

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/45 (2006.01)

G06F 9/44 (2006.01)

H04L 12/56 (2006.01)



[12] 发明专利说明书

专利号 ZL 200480032671.0

[45] 授权公告日 2009年8月5日

[11] 授权公告号 CN 100524219C

[22] 申请日 2004.11.16

[21] 申请号 200480032671.0

[30] 优先权

[32] 2003.11.25 [33] US [31] 10/723,684

[86] 国际申请 PCT/US2004/038361 2004.11.16

[87] 国际公布 WO2005/055061 英 2005.6.16

[85] 进入国家阶段日期 2006.5.8

[73] 专利权人 思科技术公司

地址 美国加利福尼亚州

[72] 发明人 唐纳德·班克斯 普拉韦恩·乔希
塞奥克·安东尼·科赫 约翰·韦德

[56] 参考文献

US6014669A 2000.1.11

US6636873B1 2003.10.21

CN1035374A 1989.9.6

EP1347373A2 2003.9.24

Compilers: Principles, Techniques, and Tools.

ALFRED V. AHO, RAVI SETHI, JEFFREY D. ULLMAN, 1,463, ADDISON. WESLEY PUBLISHING COMPANY. 1986

审查员 纵浩

[74] 专利代理机构 北京东方亿思知识产权代理有限公司

代理人 王怡

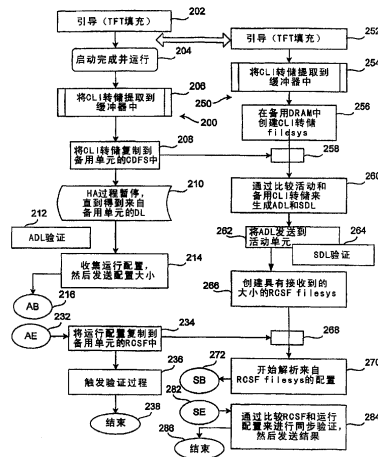
权利要求书 5 页 说明书 14 页 附图 6 页

[54] 发明名称

执行不同版本软件的冗余处理器的配置同步的方法和装置

[57] 摘要

提供了用于在活动单元和备用单元具有不同版本的软件镜像时维持配置同步的方法和装置。优选实现方式允许了即使在备用单元经历镜像升级之后系统也能在有状态模式中工作。其他实现方式允许了在活动单元具有比备用单元更高级别版本的镜像之时系统在有状态模式中工作。



1. 一种维持配置同步的方法，该方法包括：
 - a) 动态地确定具有被活动单元的第一镜像所支持但不被备用单元的第二镜像所支持的第一语法的第一组命令；
 - b) 接收来自所述第一组命令中的一个命令；
 - c) 确定来自所述第一组命令中的所述命令是可变换的；
 - d) 将来自所述第一组命令中的所述命令变换成具有被所述备用单元上的所述第二镜像所支持的第二语法的第一经变换命令；以及
 - e) 将包括所述经变换命令的经变换的运行配置复制到要应用的备用单元。
2. 如权利要求 1 所述的方法，其中所述第二语法是较低级别的语法。
3. 如权利要求 1 所述的方法，其中所述变换步骤由所述活动单元执行。
4. 如权利要求 1 所述的方法，还包括：
 - f) 确定所述第一组命令中不能被变换成所述第二语法的第二部分。
5. 如权利要求 4 所述的方法，还包括：
 - g) 确定所述第二部分的命令是否需要被同步到所述备用单元。
6. 如权利要求 1 所述的方法，其中步骤 d) 是在多个部分中执行的，每个部分包括连续的变换级别之间的变换。
7. 如权利要求 1 所述的方法，其中步骤 a) 包括：

确定用于所述第一镜像的第二组可能的命令；
确定用于所述第二镜像的第三组可能的命令；以及
将所述第二组与所述第三组相比较以确定所述第一组。
8. 如权利要求 1 所述的方法，还包括
确定被所述第二镜像支持但不被所述第一镜像支持的命令的备用差异列表。
9. 如权利要求 1 所述的方法，还包括以下步骤：将所述第一经变换命令同步到所述备用单元。

10. 一种维持配置同步的方法，该方法包括：

- a) 动态地确定具有被活动单元的第一镜像所支持并且被备用单元的第二镜像所支持的第一语法的第一组命令；
- b) 接收来自所述第一组命令中的一个命令；
- c) 确定来自所述第一组命令中的所述命令是可变换的；
- d) 将来自所述第一组命令中的所述命令变换成具有被所述第二镜像所支持的不同语法的第一经变换命令；以及
- e) 将包括所述经变换命令的经变换的运行配置复制到要应用的备用单元。

11. 如权利要求 10 所述的方法，其中所述不同语法是较高级别的语法。

12. 如权利要求 10 所述的方法，其中所述变换步骤由所述备用单元执行。

13. 如权利要求 10 所述的方法，其中步骤 d) 是在多个部分中执行的，每个部分包括连续的变换级别之间的变换。

14. 一种维持活动单元和备用单元之间的配置同步的方法，该方法包括：

由所述活动单元接收先前不是所述活动单元上的活动运行配置的一部分的新命令；

确定所述新命令是否具有被所述备用单元的备用镜像所支持的第一语法；

当确定所述第一语法不被所述备用镜像所支持时，确定所述命令是否可变换到被所述备用镜像所支持的第二语法；以及

当确定所述命令可变换到被所述备用镜像所支持的所述第二语法时，将所述新命令变换为经变换的命令。

15. 如权利要求 14 所述的方法，还包括以下步骤：如果确定所述第一语法被所述备用镜像所支持，则将所述命令同步到所述备用单元。

16. 如权利要求 14 所述的方法，还包括以下步骤：当确定所述命令不可变换到被所述备用镜像所支持的所述第二语法时，确定所述命令是否需

要被同步到所述备用单元。

17. 如权利要求 14 所述的方法，还包括以下步骤：将所述经变换的命令同步到所述备用单元。

18. 如权利要求 16 所述的方法，还包括以下步骤：当确定所述命令不可变换到被所述备用镜像所支持的所述第二语法时，调用策略函数。

19. 一种用于维持配置同步的装置，该装置包括：

用于动态地确定具有被活动单元的第一镜像所支持但不被备用单元的第二镜像所支持的第一语法的第一组命令的装置；

用于接收来自所述第一组命令中的一个命令的装置；

用于确定来自所述第一组命令中的所述命令是可变换的装置；

用于将来自所述第一组命令中的所述命令变换成具有被所述备用单元上的所述第二镜像所支持的第二语法的第一经变换命令的装置；以及

用于将包括所述经变换命令的经变换的运行配置复制到要应用的备用单元的装置。

20. 如权利要求 19 所述的装置，其中所述第二语法是较低级别的语法。

21. 如权利要求 19 所述的装置，其中所述变换步骤由所述活动单元执行。

22. 如权利要求 19 所述的装置，还包括用于确定所述第一组命令中不能被变换成所述第二语法的第二部分的装置。

23. 如权利要求 22 所述的装置，还包括用于确定所述第二部分的命令是否需要被同步到所述备用单元的装置。

24. 一种用于维持配置同步的装置，该装置包括：

用于动态地确定具有被活动单元的第一镜像所支持并且被备用单元的第二镜像所支持的第一语法的第一组命令的装置；

用于接收来自所述第一组命令中的一个命令的装置；

用于确定来自所述第一组命令中的所述命令是可变换的装置；

用于将来自所述第一组命令中的所述命令变换成具有被所述第二镜像所支持的不同语法的第一经变换命令的装置；以及

用于将包括所述经变换命令的经变换的运行配置复制到要应用的备用单元的装置。

25. 如权利要求 24 所述的装置，其中所述不同语法是较高级别的语法。

26. 如权利要求 24 所述的装置，其中所述变换步骤由所述备用单元执行。

27. 如权利要求 24 所述的装置，其中将来自所述第一组命令中的所述命令变换成具有被所述第二镜像所支持的不同语法的第一经变换命令的步骤是在多个部分中执行的，每个部分包括连续的变换级别之间的变换。

