更新的代码段 474 (450) 代替代码存储段中的代码段，例如代码段二(304)，
15 以响应操作代码。使用更新的代码段 474 (450) 控制执行系统软件。在本发明的某些方面中，需要压缩操作以容纳更新的代码段。可选择地，已更新的代码段被加到代码存储段的未用段或空白段中。

 如上所述，因为这些操作通常包括新的和/或变化的代码段起始地址，新代码段的加入或代码段的更新通常需要产生新代码段地址表。此外，
20 压缩操作也需要新代码段地址表。如上所述，压缩操作也可以是操作的压缩器 342 结果，或者是可提供怎样压缩的细节的 PMRTI 指令的结果。当 PMRTI 包括下载和压缩指令时，PMRTI 通常也包括在下载和压缩操作完成后有效的新的代码段地址表。

 图 9 是包括图 1-8b 所示的特征的示意方框图，用来说明本发明的管
25 理在无线通信设备中的系统软件下载操作的系统。系统 900 包含一个空中链路接口 902（与图 1 所示的空中链路接口 150 等同）和被分为代码段并储存在非易失性存储器永久存储区 904（与图 1 所示的存储器 108 等同）的可执行的系统软件和系统数据。非易失性永久存储区 904 包括一个文件系统段 906 和代码存储段 908。

30 用于对系统软件更新材料下载进行管理的动态指令集 910 被空中链

路接口 902 接收。动态指令集 910 与新代码段 912 都是补丁管理器运行时期指令 914 的一部分。通常，动态指令集 910 被储存在文件系统段 906 中。运行时期引擎或运行时期库 916 处理动态指令集 910。如前文所述，运行时期库 916 通常是补丁管理器代码段 918 的一部分。根据对动态指令集 910 的处理，可执行的系统软件和系统数据（永久存储区 904 的代码段）被更新。动态指令集 910 包含被选择的功能管理器，可选择的功能管理器有恢复状态监测功能管理器、备份功能管理器、压缩功能管理器和更新排序功能管理器。

系统 900 进一步包含用来以新代码段标识符的更新状态交叉引用新代码段标识符的恢复状态表 920。在文件系统段 906 中恢复状态表 920 被显示为一个独立的代码段。在本发明的一些方面中，当系统 900 启动时，本发明的恢复状态表 920 被装入易失性存储器 922。在易失性存储器 922 中，状态表 920 被更新，并且在系统被关闭时，重新存储在永久存储区 904 中。

图 10 是图 9 所示的动态指令集中的动态指令集恢复状态监测管理器 1000 的示意方框图。在无线通信设备重新启动之后，动态指令集恢复状态监测管理器（以后被称为恢复状态监测管理器 1000）读取恢复状态表 920（见虚箭头 "1" 所示），来确定新代码段是否已经被储存在永久存储区中。

图 11 是图 9 所示的恢复状态表 920 的一个示例的表示。恢复状态表 920 显示出标识为 CS-1 和 CS-2 的代码段已经被成功地从文件系统段 906 移到代码存储段 908。然而，该表显示出标识为 CS_n 的代码段是否已经被成功地储存是未知的。例如，在储存操作被完成之前，用户可能已经把无线设备关掉，或无线设备电池能源耗尽。

回到图 10，如果新代码段没有被存储，恢复状态监测管理器 1000 在永久存储区存储新代码段，并且在新代码段被存储时更新恢复状态表。例如，恢复状态监测管理器 1000 在代码存储段 908 中存储 CS_n (1002)（见虚箭头 "2" 所示），然后更新恢复状态表，以便 CS_n 状态被改变为 "载入"（未显示）。应注意，在代码段从文件系统段 906 移动到代码存储段 908 中时，通常要使用上文说明的补丁管理器功能。

在本发明的一些方面中，恢复状态监测管理器 1000 进一步确定与储存每个新代码段相关的风险性，并且如果储存新代码段的风险性是高的，则采取安全预防措施。例如：恢复状态监测管理器 1000 执行安全预防措施，诸如，检查电池组是否有足够的电能来完成新代码存储的过程，提醒用户代码的存储操作的高风险，用提示来确认用户启动的断电，以及防止用户启动的断电。回到图 11，风险性因子作为一系列列在恢复状态表中。

一些代码段的存储涉及用更新的代码段替换系统软件中关键代码段。例如，文件系统段 906 可能接收新代码段，诸如根代码段，补丁管理器代码段，代码段地址表，符号偏移地址表，读-写数据和符号访问器代码地址(未示出)。这些代码段的任何一段存储不完全，都可能导致无线设备在启动时彻底不工作。也就是说，无线设备就不得被送到服务中心重新编程。因此恢复状态监测管理器 1000 确定在文件系统段中存储这些新代码段的每一段的风险性，从而采取相应的安全预防措施。

文件系统段有能够接收包括新的补丁管理器代码段(例如 CS_n (1002))的补丁管理器运行时期指令。如前所述，补丁管理器代码段包括用于将文件系统段 906 的代码段移动到永久存储区(通常是代码存储段 908)的补丁库。在本发明的一些方面，恢复状态监测管理器 1000 维护在文件系统段 906 中的备用补丁库 1004。恢复状态监测管理器 1000 读取恢复状态表 920 以确定重新启动无线通信设备后是否新补丁管理器代码段已被成功地存储在代码存储段。如果新补丁管理器代码段 1002 没有被成功地存储在代码存储段，备用补丁库 1004 就被用来将新补丁管理器代码段 1002 从文件系统段 906 中移动到代码存储段 908。然后，新补丁管理器代码段 1002 能够被用于系统软件操作。如果没有备用补丁库 1004，无线设备不能从新补丁管理器代码段的不完全的存储中恢复。

图 12 是图 9 所示的以动态指令集中的动态指令集备份管理器 1200 为特征的系统 900 的示意方框图。动态指令集备份管理器在下文被称为备份管理器 1200。如该名称所暗示的，备份管理器提供一个系统，用于用系统软件恢复已被不可操作的代码段替换的代码段或系统数据。例如：文件系统段 906 接收更新的第一代码段 1202 替换代码存储段 908 中的

第一代码段 1204。备份管理器 1200 通常使用补丁管理器代码段 1206 的库将第一代码段 1204 从代码存储段 908 移动到文件系统段 906 (如虚线“1”所示)。利用补丁管理器代码段 1206 的库,更新的代码段 1202 被从文件系统段 906 移到代码存储段 908 (如虚线“2”所示)。备份管理器
5 1200 确定代码存储段 908 中的更新的第一代码段 1202 能否在预定约束内与系统软件工作。如果更新的第一代码段 1202 被确定不能在预定约束内与系统软件工作,备份管理器 1200 用在文件系统段 906 中的第一代码段 1204 替换在代码存储段 908 中更新的第一代码段 1202。

在本发明的一些方面中,文件系统段 906 接收带有预定约束的测试
10 代码段 1208。通过执行带有更新代码段的测试代码段,备份管理器 1200 确定代码存储段 908 中的更新的第一代码段 1202 能否在预定约束内与系统软件工作。备份管理器记录运行测试代码段 1208 的结果,并将记录的结果与预定约束做比较。在本发明的其它方面,文件系统段 906 接收带有测试代码库更新的代码段 1202。然后,备份管理器 1200 运行更新的代
15 码段中的测试代码库,就是说独立的测试代码段不必作为补丁管理器运行时期指令的一部分。

文件系统段 906 能从各种来源接收测试代码段,例如空中链路接口(图 9 所示的 902)、用户可操作的键盘接口(图未示)、测试接口、测试
端口接口如逻辑端口、串行端口或是射频端口、甚至存储器模块。

20 图 13 是图 9 所示的以动态指令集中的动态指令压缩管理器 1300 为特征的系统 900 的示意方框图。如上述说明的,压缩管理器负责操纵代码储存段中的代码段以与新代码段相适应,或容纳比被代替的代码段大的更新的代码段。例如,文件系统段 906 接收更新的第一代码段 1302,其代码段尺寸已更新,去替换具有比更新的第一代码段尺寸小的第一尺寸的第一代码段 1304。压缩管理器 1300 访问在补丁管理器代码段 1308
25 中的压缩库 1306,并调整代码存储段 908 中代码段大小以容纳更新的第一代码段 1302。

就象恢复状态监测管理器一样,压缩管理器 1300 确定在代码储存段中压缩代码段的风险性,如果压缩代码段的风险性是高的,就采取安全
30 预防措施。例如,由于无线设备不能从不正确的或没完成的压缩操作中

恢复，因而压缩补丁管理器代码段或系统软件的其他关键部分是有风险的。压缩管理器 1300 执行安全预防（措施），例如：检查电池组是否有充足的能量来完成新代码存储的过程，警告用户代码存储操作的高风险性，用提示确认用户进行的断电，和防止用户进行的断电。

5 图 14 是图 9 所示的以动态指令集中动态指令更新排序管理器 1400 为特征的系统 900 的示意方框图。当多个新的或更新的代码区要被储存时，更新排序管理器 1400 决定储存的次序。例如：文件系统 906 收到带有多个新代码段的更新排序动态指令集或更新排序管理器 1400，诸如新的第一代码段 1402，新的第二代码段 1404 和新的第三代码段 1406。更新排序管理器 1400 将新的代码段 1402 -1406 从文件系统储存 906 按排序管理器 1400 指定的顺序移动到代码储存段 908。

10 如图 14 所示，新的第二代码段 1404 被首先移动(如虚线"1"所示)，第一新代码段 1402 随后被移动(如虚线"2"所示)，第三新代码段 1406 再被移动(如虚线"3"所示)。移动和存储操作是在补丁管理器代码段 1408 的帮助下完成的。在本发明的一些方面中，排序管理器 1400 确定储存每个新代码段 1402 -1406 的风险性，并在低风险储存段移动之后，移动高风险代码段。在上面的例子中，更新排序管理器 1400 已确定储存新的第三代码段 1406 风险性最高。

20 图 15A 和 15b 是本发明的用于在无线通信设备中执行动态指令集的方法流程图。虽然为清楚起见，用连续编号的步骤进行描述，但除非明确指出，不应从这些编号(下面示出的本方法的编号)中得出先后顺序。该方法在步骤 1500 开始。步骤 1501a 将系统软件设置成符号库，每个符号库包含具有相关功能的符号。步骤 1501b 将符号库排列到代码段。步骤 1502 执行系统软件。步骤 1503 启动运行-时期引擎。通常，所启动的运行-时期引擎包括调用来自第一代码段的运行-时期库。运行-时期引擎能从易失性或非易失性存储器中启动。在步骤 1503 之后的步骤 1504 中，接收动态指令集。在步骤 1504 中接收动态指令集包括通过从一个组中所选的接口接受动态指令集，该组包括空中链路、射频(RF)硬线(hardline)、可安装存储器模块、红外的以及逻辑端口界面。在本发明的一些方面，

25 在步骤 1504 中接收动态指令集包括接收非易失性存储器的文件系统段内

30

的补丁管理器运行时期指示(PMRTI)。

步骤 1506 处理动态指令集。处理动态指令集包括根据数学和逻辑运算处理指令。在本发明的某些方面中，在处理动态指令集之后，步骤 1507(图未示)删除动态指令集。步骤 1508 操作系统数据和系统软件。步骤 1510 根据对系统软件和系统数据的操作，控制系统软件执行。

在步骤 1504 中接收补丁管理器运行时期指令通常包括接收条件操作代码和数据项。然后，在步骤 1506 中处理动态指令集包括子步骤。步骤 1506a1 使用运行时期引擎读取补丁管理器运行时期指令操作代码。步骤 1506b 根据操作代码进行一系列操作。