28. 一种用于维持配置同步的装置，该装置包括：

其上存储有第一镜像的至少一个活动单元，所述至少一个活动单元被配置成创建被所述第一镜像所支持的命令的第一列表；

其上存储有第二镜像的备用单元，所述备用单元被配置成执行以下步骤：

创建被所述第二镜像所支持的命令的第二列表；

确定在所述第一列表上但不在所述第二列表上的命令的活动差异列表；以及

确定在所述第二列表上但不在所述第一列表上的命令的备用差异列表

其中，当该装置确定所述第一镜像所支持的命令不被所述第二镜像所支持时，确定该命令是否可变换到被所述第二镜像所支持的命令；

当确定该命令可变换到被所述第二镜像所支持的命令时，将该命令变换为经变换的命令。

29. 如权利要求 28 所述的装置，其中所述至少一个活动单元还被配置成执行以下步骤：

生成所述第一镜像的运行配置；以及

确定是所述运行配置的一部分并且是所述活动差异列表的一部分的命令的第三列表。

30. 如权利要求 29 所述的装置，其中所述至少一个活动单元还被配置成执行以下步骤：当所述第三列表是空集时，将所述运行配置同步到所述

备用单元。

31. 如权利要求 29 所述的装置，其中所述活动单元还被配置成执行以下步骤：

确定所述命令的第三列表是否是可变换的；

如果所述命令的第三列表是可变换的，则将所述命令的第三列表变换成第二语法；以及

将经变换的命令的第三列表同步到所述备用单元。

32. 如权利要求 28 所述的装置，其中所述装置包括多个活动单元。

33. 如权利要求 28 所述的装置，其中所述备用单元还被配置成将经同步的运行配置的命令变换成较高级别的语法。

执行不同版本软件的冗余处理器的配置同步的方法和装置

技术领域

本发明涉及共享配置要求的冗余设备。更具体而言，本发明涉及维持网络设备的高可用性。

背景技术

在被配置用于使用冗余性的系统内，需要至少两个处理单元，其中至少一个被指定为“活动单元”，而至少另一个被指定为“备用单元”。这里所使用的“单元”是指一个或多个处理器、线路卡等等。备用单元的任务是在活动单元发生故障时接替活动单元，在理想情况下这是没有任何服务中断的情况下进行的。活动单元负责提供服务并管理系统配置。为了提供备用能力，在任何时刻，其配置视图必须与活动单元所拥有的视图保持同步。初始配置以及所有后续的配置变化都必须被同步到备用单元。

一般而言，系统可以被配置处于“冷”、“暖”或“热”备用状态。尽管这些术语未精确地定义，但是象征性的温度增加指示出备用单元的相对较高的准备就绪状态。一般而言，冷备用状态是指硬件冗余性。当在冷备用状态期间发生故障时，网络设备将会重新启动，并且备用单元变成活动。在暖备用状态中，存在硬件冗余性和足够的应用冗余性，从而在切换（switch over）时网络设备不必重新启动。但是，经由网络设备的会话丢失并且需要重新启动。热备用状态是指存在硬件冗余性和足够的应用冗余性，从而使得一旦发生切换，设备不被重新启动，并且经由网络设备的会话经过切换之后仍被保持。在本发明的上下文中，我们描述的是工作在被成为“有状态切换”（SSO）的热备用模式中的系统中的配置同步。

本发明的受让人开发了这样的方法，在这些方法中，活动单元可以执行与备用单元的镜像不同的镜像。除此之外，这些方法还允许了在网络设备在无任何服务中断的情况下工作的同时网络设备的备用单元的软件镜像

被改变。这种改变提供了更新的镜像（即升级）或更旧的镜像（即降级）以在备用单元上执行。在这个改变期间，系统继续在 SSO（热备用）模式中工作。

发明内容

提供了用于在活动单元和备用单元执行不同版本的软件镜像时维持配置同步的方法和装置。优选实现方式允许了即使在备用单元执行与活动单元上执行的镜像不同的镜像的情况下系统也能在热备用状态中工作。

根据本发明的某些方面，提供了维护其中两个镜像不同的网络设备中的配置同步的方法。该方法包括以下步骤：a) 动态地确定具有被活动单元的第一镜像所支持但不被备用单元的第二镜像所支持的第一语法的第一组命令；以及 b) 将第一组命令的至少第一部分变换成具有被备用单元上的第二镜像所支持的第二语法的第一经变换命令。

第二语法可以是较低级别的语法。变换步骤可以由活动单元执行。该方法还可以包括确定第一组命令中不能被变换成第二语法的第二部分以及确定第二部分的命令是否需要被同步到备用单元。

步骤 b) 可以在多个部分中执行，每个部分包括连续的变换级别之间的变换。步骤 a) 可以包括以下步骤：确定用于第一镜像的第二组可能的命令；确定用于第二镜像的第三组可能的命令；以及将第二组与第三组相比较以确定第一组。该方法可以包括确定被第二镜像支持但不被第一镜像支持的命令的备用差异列表。该方法可以包括以下步骤：将第一经变换命令同步到备用单元。

本发明的其他方面还提供了一种维持配置同步的方法。该方法包括以下步骤：a) 动态地确定具有被活动单元的第一镜像所支持并且被备用单元的第二镜像所支持的第一语法的第一组命令；以及 b) 将第一组命令的至少第一部分变换成具有被第二镜像所支持的不同语法的第一经变换命令。不同语法可以是较高级别的语法。变换步骤可以由备用单元执行。步骤 b) 可以在多个部分中执行，每个部分包括连续的变换级别之间的变换。

本发明的其他实现方式提供了一种维持活动单元和备用单元之间的配置同步的方法。该方法包括以下步骤：由活动单元接收先前不是活动单元上的活动运行配置的一部分的新命令；以及确定新命令是否具有被备用单元的备用镜像所支持的第一语法。

该方法可以包括以下步骤：如果确定第一语法被备用镜像所支持，则将命令同步到备用单元。该方法可以包括以下步骤：当确定第一语法不被备用镜像所支持时，确定命令是否可变换到被备用镜像所支持的第二语法。

该方法可以包括以下步骤：当确定命令可变换到被备用镜像所支持的第二语法时，将新命令变换为经变换的命令。该方法可以包括以下步骤：将经变换的命令同步到备用单元。当确定命令不可变换到被备用镜像所支持的第二语法时，该方法还可以包括以下步骤：确定是否允许同步该命令或者是否需要调用策略以确定适当的纠正性动作。

本发明的某些实施例提供了一种包含在机器可读介质中的计算机程序。计算机程序包括用于控制至少一个网络设备执行以下步骤的指令：
a) 动态地确定具有被活动单元的第一镜像所支持但不被备用单元的第二镜像所支持的第一语法的第一组命令；以及将第一组命令的至少第一部分变换成具有被备用单元上的第二镜像所支持的第二语法的第一经变换命令。第二语法可以是较低级别的语法。变换步骤可以由活动单元执行。

该计算机程序还可以包括用于控制至少一个网络设备执行以下步骤的指令：确定第一组命令中不能被变换成第二语法的第二部分。如果命令的第二部分不能被变换，则它们将不会与备用单元同步。它们将会被对照 DCL 来验证，并且如果在其中找到这些命令，则当它们在活动单元上被执行之后，它们将不被同步。如果在 DCL 中未找到命令，则将调用策略并且将会通过策略判决的结果确定进一步的动作。

步骤 b) 可以在多个部分中执行，每个部分包括连续的变换级别之间的变换。步骤 a) 可以包括以下步骤：确定用于第一镜像的第二组可能的命令；确定用于第二镜像的第三组可能的命令；以及将第二组与第三组相比较以确定第一组。

本发明的其他实施例提供了另一种包含在机器可读介质中的计算机程序。该计算机程序包括用于控制至少一个网络设备执行以下步骤的指令：

a) 动态地确定具有被活动单元的第一镜像所支持并且被备用单元的第二镜像所支持的第一语法的第一组命令；以及 b) 将第一组命令的至少第一部分变换成具有被第二镜像所支持的不同语法的第一经变换命令。不同语法可以是较高级别的语法。变换步骤可以由备用单元执行。

步骤 b) 可以在多个部分中执行，每个部分包括连续的变换级别之间的变换。

本发明的其他实施例提供了一种用于维持配置同步的装置。该装置包括：至少一个被配置成用于动态地确定具有被活动单元的第一镜像所支持但不被备用单元的第二镜像所支持的第一语法的第一组命令的设备；以及至少一个被配置成用于将第一组命令的至少第一部分变换成具有被备用单元上的第二镜像所支持的第二语法的第一经变换命令的设备。第二语法可以是较低级别的语法。变换步骤可以由活动单元执行。

该装置还可以包括至少一个被配置成用于确定第一组命令中不能被变换成第二语法的第二部分的设备。该装置还可以包括至少一个被配置成用于确定第二部分的命令是否需要被同步到备用单元的设备。

本发明的其他实施例提供了一种用于维持配置同步的装置。该装置包括：至少一个被配置成用于动态地确定具有被活动单元的第一镜像所支持并且被备用单元的第二镜像所支持的第一语法的第一组命令的设备；以及至少一个被配置成用于将第一组命令的至少第一部分变换成具有被第二镜像所支持的不同语法的第一经变换命令的设备。不同语法可以是较高级别的语法。变换步骤可以由备用单元执行。步骤 b) 可以在多个部分中执行，每个部分包括连续的变换级别之间的变换。

本发明的其他实施例提供了一种用于维持配置同步的装置。该装置包括：其上存储有第一镜像的至少一个活动单元，所述至少一个活动单元被配置成创建被第一镜像所支持的命令的第一列表。该装置还包括其上存储有第二镜像的备用单元，该备用单元被配置成执行以下步骤：创建被第二镜像所支持的命令的第二列表；确定在第一列表上但不在第二列表上的命

令的活动差异列表；以及确定在第二列表上但不在第一列表上的命令的备用差异列表。

所述一个或多个活动单元还可以被配置成执行以下步骤：生成第一镜像的运行配置；以及确定是运行配置的一部分并且是活动差异列表的一部分的命令的第三列表。所述一个或多个活动单元还可以被配置成执行以下步骤：当第三列表是空集时，将运行配置同步到备用单元。

所述一个或多个活动单元还可以被配置成执行以下步骤：将命令的第三列表变换成第二语法；以及将经变换的命令的第三列表同步到备用单元。该装置可以包括多个活动单元。备用单元还可以被配置成将经同步的运行配置的命令变换到较高级别的语法。

附图说明

图 1 是示出具有活动单元和备用单元的网络设备的图。

图 2 是概括成批（bulk）配置同步过程的流程图。

图 3A 和图 3B 是指示成批配置同步过程的细节的流程图。

图 4 是概括逐行配置同步过程的流程图。

图 5 示出可以被配置成实现本发明的某些方面的网络设备的简化版本。

具体实施方式

在下面的描述中阐述了许多具体细节，以便全面理解本发明。但是，没有这些具体细节中的某些或全部也可以实施本发明，这对于本领域的技术人员来说是很明显的。在另外一些例子里，没有对公知过程步骤进行详细的描述，以免不必要地模糊了本发明。

思科系统公司（Cisco Systems, Inc.）已经开发出了允许操作系统镜像的不同版本在网络设备的活动和备用单元上运行的方法和设备。一个这种方法被称为服务中软件升级（“ISSU”），它通过在备用单元加入配置时检查兼容性并且在工作期间在有状态客户端之间进行协商，来完成这一点。ISSU 的某些方面例如在美国专利申请 No. 10/156,563 和 10/156,556 中

有描述，这里通过引用将这些专利申请并入，用于所有目的。虽然本发明的某些优选实现方式涉及 ISSU，但是本发明并不局限于与 ISSU 一起使用。

图 1 示出网络设备 100，其具有机箱 101 内的活动单元 105 和备用单元 110。由于网络设备 100 包括不止一个单元，因此这样配置的单元有时在这里被称为“冗余的”。网络设备 100 还包括多个线路卡 105。活动单元 105 负责为路由器提供服务。

当活动单元 105 发生故障时，备用单元 110 接管。优选地，备用单元保持在“热”备用状态中，从而允许在活动单元 105 发生故障时进行有状态切换。备用单元 110 和活动单元 105 的配置状态必须保持同步，以允许在活动单元 105 发生故障时进行有状态切换。如果，例如，更新版本的操作系统镜像被安装在了备用单元上，则这种同步可能会成问题。进一步使情况复杂化的是，在活动单元处于工作中的同时，活动单元的配置状态可能被修改。

根据本发明的某些优选实现方式，同步过程发生在两个不同的阶段中。首先，在备用单元初始化并向“热备用”状态发展期间，可以触发成批配置同步或“bulk config sync”过程，在这个过程期间，运行配置与活动单元同步。这例如可以通过将运行配置复制到备用单元中的指定 RAM 位置来实现。

根据优选实现方式，通过遍历所有静态定义的解析节点并将“命令行接口（“CLI”）转储（dump）”的结果存储在缓冲器中，来收集单元中可以实现的所有可能的命令。于是输出缓冲器反映了单元上的镜像的当前解析器链的瞬象。通过比较备用单元和活动单元的 CLI 转储结果，确定（优选通过备用单元）哪些命令在活动单元上可用但在备用单元上不可用，以及反之。正如下文中更详细描述，一个比较用在活动单元运行更高级别版本的命令时。这个比较是确定哪些命令在被同步到备用单元之前必须被变换的基础。这些命令在同步之前被变换成较低级别的语法。其他命令被同步到备用单元，然后可以被变换成较高级别的语法。当备用单元运行更高级别版本的命令时，会发生这种情况。