在某些方面，在步骤 1501b 中将符号库排列为代码段的操作包括在代码段的开始处开始符号库并从各自的代码段起始地址排列需偏移的符号。本发明还包括如下步骤。步骤 1501c 在相应的起始地址存储起始代码段。步骤 1501d 维护代码段地址表(CSAT)，代码段地址表用相应的起始地址交叉引用代码段标识符。步骤 1501e 维护符号偏移地址表(SOAT)，该符号偏移地址表用相应的偏移地址和相应的代码段标识符交叉引用符号标识符。

在本发明的某些方面中，在步骤 1504 中接收补丁管理器运行时期指令包括接收符号标识符。本方法还包括如下步骤。步骤 1506a2 通过使用代码段地址表和偏移地址表定位与接收的符号标识符相应的符号。根据操作代码在步骤 1506b 中进行的一系列操作包括一些子步骤。当定位的符号是数据项时，步骤 1506b1 提取数据。当定位的符号是指令时，步骤 1506b2 执行符号。

在本发明的某些方面中，在步骤 1506b1 中处理动态指令集包括其它子步骤。步骤 1506b1a 利用运行-时期引擎获取管理器运行时期指令长度。步骤 1506b1b 根据操作代码从管理器运行时期指令中提取数据项。步骤 1506b1c 根据操作代码使用提取的数据进行一系列操作。

图 16 是动态指令集的一个例子的操作流程图。图 16 中的几个步骤和图 15 中的相同，为了简洁起见这里不再重复。在步骤 1606 中处理动态指令集包括一些子步骤。步骤 1606a 访问存储在文件系统段中第二代代码段中的系统数据。步骤 1606b 分析此系统数据。步骤 1606c 创建更新的系

统数据。在步骤 1608 中操作系统数据和系统软件包括用更新的系统数据代替第二代代码段中的系统数据，在步骤 1610 中控制执行系统软件包括在系统软件的执行中使用更新的系统数据。

图 17 是动态指令集另一示例的操作流程图。图 17 中的几个步骤和图 15 中的相同，为了简洁这里就不再重复。步骤 1701c 在非易失性存储器的代码存储段存储多个代码段。在步骤 1706 中处理动态指令集包括一些子步骤。步骤 1706a 访问存储在代码存储段(CSS)中第三代代码段中的系统数据。步骤 1706b 分析此系统数据。步骤 1706c 创建更新的系统数据。在步骤 1708 中操作系统数据和系统软件包括用更新的系统数据代替第三代代码段中的系统数据。在步骤 1710 中对系统软件执行的控制包括在系统软件的执行中使用更新的系统数据。

图 18 是动态指令集第三示例的操作流程图。图 18 中的几个步骤和图 15 中的相同，为了简洁这里不再重复。步骤 1801c 在非易失性存储器的代码存储段存储多个代码段。步骤 1801d 将读写数据加载入易失性存储器。在步骤 1806 中处理动态指令集包括子步骤。步骤 1806a 访问易失性存储器中的读写数据。步骤 1806b 分析此读写数据。步骤 1806c 创建更新的读写数据。在步骤 1808 中操作系统数据和系统软件包括用更新的读写数据代替易失性存储器中的读写数据。在步骤 1810 中控制系统软件执行包括在系统软件的执行中使用更新的读写数据。

图 19 是动态指令集操作的第四示例的流程图。图 19 中的几个步骤和图 15 中的相同，为了简洁这里就不再重复。处理动态指令集包括子步骤。步骤 1906a 监测系统软件的执行以响应操作代码。步骤 1906b 收集性能数据。步骤 1906c 存储此性能数据。步骤 1906d 通过空中链路接口传输存储的数据。在步骤 1908 中操作系统数据和系统软件包括在系统软件的评估中使用此性能数据。步骤 1910 控制系统软件的执行。

图 20 是动态指令集操作的第五示例的流程图。图 20 中的几个步骤和图 15 中的相同，为了简洁这里就不再重复。步骤 2001c 在非易失性存储器的代码存储段存储多个代码段。在步骤 2003 中接收补丁管理器运行时期指令包括接收新代码段。在步骤 2008 中操作系统数据和系统软件包括将新代码段加入到代码存储段中，在步骤 2010 中对执行系统软件的控制

制包括在系统软件的执行中使用新代码段。

可选择地，在步骤 2003 中接收新代码段包括接收更新的代码段。在步骤 2008 中操作系统数据和系统软件包括用更新的代码段代替代码存储段中的第四代码段。

5 图 21 是本发明的用于管理无线通信设备中的系统软件下载操作的方法的流程图。该方法从步骤 2100 开始。在步骤 2102 中将系统软件设置为符号库，每个符号库包含具有相关功能的符号。在步骤 2104 中将符号库排列为储存在非易失性存储器的代码存储段中的代码段。在步骤 2106 中执行系统软件。在步骤 2108 中启动运行-时期引擎。步骤 2110 接收在文件系统段非易失性存储器中的补丁运行时期指令（PMRTIS），补
10 丁管理器运行时期指令包括动态指令集和新代码段。在步骤 2112 中对动态指令集进行处理。在步骤 2114 中依据对动态指令集的操作，管理通过空中链路接口接收的系统软件更新材料的下载。步骤 2116 执行更新的系统软件。

15 在步骤 2114 中对通过空中链路接口接收的系统软件更新材料的下载的管理包括处理动态指令集管理功能。动态指令集管理功能包括恢复状态监测功能、备份功能、压缩功能和更新排序功能。

 图 22 是图 21 所示的附加细节的流程图，着重说明了本发明的恢复状态监测功能。在步骤 2112 中处理恢复状态监测动态指令集包括一些子
20 步骤。在步骤 2200 中，维护用新代码段的更新状态交叉引用新代码段标识符的恢复状态表。在重新启动无线通信设备后，在步骤 2202 中读取恢复状态表，确定新代码段是否已被存储在永久存储区中。如果新代码段没有被存储，在步骤 2204 中在永久存储区中存储新代码段。当新代码段已经被存储时，在步骤 2206 中更新恢复状态表。

25 在本发明的一些方面包括一些附加步骤。步骤2202a确定存储的每一新代码段的风险性。如果存储的新代码段风险性高，步骤2202b采取安全预防措施。在步骤2102b中的安全预防措施包括所选择的安全预防措施，可选择的安全预防措施有检查电池组是否有充足的能量来完成新代码存储过程，警告用户代码存储操作的高风险性，用提示确认用户启动的断
30 电，和防止用户启动断电。

在步骤2202a中确定存储每一新代码段的风险性，包括确定从启动代码段、补丁管理器代码段、代码段地址表、符号偏移地址表、读写数据和符号访问器代码地址中所选的新代码段的风险性。

在本发明的一些方面中，将系统软件设置成符号库（图21所示的步骤2102）包括形成带有补丁库的补丁管理器代码段以将在文件系统段中的代码段移动到永久存储区。在文件系统段非易失性存储器中接收补丁管理器运行时期指令（图21所示的步骤2110）包括接收新补丁管理器代码段。随后，操作恢复状态监测动态指令集包括进一步的子步骤。步骤2202c在文件系统段维护备用补丁库。在重新启动无线通信设备后，在步骤2202中读取恢复状态表，以确定新补丁管理器代码段是否已被成功地存储在代码存储段中。如果新补丁管理器代码段没有被成功存储在代码存储段，步骤2202d利用备用补丁库将新补丁管理器代码段从文件系统段移动到代码存储段。

图 23 是图 21 所示的附加细节的流程图，着重说明了本发明的备份功能。接收新代码段（图 21 所示的步骤 2110）包括接收更新的第一代码段以替换代码存储段中第一代码段。然后，在步骤 2112 中对备份管理器动态指令集的处理包括一些子步骤。步骤 2300 将第一代码段从代码存储段移动到文件系统段。步骤 2302 确定代码存储段中的更新的第一代码段是否可在预定约束内与系统软件操作。如果更新的第一代码段被确定为不能在预定约束内与系统软件操作，在步骤 2304 中用来自文件系统段的第一代码段替换代码存储段中的更新的第一代码段。

在本发明的一些方面中，接收新的代码段（图 21 所示的步骤 2110）包括接收带有预定约束的测试代码段。然后，在步骤 2302 对代码存储段中的更新的第一代码段能否在预定约束内与系统软件操作的确定包括有一些子步骤。步骤 2302a 执行带有更新的代码段的测试代码段。步骤 2302b 记录执行测试代码段的结果。在步骤 2302c 中用预定约束比较所记录的结果。

在本发明的一些方面中，接收新代码段（图 21 所示的步骤 2110）包括接收带有测试代码库的更新代码段。然后，在步骤 2302 中确定代码存储段中更新的第一代码段能否能在预定约束内与系统软件工作，包括执

行更新的代码段中测试代码库。接收新代码段（图 21 所示的步骤 2110）包括接收带有预定约束的测试代码段，可通过从包括空中链路接口、用户可操作的键盘接口以及测试接口的组中选择的接口来接收。

图 24 是图21所示的附加细节的流程图，着重说明了本发明的压缩功能。接收新代码段（图21所示的步骤2110）包括接收更新的第一代码段，带有更新的代码段尺寸，替换比更新的第一代码段尺寸小的第一代码段。在步骤2112中对压缩管理动态指令集的处理包括一些子步骤。步骤2400访问补丁管理器代码段中的压缩库。在步骤2402中调整代码存储段中的代码段的大小以容纳更新的第一代码段。

10 在本发明的一些方面中，本发明包括进一步的步骤：在步骤2400a中确定在代码储存段中压缩代码段的风险性。如果压缩代码段的风险性是高的，在步骤2400b中采用安全预防措施。在步骤2400b中采用执行安全预防措施包括采取从如下一组措施中选择的安全预防措施，即：检查电池组是否有充足的能量来完成新的代码存储过程，警告用户代码存储操作的高风险，用提示确认用户启动的断电，和防止用户开始断电。

图 25 是图 21 所示的附加步骤的流程图，着重说明了本发明的更新排序功能。接收动态指令集（图 21 所示步骤 2110）包括接收更新排序指令，接收新的代码段包括接收多个新代码段。步骤 2112 对更新排序管理动态指令集的操作包括将新的代码段从文件系统存储区按照排序指令指定的顺序移动到代码存储段。

20 在本发明的一些方面中，在步骤 2112 中操作处理更新排序动态指令集包括子步骤。步骤 2500 确定存储每一新代码段的风险性。在步骤 2502 中，将高风险代码段排序在低风险储存段被移动之后移动。

已经提供了用于在无线通信设备中执行动态指令集的系统和方法，从而有助于管理软件更新材料的下载。因为代码段中符号库的排列，并用表访问存储器中代码段的起始地址和符号库中符号偏移地址，所以系统容易被更新。动态指令集的使用允许根据设备的具体特征对其进行定制修改。举出了几个常用实施例用以阐明动态指令集的可能应用。然而，本发明不仅限于这些实施例。本领域所属技术人员可以想到本发明的其它变化和实施方案。