在完成成批配置同步之后，在网络设备处于工作中的同时，活动单元的配置状态将会经常改变。因此，有必要经由进行中的逐行同步过程来使备用单元维持在“热”备用状态。

图 2、图 3A 和图 3B 是示出根据本发明的某些实现方式的成批配置同步（“bulk config sync”）过程的流程图。在成批配置同步期间，备用单元将会接收活动单元的运行配置文件，该文件将被用于配置备用单元。图 2 的左侧部分指示成批配置同步期间活动单元的功能 200，右侧部分指示备用单元的功能 250。

在步骤 202 和 252 中，活动单元和备用单元初始化。活动单元进行到“活动”状态，而同时备用单元在其初始化期间被暂停，直到活动单元达到活动状态，从而，活动单元启动完成并运行（步骤 204），同时备用单元在初始化期间被停止，等待活动单元通知它继续进行。在初始化过程期间，通过遍历所有静态定义的解析节点来收集在每个单元中可以实现的所有的配置命令，从而创建该单元所支持的所有可能的 CLI 命令的列表，在这里这被称为“CLI 转储”。

所有模式（除了“exec”模式之外）下的所有命令都在一个模式即“全局配置模式”下被收集，这是因为各个配置命令在整个 CLI 上都是唯一的。在这个 CLI 收集过程中，活动单元中创建的二元树被填充以解析链遍历期间生成的所有命令，以去除任何冗余条目。然后，所有命令都被转储到缓冲器中，并被写入物理上位于备用单元中的 DRAM 占位符中，以便活动 CLI 转储可以被备用单元所读取。

同时，通过经历相同的过程，在备用单元中也创建二元树。通过对相同条目的备用二元树搜索，遍历来自活动 CLI 转储的每行命令。如果未找到命令，则该命令被输入到“差异（Diff）列表”中，该列表将被传送给活动单元。

优选实现方式还包括在配置同步初始化期间由活动单元和备用单元创建变换函数表（“TFT”）。根据某些这样的实现方式，TFT 是一个跳跃列表，其被可编程地填充以用于在解析链被初始化时通过每个组件对命令进行升级或降级的函数指针。每个预先编程的函数可以对一个命令进行升

级或降级。调用者判定是否调用函数，并且这些函数天生就会按其定义进行升级或降级。因此，在这个示例性实现方式的步骤 202 和步骤 252 期间，活动单元和备用单元初始化解析链，并以用于升级和降级函数的函数指针来填充其各自的 TFT。

在步骤 206 和步骤 254 的某些实现方式中，两个单元都将其各自的 CLI 转储输出提取到单独的缓冲器中。在某些实现方式中，备用 CLI 转储保持二元树格式。活动单元还向备用单元发送指示活动单元的 CLI 转储输出的大小的信息。在步骤 256 中，备用单元在可由备用单元访问的存储器中（例如在备用单元的 DRAM 区域中）创建一个文件，该文件在这里被称为 CLI 转储文件系统或“fileSYS”（“CDFS”）。在步骤 208 和步骤 258 中，活动单元将其 CLI 转储复制到备用单元的 CDFS 中。然后活动单元暂停 HA 初始化过程，直到接收到来自备用单元的更多信息为止。

在步骤 260 中，备用单元通过将 CDFS 的内容与其自己的 CLI 转储输出相比较，来比较来自活动单元和备用单元的 CLI 转储，在本示例中其自己的 CLI 转储输出是以二元树格式存储的。在另一种实施例中，该计算是由网络设备的另一个部分执行的。在该步骤期间，备用单元确定在活动单元上可用但在备用单元上不可用的命令的列表。这个列表被称为活动差异列表（“ADL”）。ADL 被用于识别如果不修改就直接同步则无法在备用单元上成功执行的命令：或者 ADL 上的命令必须在同步之前被转换成可在备用单元上执行的格式，或者这些命令根本无法同步。

某些命令可能在备用镜像中被删除，但仍被活动镜像所支持。某些命令可能是在活动单元上新创建的，但尚未被备用单元所支持。这组命令不能被变换以使其起作用，从而它们不能被同步到备用单元。根据优选实现方式，不关心列表（“DCL”）被填充以只在活动单元中可用而不需要被同步到备用单元的命令条目。

在步骤 260 中，备用单元还确定“SDL”（备用差异列表），该列表是在备用单元上可用但在活动单元上不可用的命令的列表。SDL 被用于验证为备用单元上不是备用单元的 DCL 的一部分并且不是活动单元的 CDFS 的一部分的那些命令注册了适当的变换函数。如果这些命令的用于对命令

进行升级/降级的变换函数未被注册，则加载在活动单元和备用单元上的两个镜像被声明为不兼容（例如 ISSU 不兼容）。工作模式被降级到无状态冗余模式。

在步骤 262 中，备用单元将 ADL 发送到活动单元，活动单元在步骤 212 中验证 ADL。在步骤 264 中，备用单元验证 SDL。作为该验证过程的一部分，使每个 DL 的每个条目经过 TFT，以确定是否注册了匹配的变换函数。

在步骤 214 中，活动单元收集其运行配置，并向备用单元发送指示经变换的运行配置的文件大小的信息。在步骤 266 中，备用单元部分基于文件大小信息创建一个文件（在这里是“RCSF filesys”），活动单元的运行配置将被复制到该文件中。

在步骤 216 至步骤 232 中，如果必要的话，运行配置的命令被变换。如果在 ADL 中找到运行配置的命令的匹配条目，则将运行配置的那一行馈送到对命令版本进行降级的至少一个变换函数。重复这个变换过程，直到或者（1）发现来自变换的输出命令是可在备用单元上执行的命令，或者（2）系统确定命令不兼容。如果命令不兼容，则系统优选地将工作模式改变成无状态模式，或将系统置于非冗余模式中，并重启动备用单元。这个过程在图 3A 中示出。

在步骤 318 中，备用单元将收集到的运行配置的每一行馈送到成批配置同步变换引擎中。在步骤 320 中，将运动配置的每个命令与 ADL 相比较，以确定是否存在匹配。如果没有匹配，则意味着命令可以被备用单元使用。因此，命令被写入到输出缓冲器（步骤 330），并且过程继续到步骤 232。

如果在步骤 320 中找到匹配，则意味着命令不能被备用单元使用。过程继续到步骤 322，在该步骤中，活动单元确定命令是否可变换。如果是的话，则在步骤 324 中命令被降级。

优选地，命令每次被变换一个级别，而不是一次就被直接转换到目标格式。例如，假设 `cmd_x` 被升级为 `cmd_y`，然后被升级为跨操作系统镜像的多个版本的 `cmd_z`。在升级场景中，`cmd_x` 首先被转换为 `cmd_y`，然后

被转换为 `cmd_z`; `cmd_x` 不被直接变换为 `cmd_z`。从而，活动单元上执行的所有配置命令都以相对于其在活动单元上的情况被降级或保持不变的方式被同步到备用单元。

当命令被降级时，经变换的命令的输出可能不同于原始输入的命令的输出。为了针对解决这一潜在问题，针对这种经变换的命令，支持解析器返回代码（“PRC”），以便可以验证在备用单元上执行的结果。降级情形发生在较高（较后）版本的镜像运行在活动单元上时，因而某些在活动单元上可配置的命令在备用单元上则不是有效的。因此，这些命令或者必须被变换成在备用单元上可配置的较低的版本，或者这些命令不应当被同步。

如果命令不可变换，则在步骤 326 中备用单元确定命令是否在其“不关心列表”上。如果是的话，则命令不需要被同步（因此不被同步）。

如果命令不在 DCL 上，则调用策略处理器（步骤 328）。这种情形不会意外地发生，这是因为任何可能导致这种差错的情况都会在镜像被装运之前被得知和公布。策略处理器例如是调用一组规则的软件。如果策略处理器确定命令较重要但不能被同步，则在某些实现方式中，将丢弃命令，声明“丧失同步”状态并报告差错，如步骤 329 中所示。命令不被写入到输出缓冲器，也不被用于同步备用单元。备用单元将会被重新启动，系统将会返回非冗余配置中或无状态工作模式中。然后过程继续到步骤 328。

但是，如果图 3A 中所示的过程正常地完成，则方法以图 2 的步骤 234 和 268 继续。在这里，活动单元将其经变换的运行配置复制到备用单元所准备的文件中（在这里是“CDFS filesys”），该文件可能位于备用单元的 DRAM 中。

在升级场景中，在备用单元上支持较高版本的命令，因此升级转换发生在备用单元上下文中。即，备用单元支持比从活动单元同步的镜像的版本更高的命令。如果备用单元在运行更新的镜像版本，则每个命令行通过 TFT 被处理，以查明它是否能被进一步升级。然后在备用单元上执行该命令。

一个这种命令升级过程由图 3B 的流程图所示。在步骤 374 中，备用

单元读取从活动单元接收到的运行配置文件的每一行（在这里是通过读取 RCSF 进行的），并将每一行发送到升级变换引擎。在步骤 376 中，备用单元确定命令是否可变换（在本示例中是确定命令是否可被升级）。如果是的话，则在步骤 378 中命令被变换（在这里是被升级）。重复步骤 376 和步骤 378，直到在备用单元的 TFT 中没有升级变换函数可用。此时，确定命令不能再被进一步变换，并且在步骤 380 中处理命令行。

在备用单元处理了 RCSF 的所有命令行之之后，在步骤 284 中触发验证过程（见图 2），以确定是否所有经解析的命令都被正确地配置。步骤 284 确保了不存在由于在活动单元和备用单元上运行不同版本的镜像而引起的配置失配。

逐行配置同步

在成批配置同步完成并且网络设备正常工作之后，只要在活动单元上发生配置变化，运行在备用单元上的镜像就必须被升级，以使其保持与活动单元的镜像同步。这个过程在图 4 中示出。

在步骤 405 中，新的命令从活动单元进入。在步骤 410 中，确定该命令是否在 ADL 中。如果不是的话，则在步骤 415 中，新命令被同步。如果是的话，则确定命令是否可变换（步骤 420）。如果命令是可变换的，则在步骤 425 中它被变换。然后过程返回步骤 410，在该步骤中对经变换的命令进行估计，以确定它是否在 ADL 上。

如果命令不可变换，则将其与 DCL 相比较（步骤 430）。如果命令在 DCL 上，则命令不被同步（步骤 435）。如果命令不在 DCL 上，则策略处理器被调用（步骤 440），并且命令不被同步。根据策略处理器的判决，会随之发生其他后果。例如，如步骤 445 所示，可以仅仅是丢弃命令并发出警告。或者，如步骤 450 所示，可以对备用状态进行降级。

图 5 示出可以被配置成实现本发明的某些方法的网络设备的示例。网络设备 560 包括活动中央处理单元（CPU）562、备用 CPU 592、接口 568 和总线 567（例如 PCI 总线）。一般而言，接口 568 包括适用于与适当介质通信的端口 569。在某些实施例中，接口 568 中的一个或多个包括至少一个独立处理器 574，并且在某些情况下包括易失性 RAM。独立处理器

574 例如可以是 ASIC 或任何其他适当的处理器。

根据某些这样的实施例，这些独立处理器 574 执行这里所描述的逻辑的至少某些功能。在某些实施例中，接口 568 中的一个或多个控制诸如介质控制和管理这样的通信密集型任务。通过为通信密集型任务提供单独的处理器，接口 568 允许了活动 CPU 562 高效地执行诸如路由选择计算、网络诊断、安全性功能等其他功能。

接口 568 一般是作为接口卡（有时称为“线路卡”）提供的。一般而言，接口 568 控制经由网络的数据分组的发送和接收，并且有时支持与网络设备 560 一起使用的其他外围设备。可以提供的接口有 PC 接口、以太网接口、帧中继接口、线缆接口、DSL 接口、令牌环接口等等。此外，可以提供各种甚高速接口，例如快速以太网接口、千兆比特以太网接口、ATM 接口、HSSI 接口、POS 接口、FDDI 接口、ASI 接口、DHEI 接口等等。

在本发明的某些实现方式中，当在适当的软件或固件的控制之下动作时，活动 CPU 562 和备用 CPU 592 可以负责实现与本申请中描述的所需网络设备的功能相关联的特定功能。根据某些实施例，活动 CPU 562 和备用 CPU 592 在包括操作系统（例如由思科系统公司开发的专用操作系统 Cisco IOS，等等）和任何适当的应用软件在内的软件的控制下实现所有这些功能。

CPU 562 可以包括一个或多个处理器 563，例如来自 Motorola 微处理器家族或 MIPS 微处理器家族的处理器。在另一个实施例中，处理器 563 是专门设计的用于控制网络设备 560 的操作的硬件。在特定实施例中，存储器 561（例如非易失性 RAM 和/或 ROM）也构成 CPU 562 的一部分。但是，有许多不同的将存储器耦合到系统的方式。存储器块 561 可以用于多种用途，例如缓存和/或存储数据、编程指令等等。类似地，备用 CPU 592 可以包括一个或多个处理器 593 和存储器块 591。

不论网络设备的配置如何，它都可以采用一个或多个存储器或存储器模块（例如存储器块 565），这些存储器或存储器模块被配置成存储数据、用于通用网络操作的程序指令以及/或者其他与这里所描述的技术功能

相关的信息。程序指令例如可以控制操作系统和/或一个或多个应用程序的操作。

由于这种信息和程序指令可以被用来实现这里所描述的系统/方法，因此本发明涉及包括用于执行这里所描述的各种操作的程序指令、状态信息等机器可读介质。机器可读介质的示例包括但不限于：磁介质，例如硬盘、软盘和磁带；光介质，例如 CD-ROM 盘；磁光介质；以及被专门配置成存储和执行程序指令的硬件设备，例如只读存储器设备（ROM）和随机访问存储器（RAM）。本发明也可以实现在传播经过适当介质的载波中，例如无线电波、光线路、电线路等等。程序指令的示例既包括例如由编译器产生的机器代码，也包括包含可由计算机利用解释器执行的更高级别的代码的文件。

虽然图 5 所示的系统示出了本发明的一个特定网络设备，但是它决不是唯一的可在其上实现本发明的网络设备体系结构。例如，常常使用具有处理通信以及路由选择计算等的单个处理器的体系结构。此外，其他类型接口和介质也可以与网络设备一起使用。接口/线路卡之间的通信路径可以是基于总线的（如图 5 所示），或者是基于交换结构的（例如纵横式交换机）。