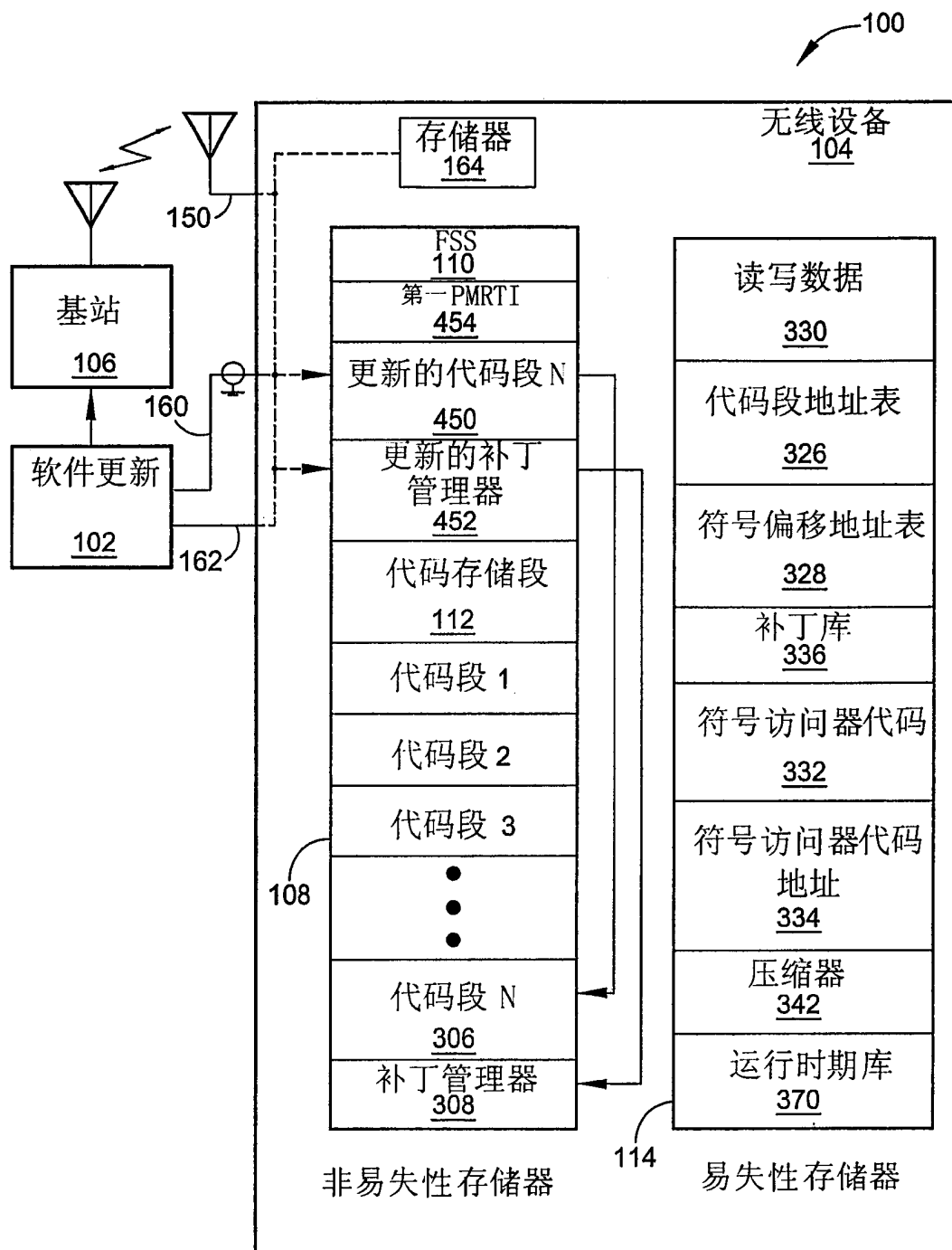


图 1

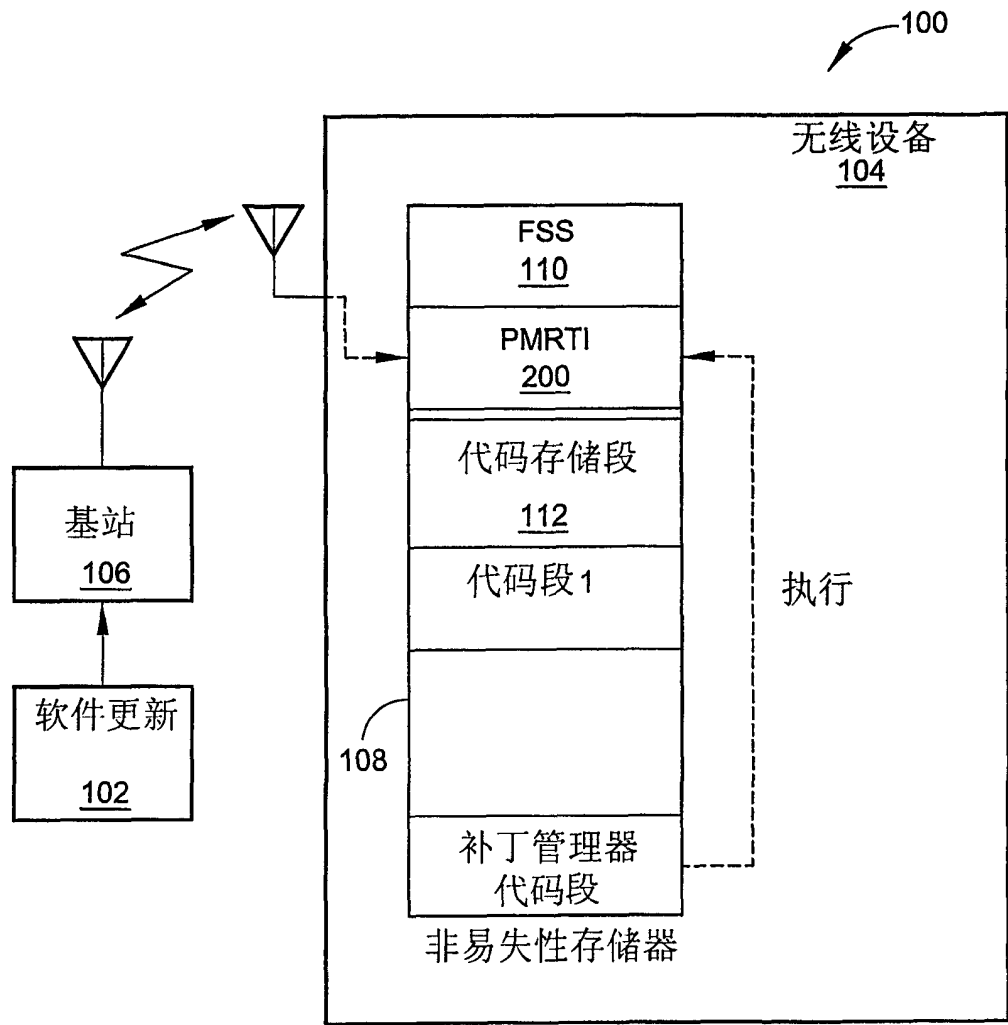


图 2

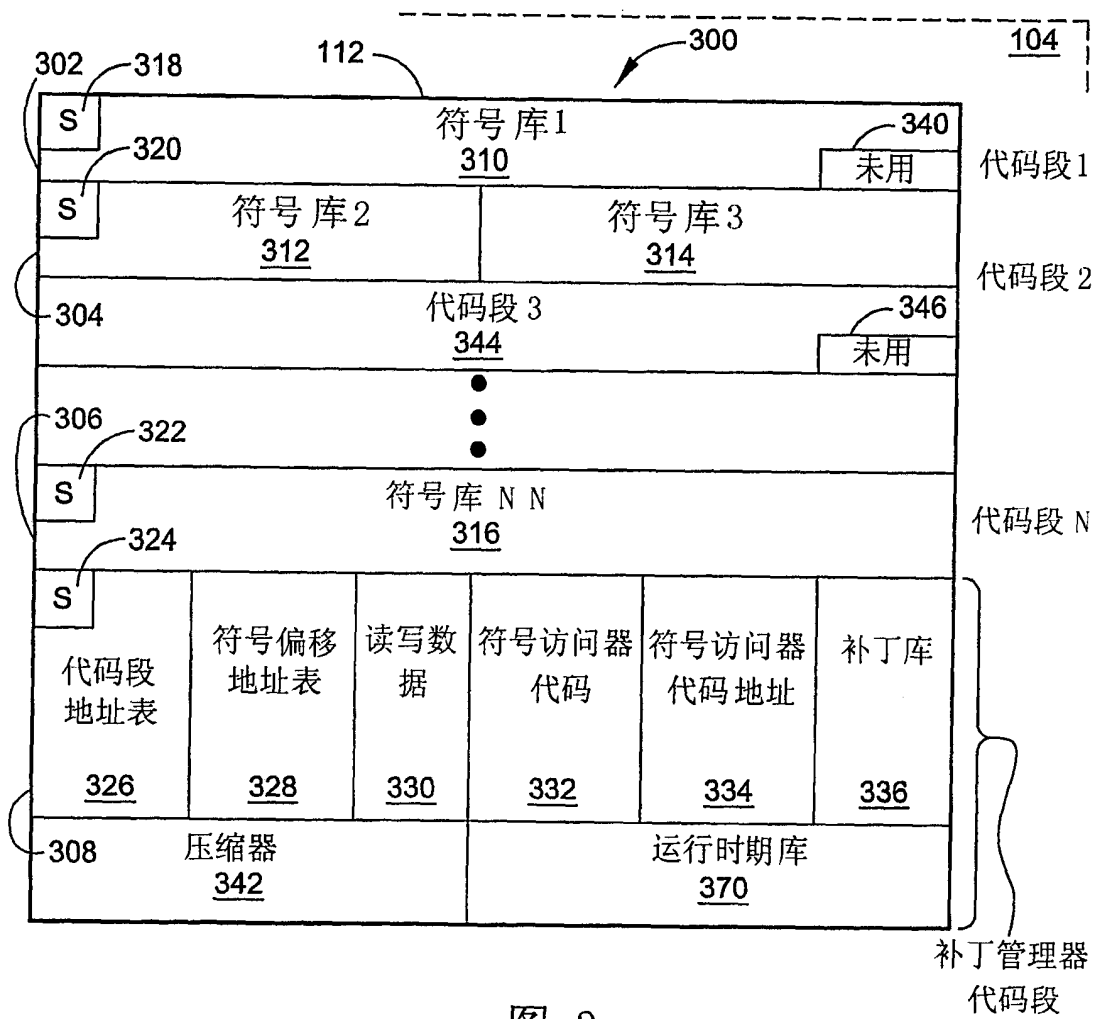


图 3

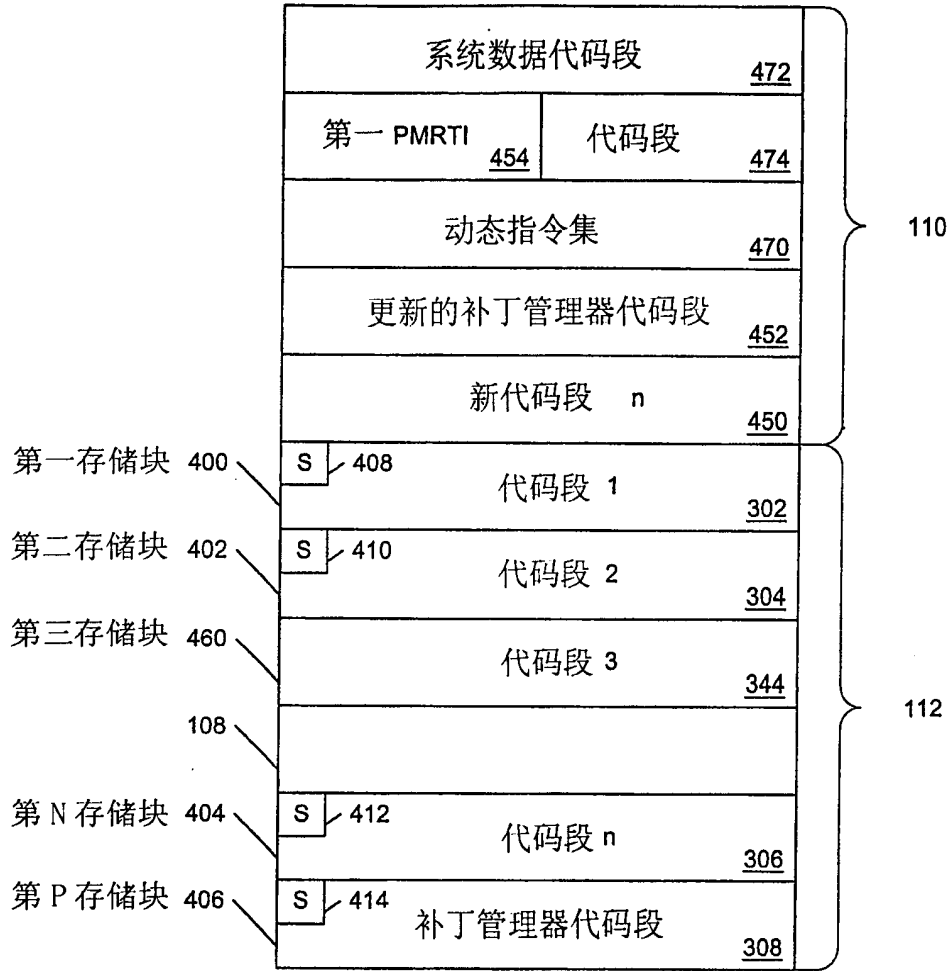


图 4

326

代码段地址表	
标识符	地址
CS_1	开始地址 1 (00100)
CS_2	开始地址 2 (00200)
⋮	
CS_N	开始地址 N (00700)
PM	开始地址 P (01000)