其他实施例

一般而言，本发明的技术可以实现在软件、硬件、固件和/或微代码上。例如，它们可以实现在操作系统内核中、单独的用户进程中、绑定到网络应用中的库程序包中、专用构造的机器上或网络接口卡上。在所有情况下，这些处理单元都必须按冗余关系成对。在本发明的特定实施例中，本发明的技术是在诸如操作系统这样的软件中或运行在操作系统上的应用程序中实现的。

本发明的技术的软件或软件/硬件混合实现方式可以实现在由存储在存储器中的计算机程序选择性地激活或重配置的通用可编程机器上。这种可编程机器可以是设计成处理网络流量的网络设备，例如以上参考图 5 所描述的网络设备。在另一种实施例中，本发明的技术可以实现在诸如个人计算机或工作站之类的通用网络主机上。此外，本发明可以至少部分地实

现在网络设备或通用计算设备的卡（例如接口卡）上。

虽然这里示出和描述了本发明的示例性实施例和应用，但是许多保持在本发明的概念、范围和精神内的变化和修改都是可能实现的，并且本领域的普通技术人员在熟读本申请之后将会清楚看出这些变化。

例如，本发明的方法可以以 1:1 或 1:N 处理单元关系来实现。换言之，可以存在具有单个备用单元的单个活动单元或者多个活动单元。

因此，当前的实施例应当被认为是示例性的而非限制性的，并且本发明并不局限于这里给出的细节，而是可以在所附权利要求范围和等同物内被修改。