图 5

	0	1	2	3	4	5	6	7
0				X				
1						Y		
⋮	⋮							
n					Z			

310

符号库 1

图 6

符号偏移地址表		
符号 ID	代码段 ID	偏移
X_1	CS_1	03
Y_1	CS_1	15
P_1	CS_2	11
Q_1	CS_2	33
AA_3	CS_2	47
• • •		

328

图 7

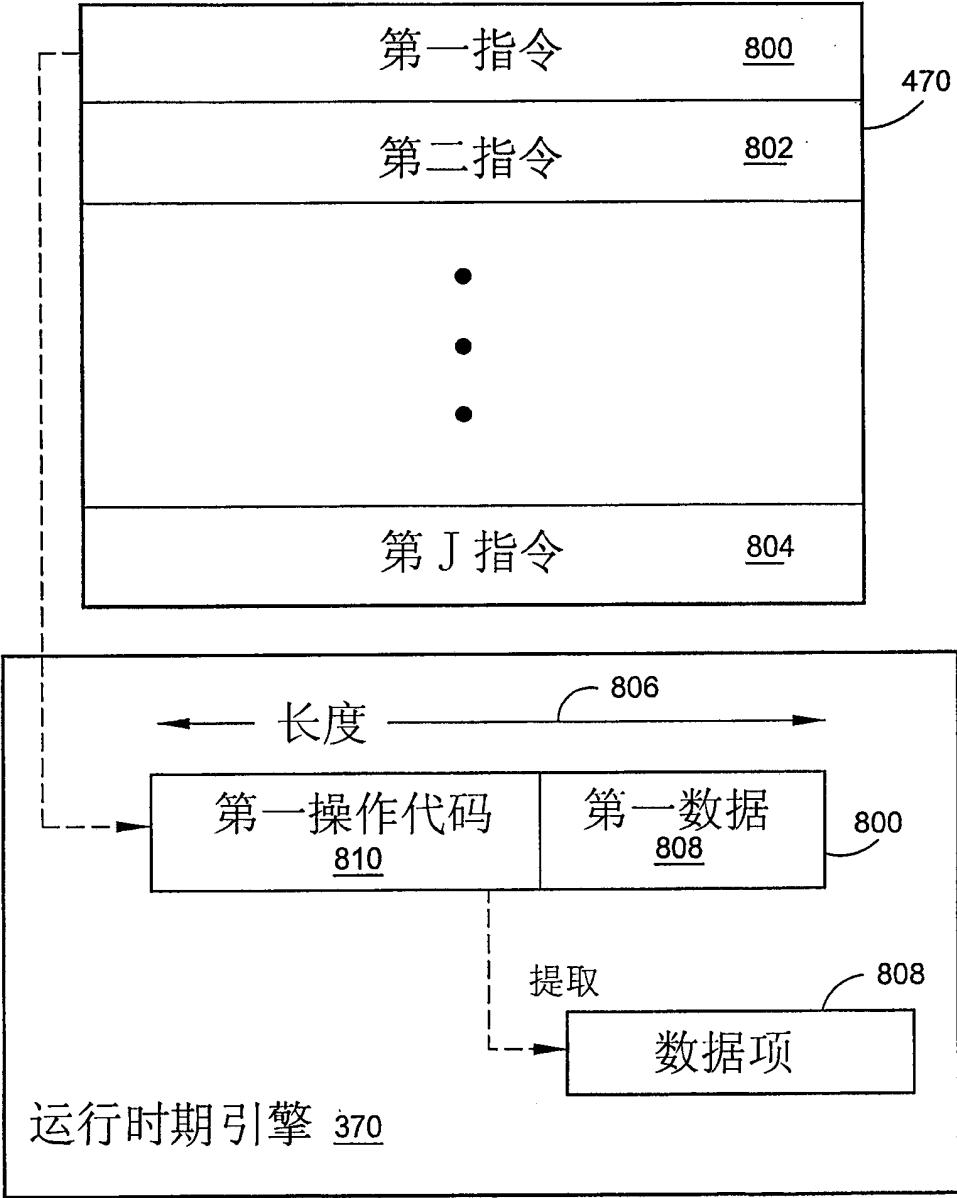


图 8a

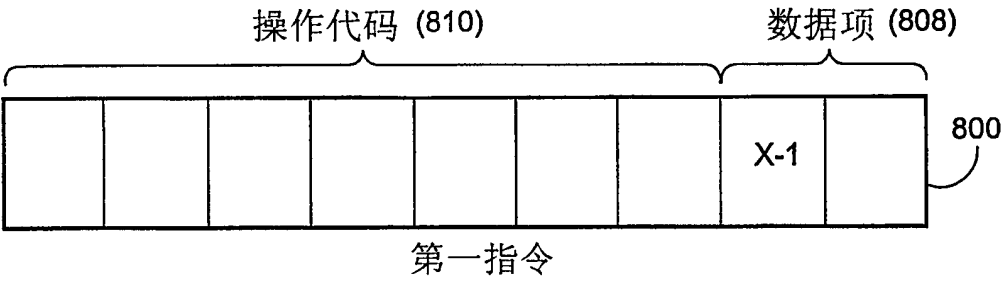


图 8b

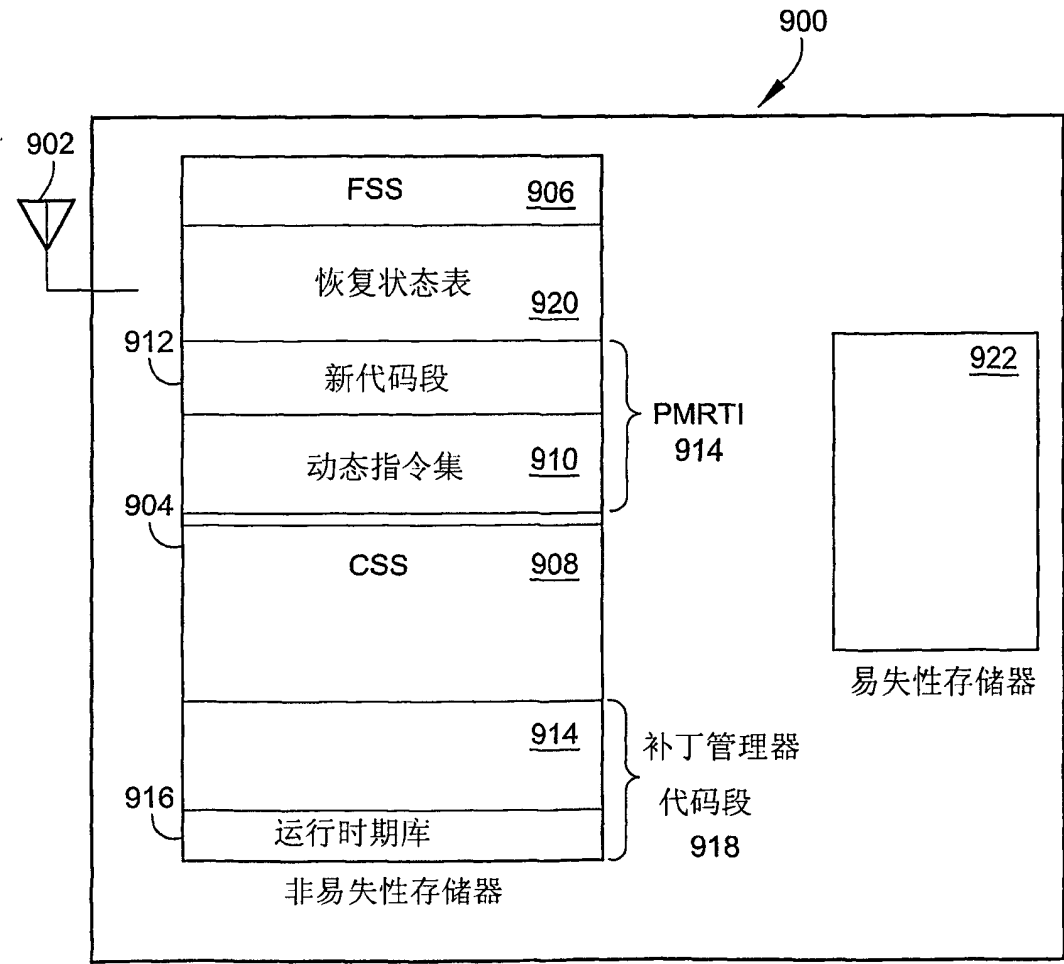


图 9

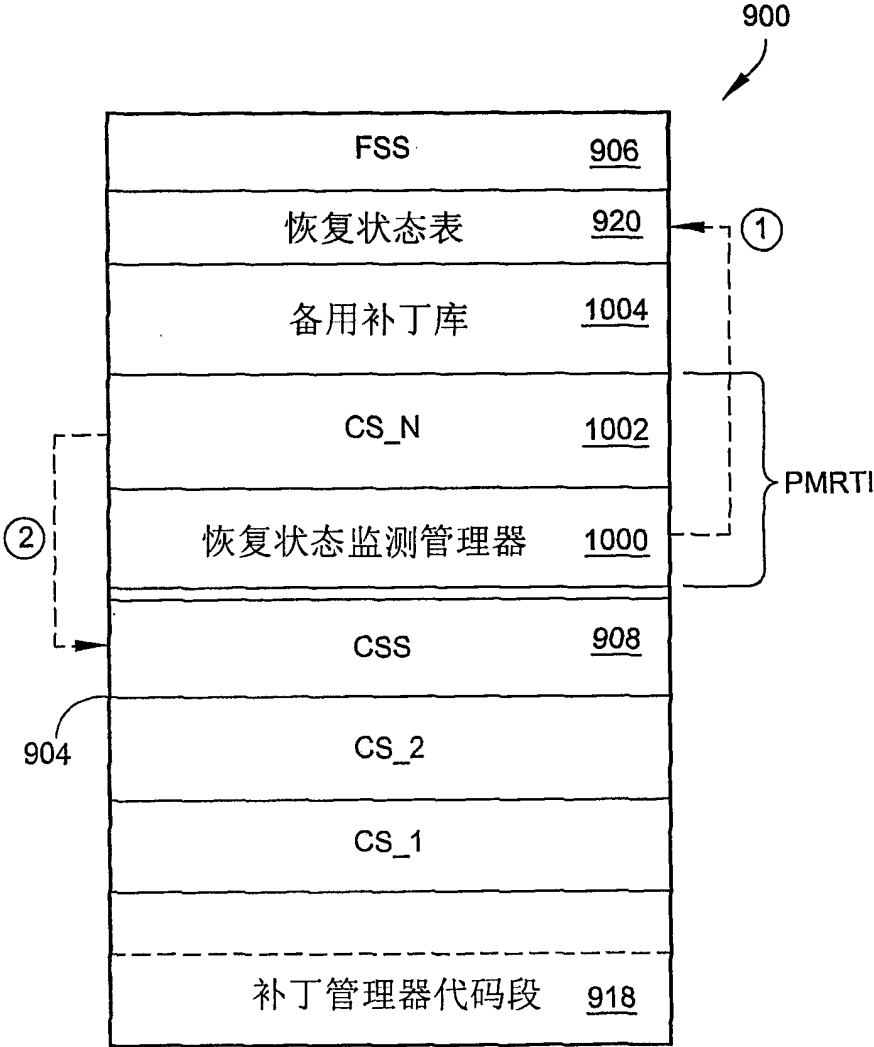


图 10

920

恢复状态表		
标识符	状态	风险性
NCS_1	已装入	高
NCS_2	已装入	低
• • •	• • •	• • •
NCS_N	未知	高