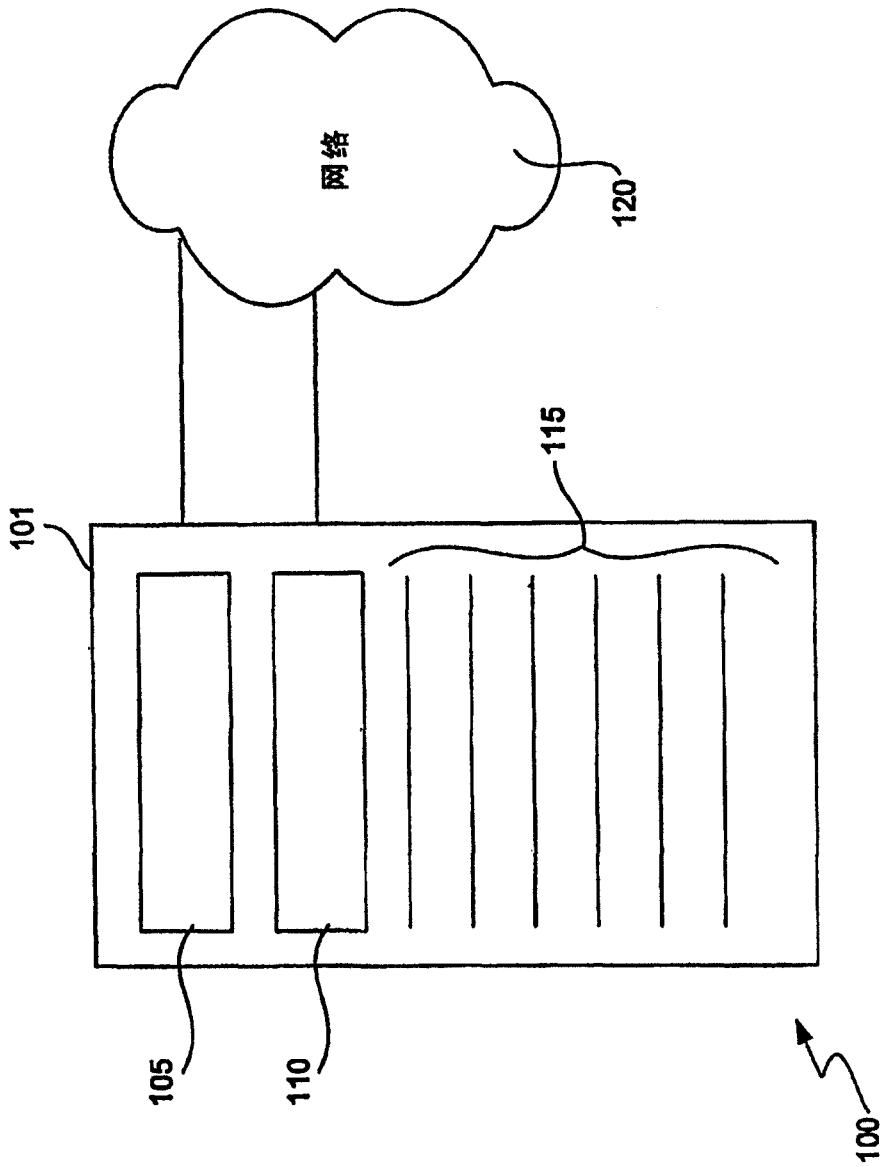


图1

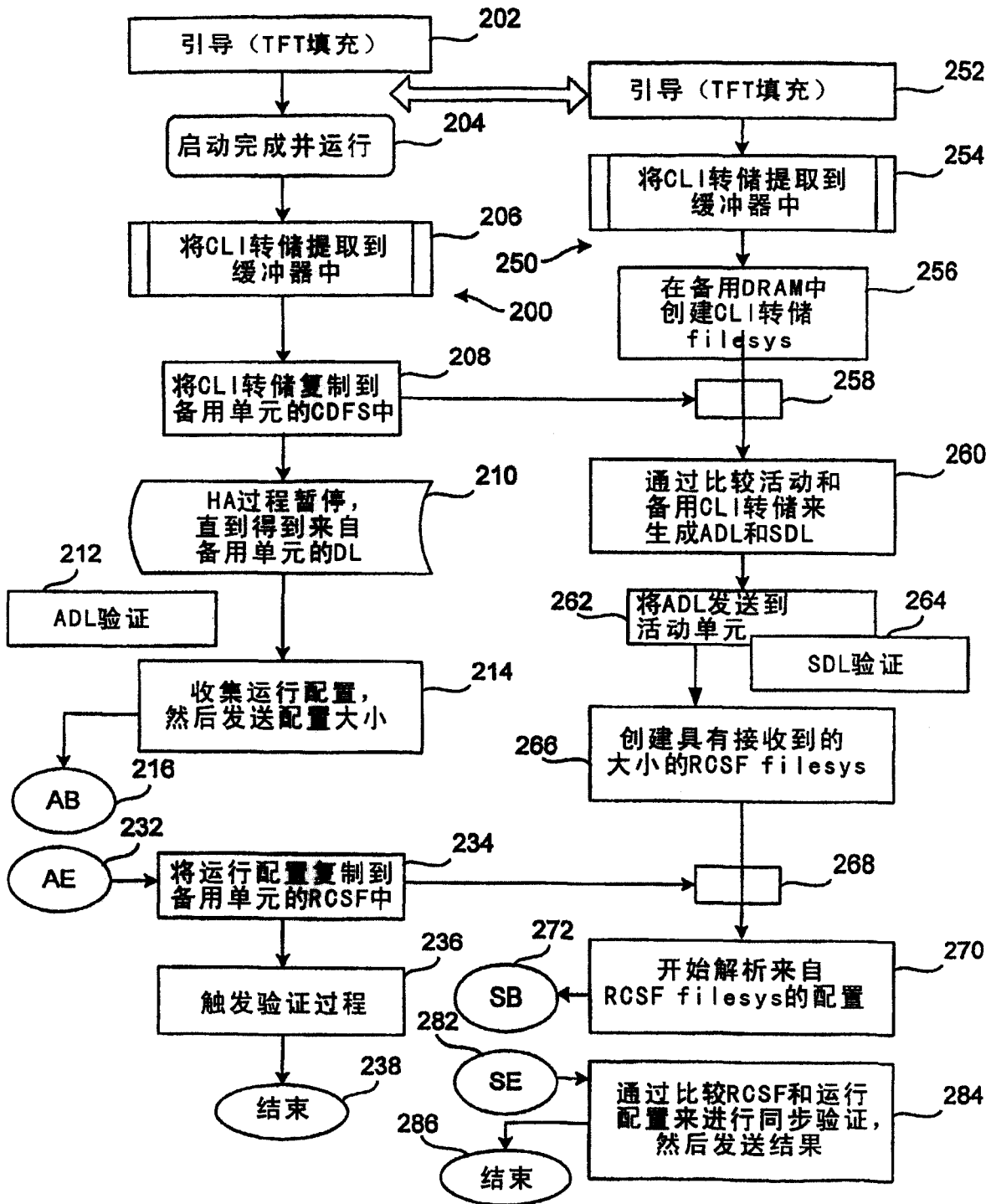


图2

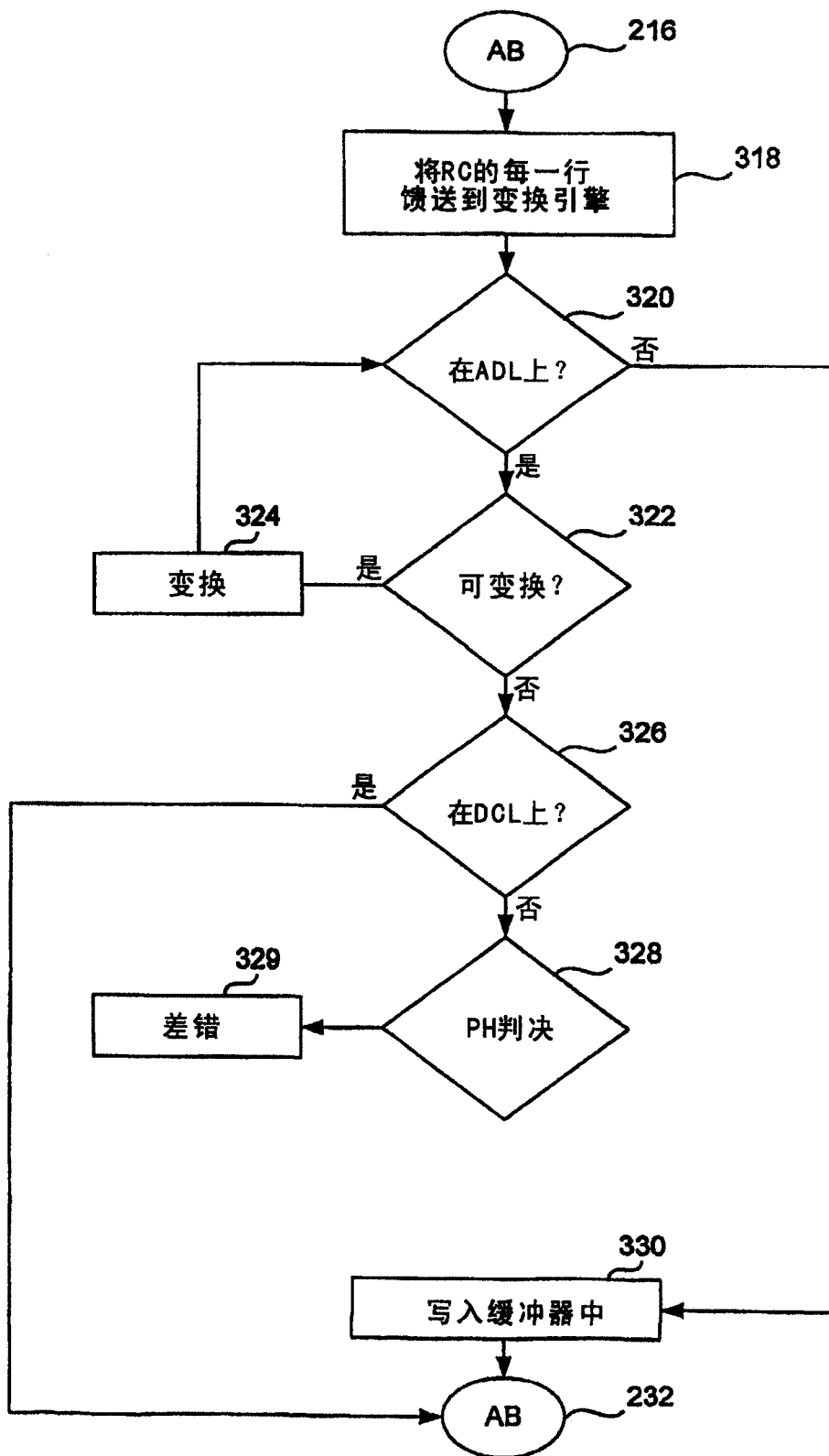


图3A

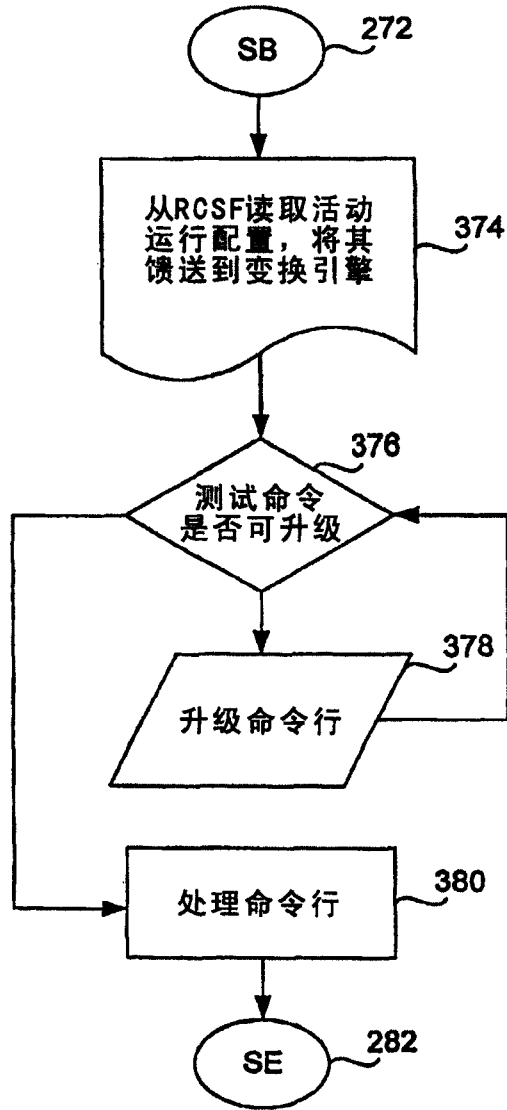


图3B

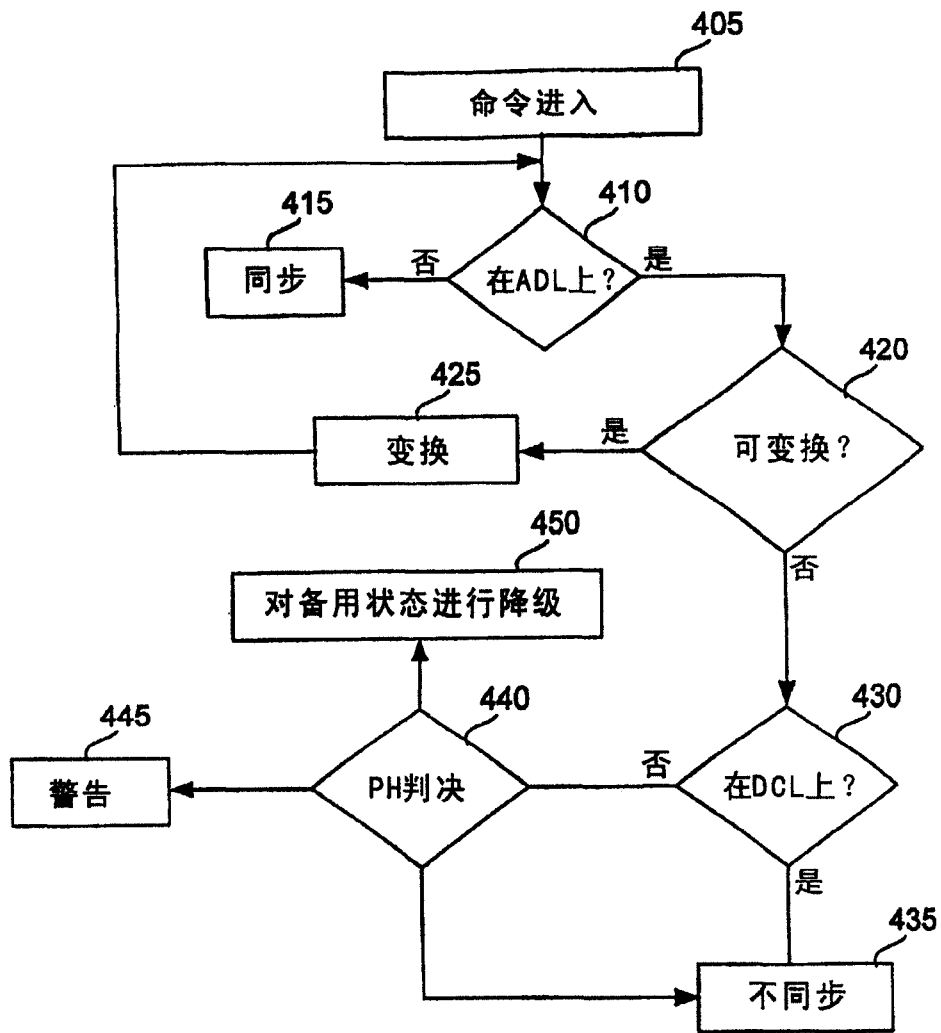


图4

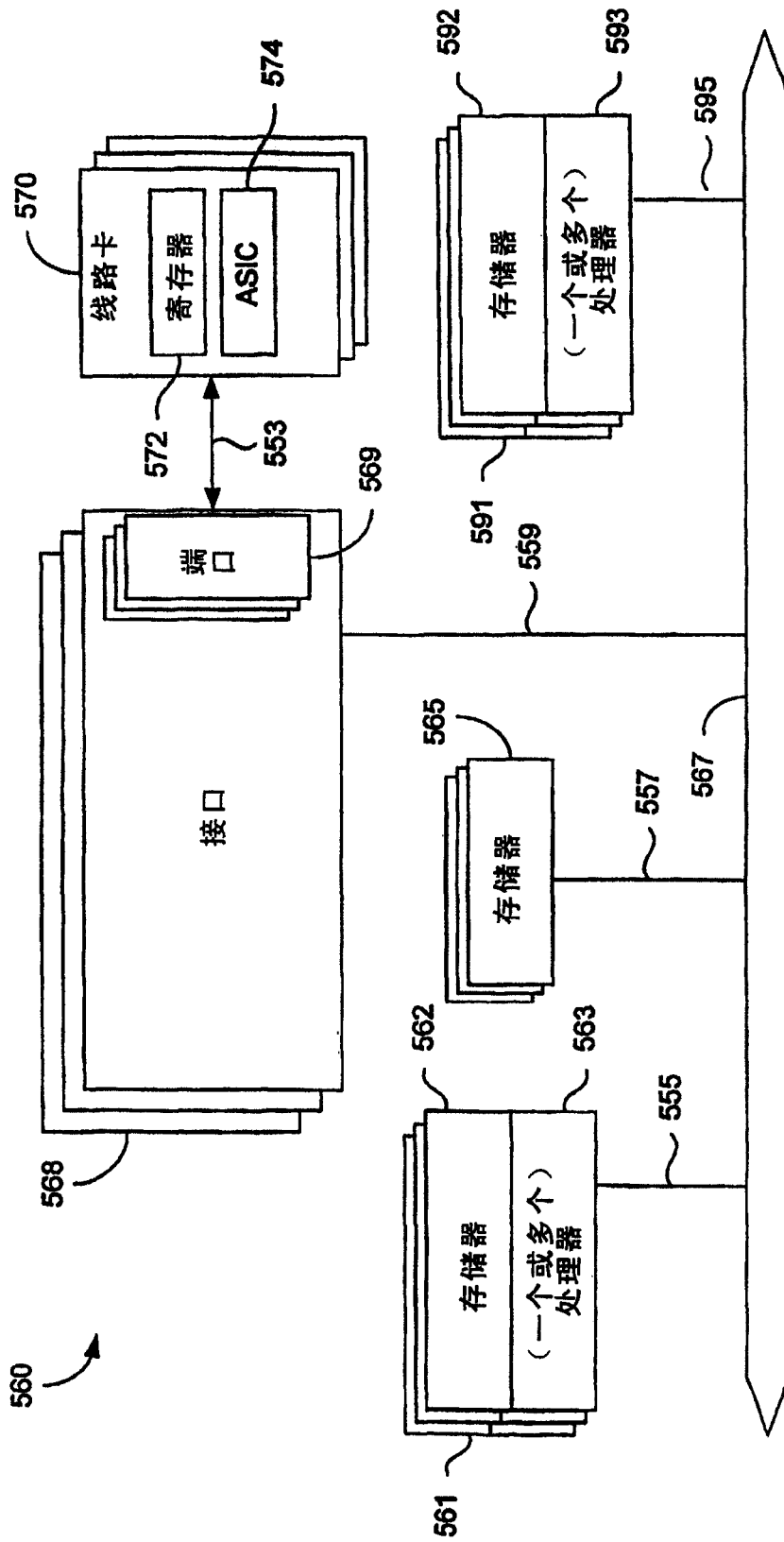


图5