图 11

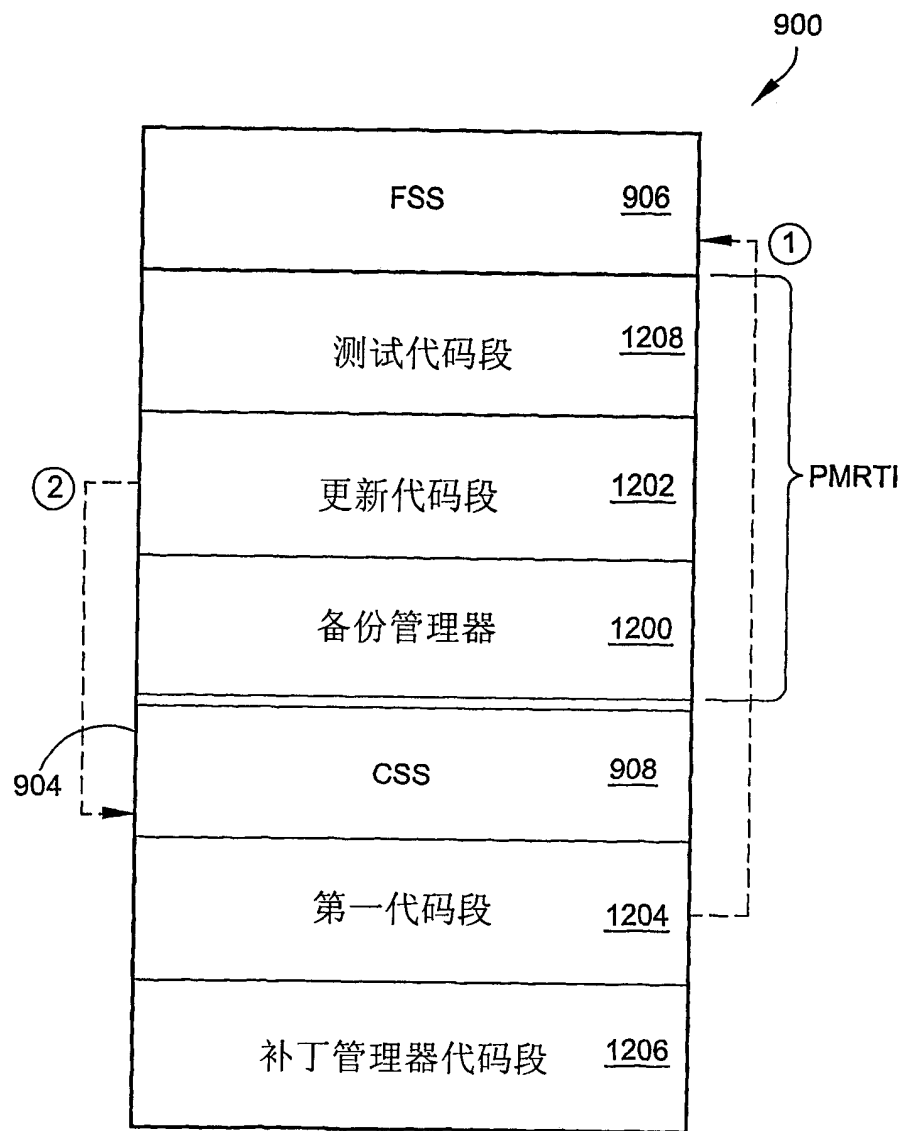


图 12

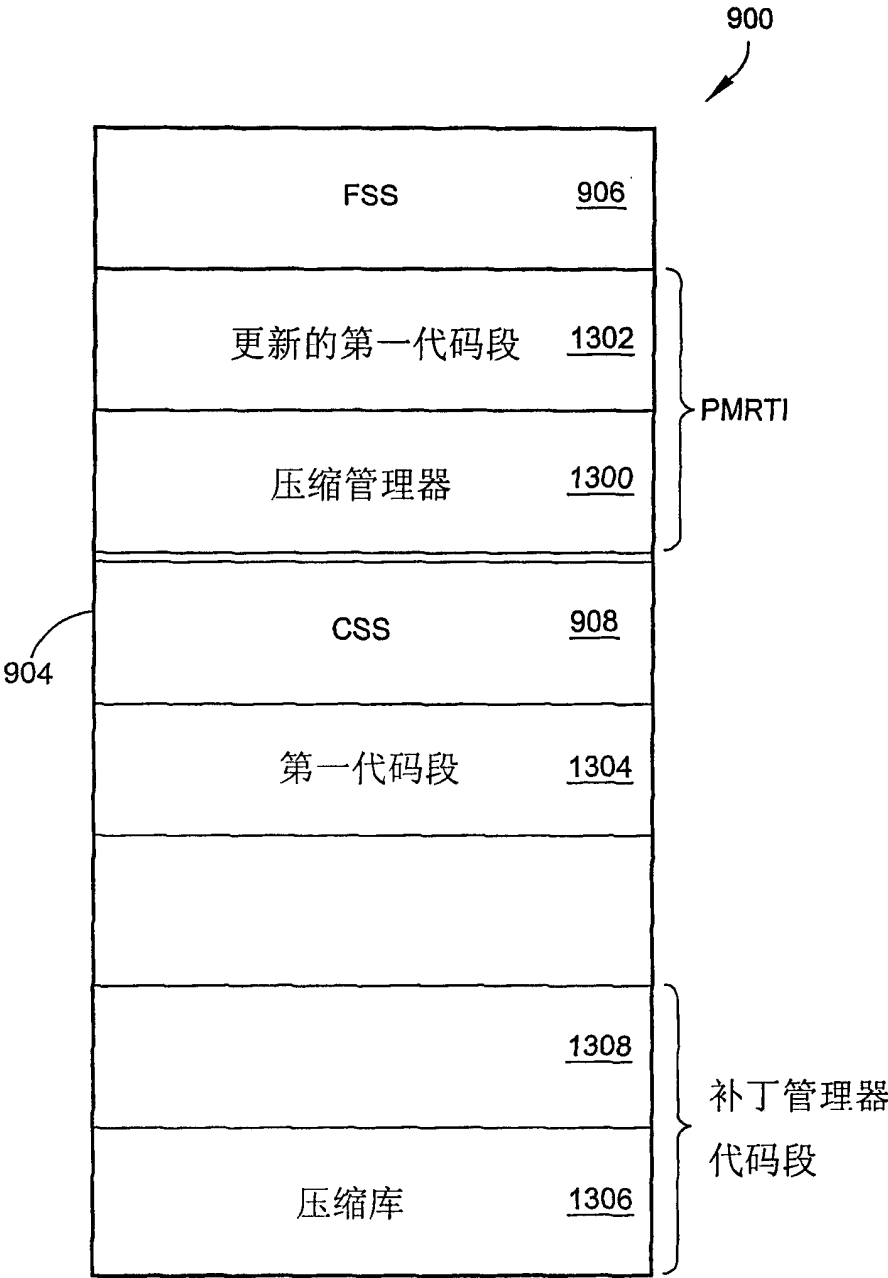


图 13

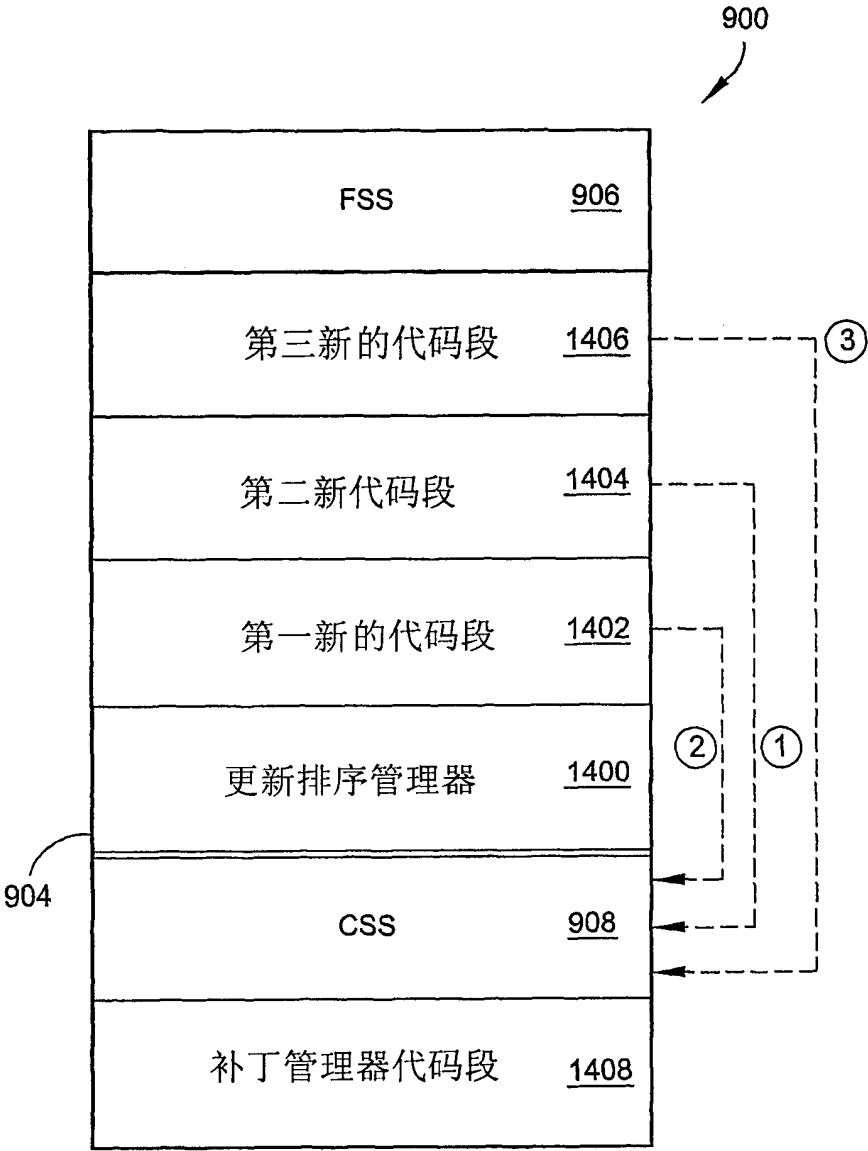


图 14

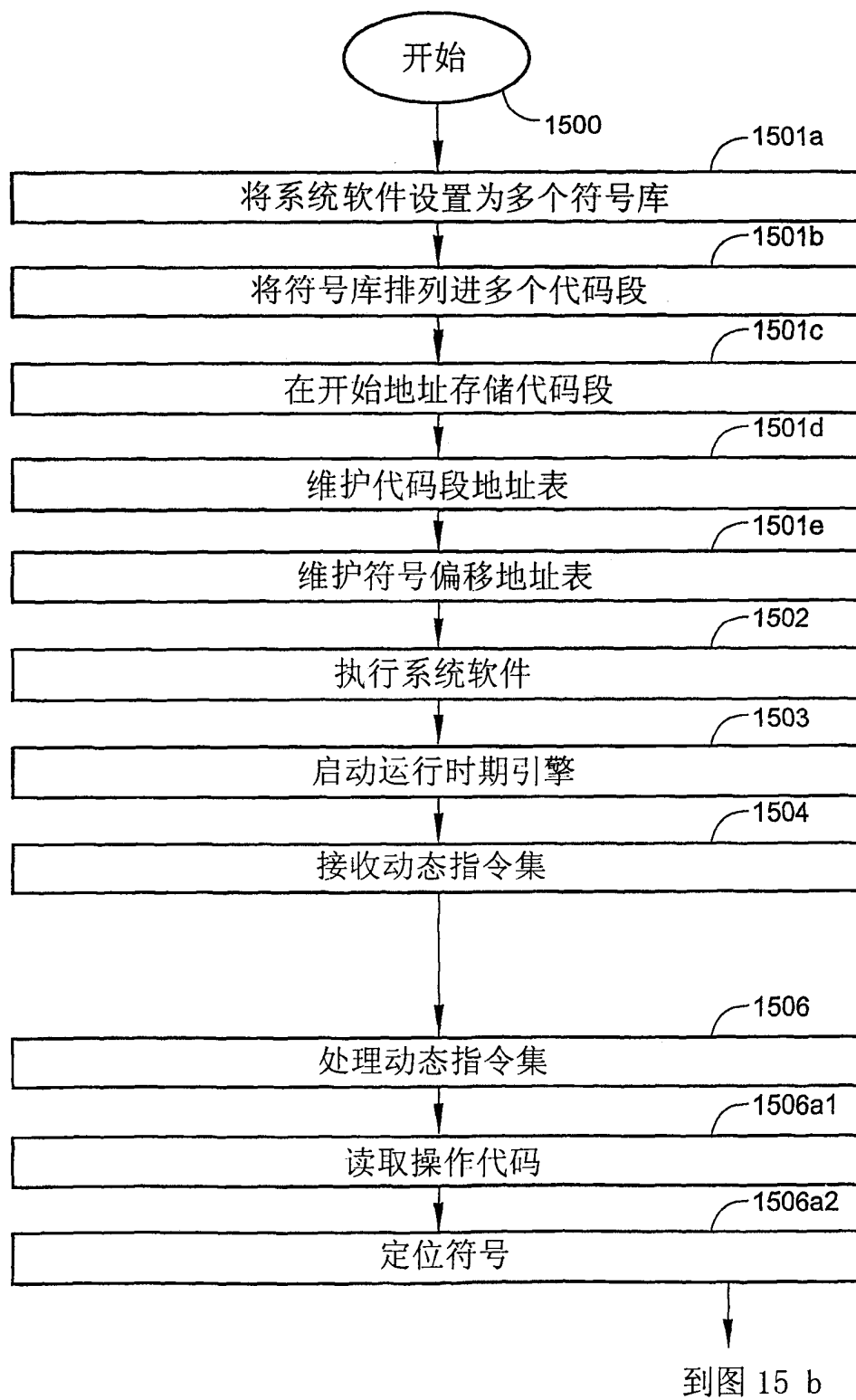


图 15a

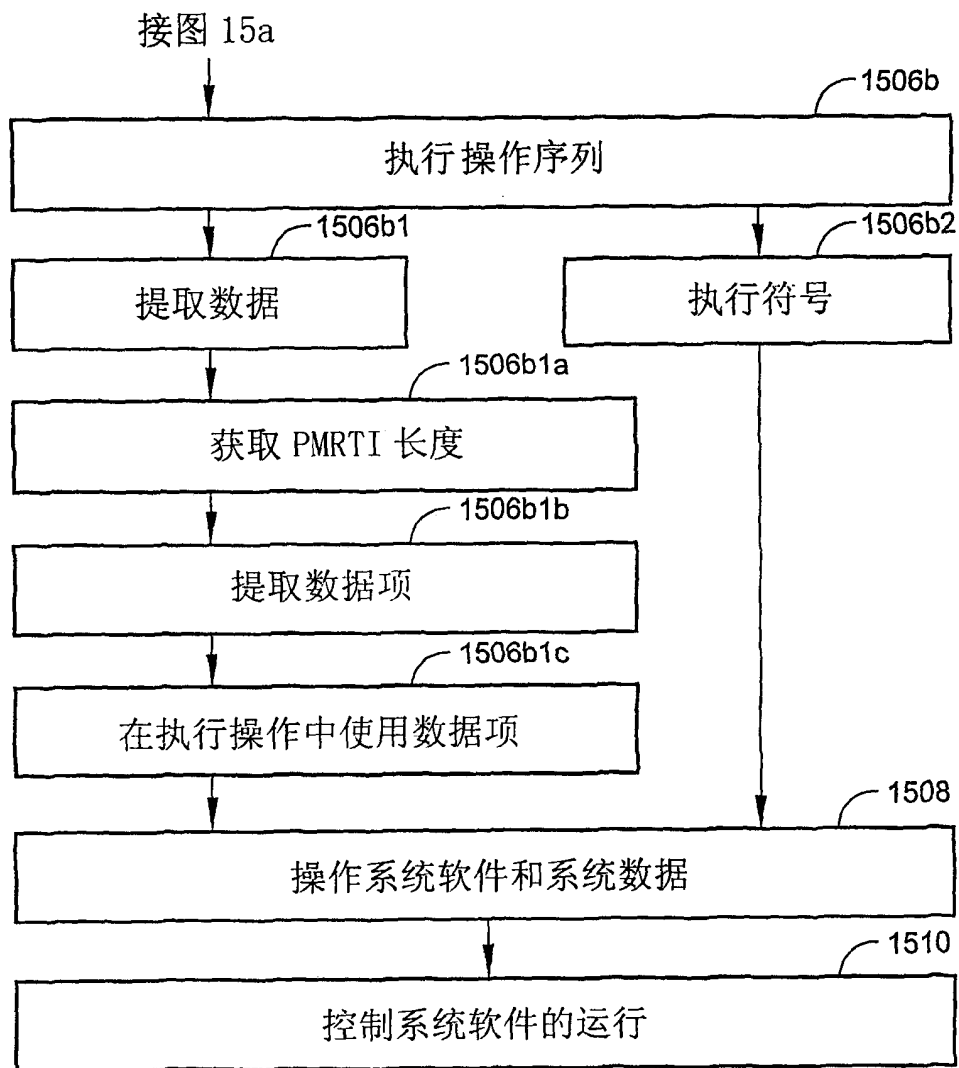


图 15b

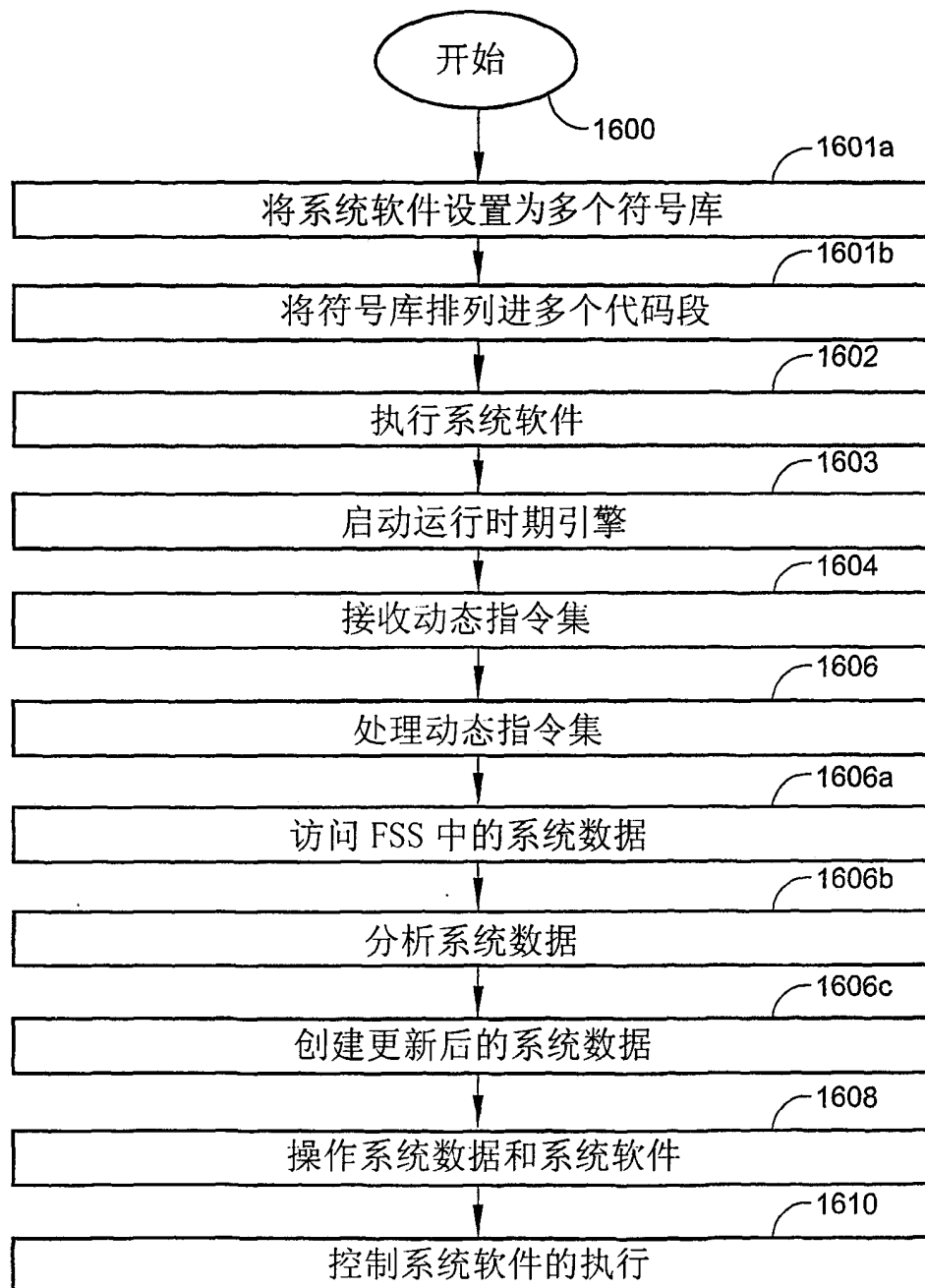


图 16

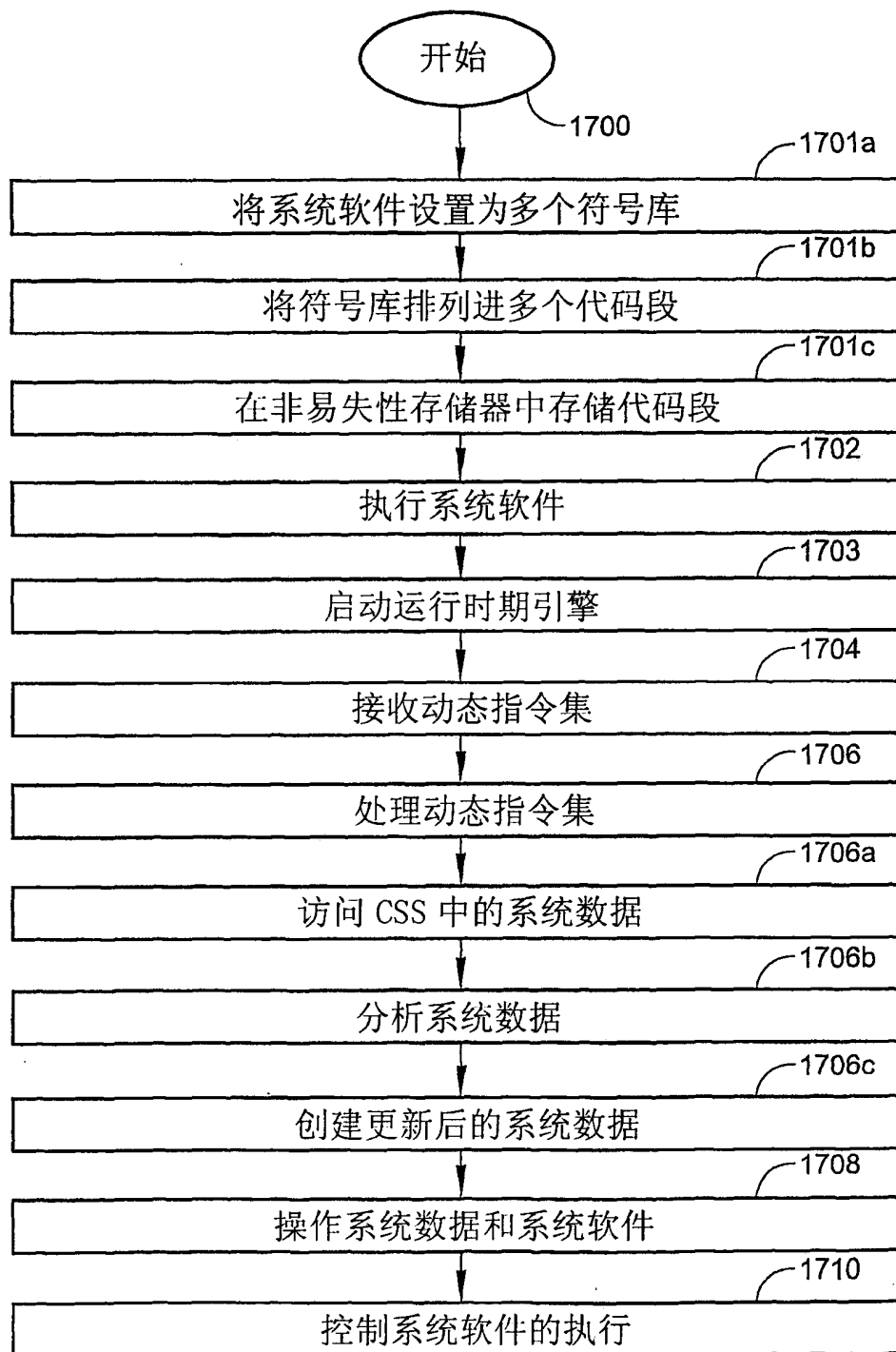


图 17

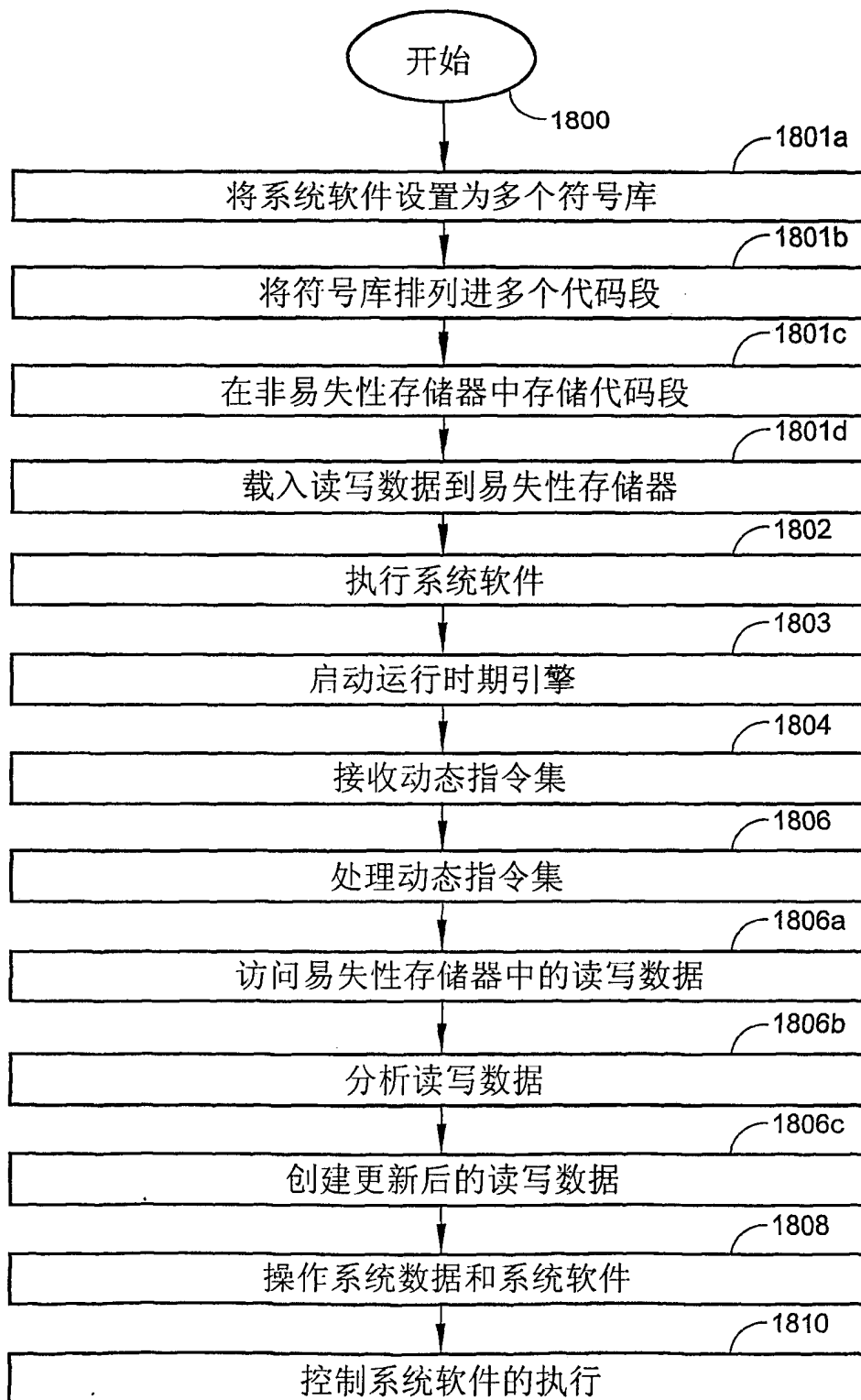


图 18

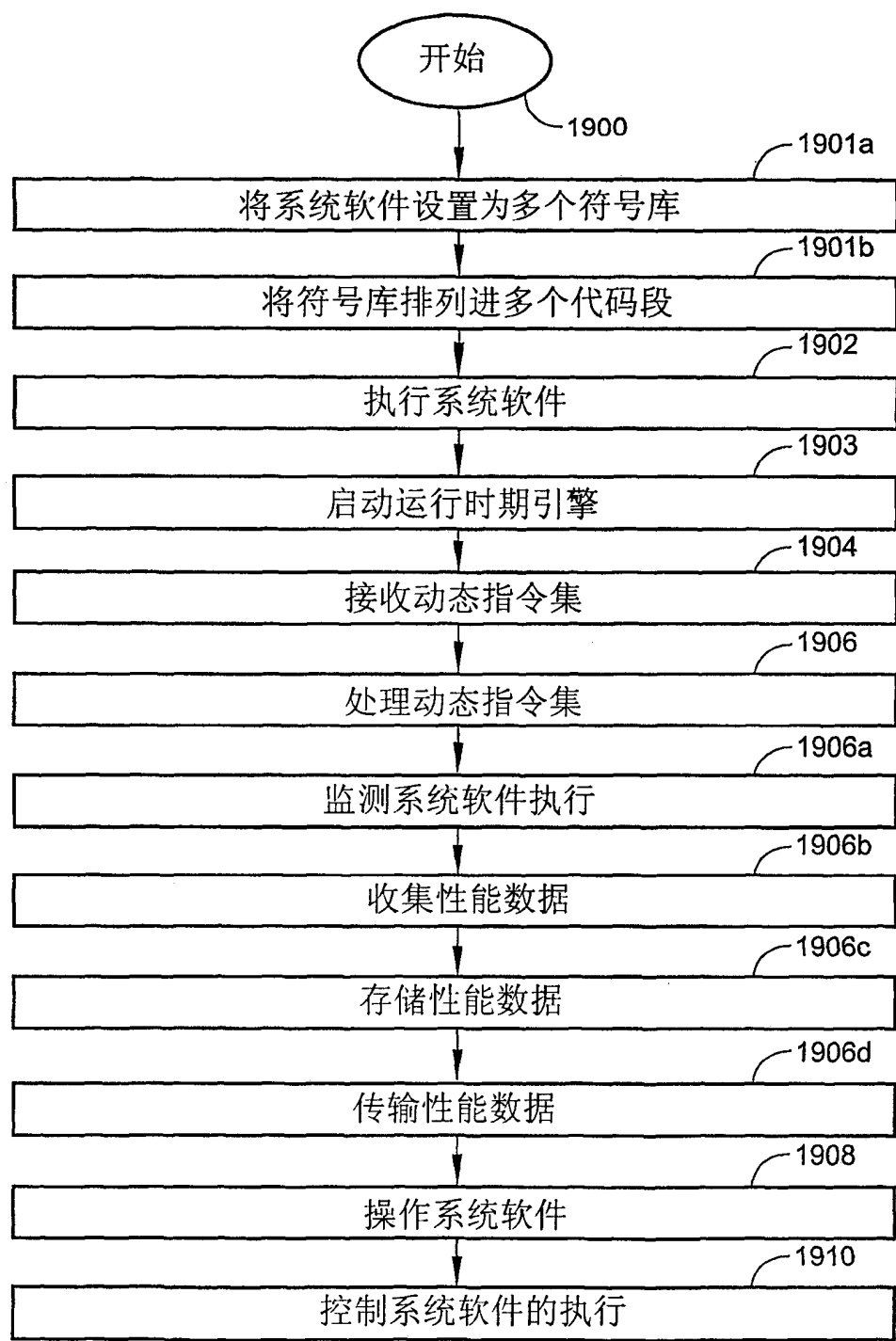


图 19

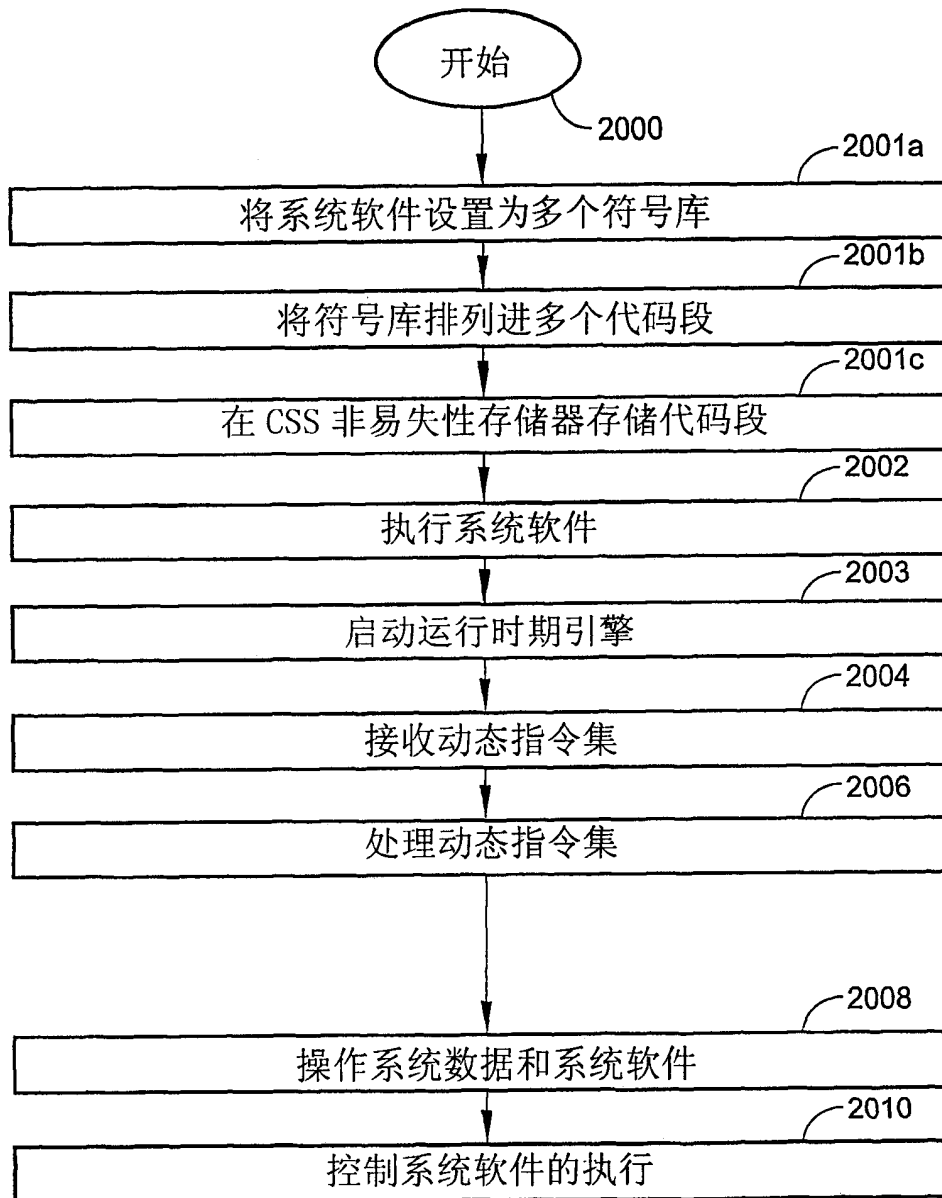


图 20

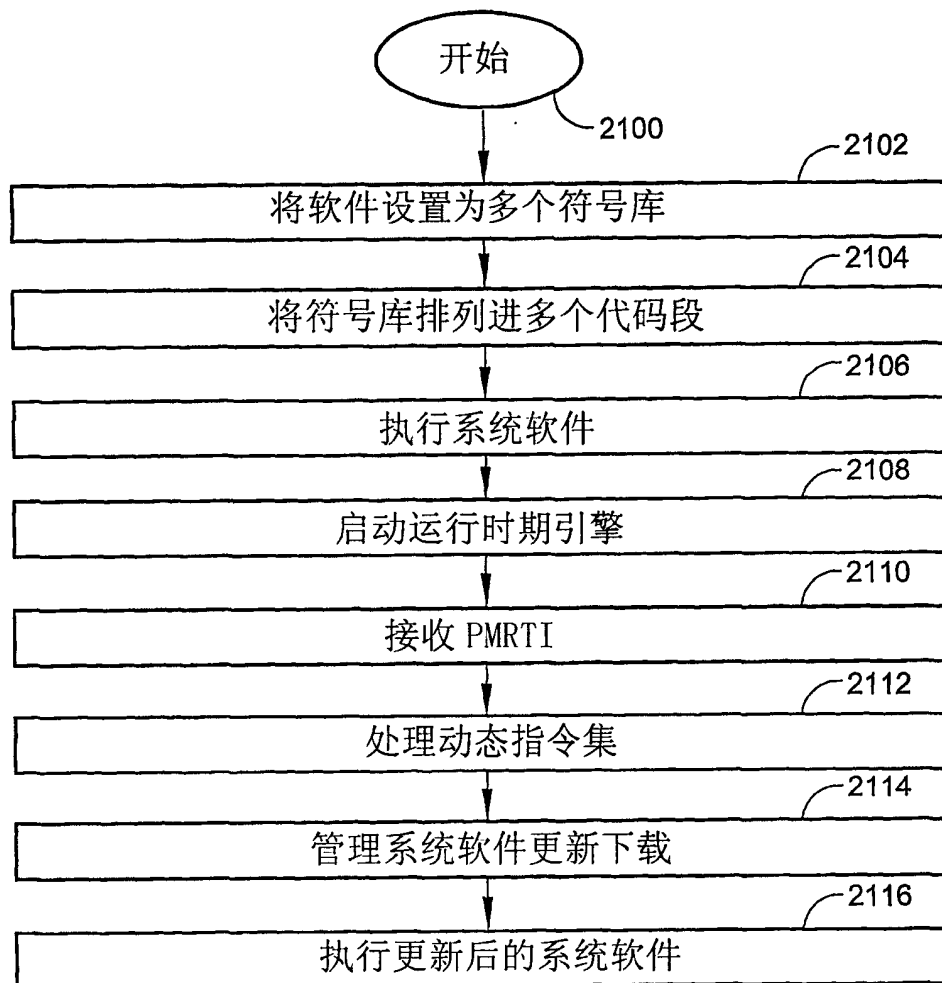


图 21



图 22

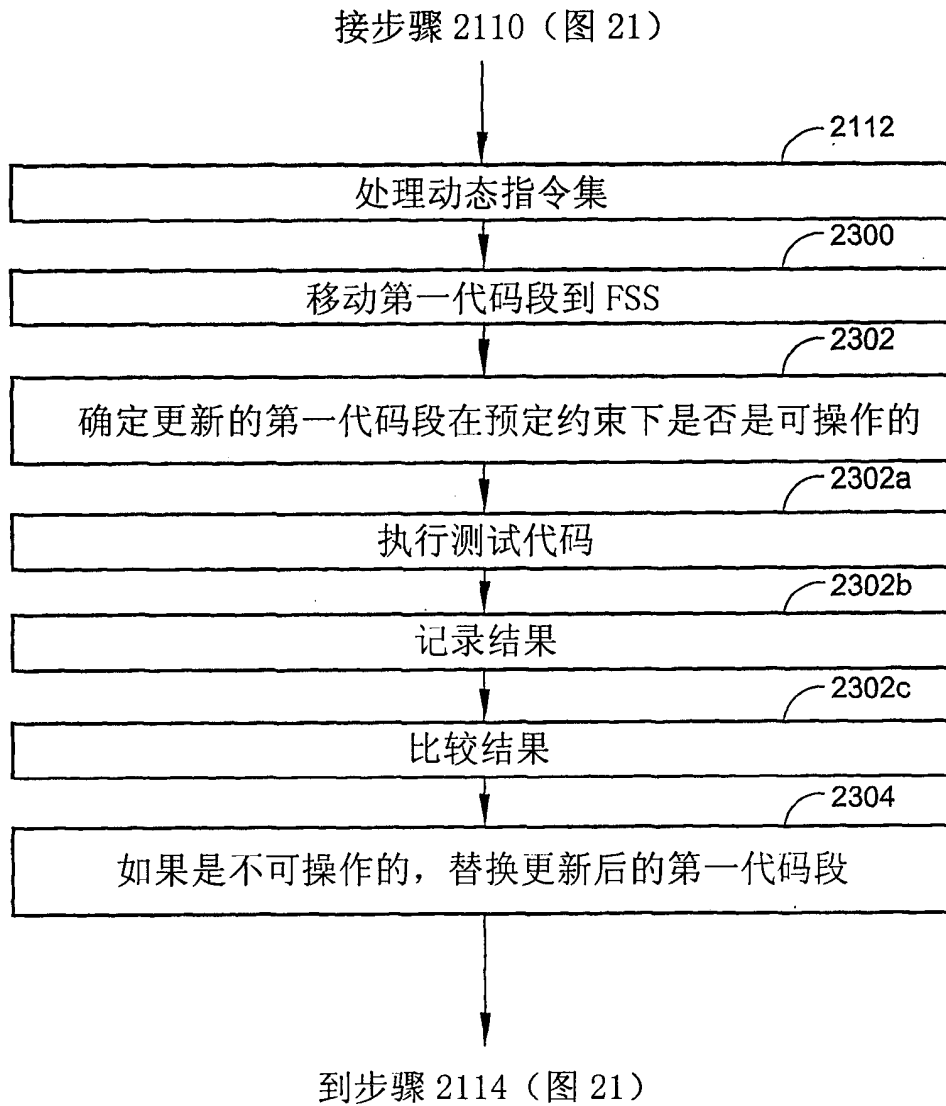


图 23

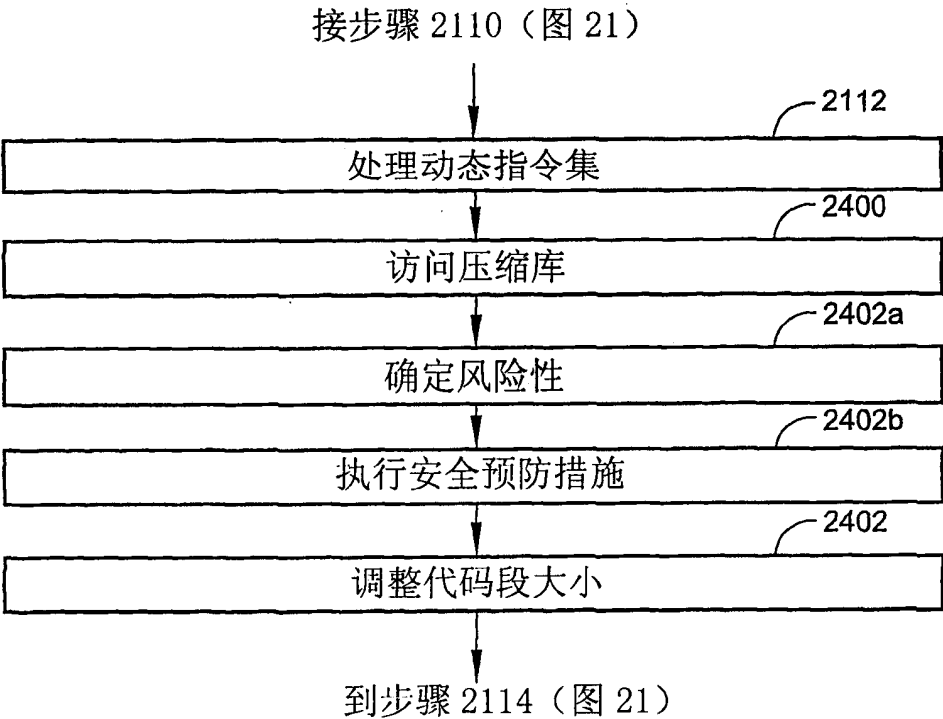


图 24

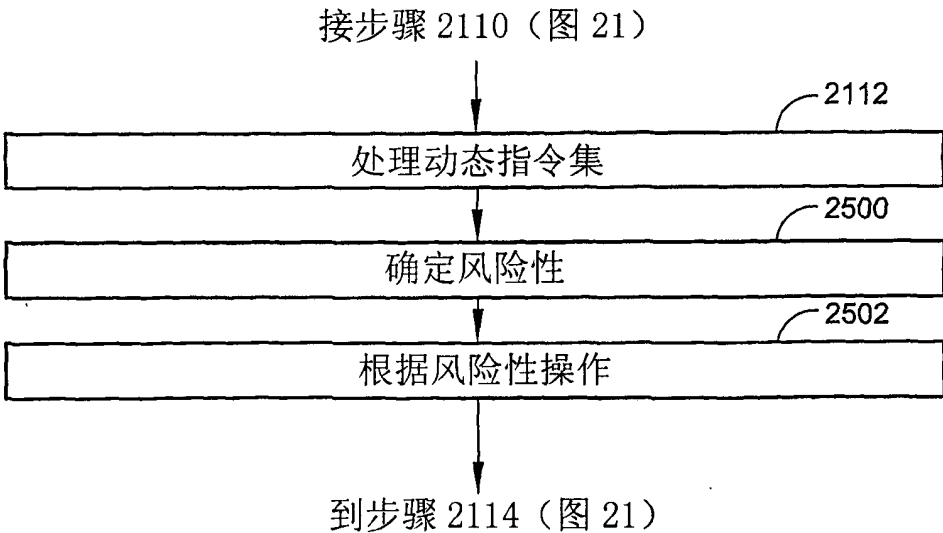


图 25