



(12) 发明专利

(10) 授权公告号 CN 101861560 B

(45) 授权公告日 2015. 11. 25

(21) 申请号 200880116618. 7

(22) 申请日 2008. 09. 19

(30) 优先权数据

60/973, 691 2007. 09. 19 US

(85) PCT国际申请进入国家阶段日

2010. 05. 18

(86) PCT国际申请的申请数据

PCT/US2008/077007 2008. 09. 19

(87) PCT国际申请的公布数据

W02009/039365 EN 2009. 03. 26

(73) 专利权人 泰普索福特科技公司

地址 美国加利福尼亚

(72) 发明人 R·J·玛尔斯登

(74) 专利代理机构 北京市金杜律师事务所

11256

代理人 王茂华

(51) Int. Cl.

G06F 3/041(2006. 01)

(56) 对比文件

CN 101036105 A , 2007. 09. 12, 说明书第 7

页第 15-20 行, 第 10 页第 14-26 行、图 1, 4.

CN 1075222 A , 1993. 08. 11, 说明书第 5 页
第 16-19 行.

CN 1864125 A , 2006. 11. 15, 说明书第 6 页
第 1-8 行、图 1.

CN 1900883 A , 2007. 01. 24, 全文.

CN 2662340 Y , 2004. 12. 08, 说明书第 2 页
第 8 行至第 3 页第 6 行、图 1.

WO 2006/133018 A2 , 2006. 12. 14, 说明书第
9 页倒数第 10 行至第 15 页第 11 行、图 3A-4A.

审查员 梁静静

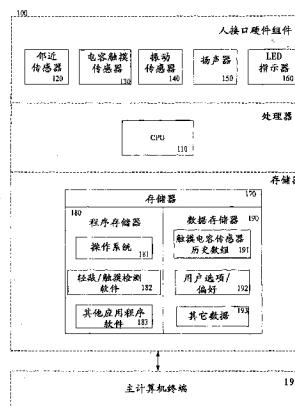
权利要求书2页 说明书10页 附图16页

(54) 发明名称

可清洁的触摸及轻敲敏感表面

(57) 摘要

一种具有触摸电容和振动传感器的触摸敏感表面。这种表面允许用户把其手指放在键上, 并如同他们在常规键盘上那样打字。当用户把其手指放在键上时, 触摸电容传感器(每个键一个)向处理器报告被触摸的每个键的信号强度水平, 但是处理器不发出击键, 直到检测到对应的“轻敲”(即振动)。当检测到轻敲时, 处理器参考在发生轻敲的时刻之前、期间、和/或紧跟其后的触摸电容传感器的状态。



1. 一种检测固态平坦触摸敏感表面上的用户输入以确定用户输入的位置的方法,所述方法包括:

基于所述触摸敏感表面中所包括的多个触摸传感器,记录对所述触摸敏感表面的用户触摸;

基于由耦合到所述触摸敏感表面的一个或多个振动传感器感测的轻敲事件,从所述一个或多个振动传感器接收轻敲事件信号,其中所述触摸传感器在物理上和操作上不同于所述振动传感器;

将所述用户触摸与所述轻敲事件信号相关,以区分所述用户将手放在所述触摸敏感表面上和所述用户在所述触摸敏感表面上断言选择;以及

基于所记录的用户触摸,在接收到所述轻敲事件信号后通过参考在发生轻敲的时刻之前、期间或者紧跟其后的触摸传感器的状态来断言一个选择。

2. 如权利要求 1 所述的方法,其中所述表面包括多个与对应的触摸传感器相关联的字母数字键。

3. 如权利要求 2 所述的方法,其中所述触摸敏感表面包括标识每个键的位置的第一触觉标记,其中所述第一触觉标记是所述触摸敏感表面上的浅凹陷。

4. 如权利要求 1 所述的方法,还包括当断言一个选择时发出可听的声音。

5. 如权利要求 1 所述的方法,还包括确定所述触摸传感器其中之一何时已根据滑动运动被触摸,其中被断言的选择与未被确定为已根据滑动运动被触摸的触摸传感器其中之一相关联。

6. 如权利要求 1 所述的方法,还包括:

基于所记录的用户触摸检测擦拭运动;以及
当检测到擦拭运动时中止断言。

7. 如权利要求 2 所述的方法,还包括通过有线或者无线方法与主计算机通信。

8. 如权利要求 1 所述的方法,其中所述表面执行鼠标指点设备的功能。

9. 如权利要求 1 所述的方法,其中所述表面是动态显示计算设备上的触摸屏。

10. 如权利要求 1 所述的方法,其中所述表面是用户可定义的。

11. 如权利要求 1 所述的方法,其中所述触摸传感器或者振动传感器中至少一个的灵敏度可调整。

12. 如权利要求 1 所述的方法,还包括:

感测所述用户何时要使用所述表面;以及
当感测到用户要使用所述表面时,从所述表面的低功耗状态改变到满功率操作状态。

13. 如权利要求 1 所述的方法,还包括:

通过触摸传感器的阵列确定自上次清洁事件以来所记录的用户触摸的数量;以及
如果所记录的用户触摸的被确定的数量超过阈值,则启用警告指示器。

14. 如权利要求 1 所述的方法,还包括:

累积记录与所述触摸敏感表面的用户交互作用的水平,直到达到第一污染阈值为止;
以及

当达到所述第一污染阈值时断言警报,

其中,所述警报是警告灯、可听声音、振动或者发送到主计算机终端的命令中的至少一

个,

其中,用户交互作用的水平包括以下一个或多个:所记录的用户触摸,接收到的轻敲事件,感测的所述表面的移动,以及确定何时用户靠近所述触摸敏感表面。

15. 如权利要求 14 所述的方法,其中累积记录继续到超过第一用户清洁阈值为止,其中所述第一用户清洁阈值基于擦拭的数量和被擦拭的触摸表面面积,该方法还包括当超过所述第一用户清洁阈值时解除断言所述警报。

16. 如权利要求 1 所述的方法,其中通过在键的位置轻敲所述表面、抬起、然后快速接着再次轻敲并保持,所述键被断言为保持按下,其中通过抬离所述键释放被断言为保持按下的键。

17. 如权利要求 2 所述的方法,还包括通过激活以下之一的禁用键组合来中止所述表面的操作:单个键,两个或更多个同时的键激活,以及一系列两个或更多个顺序的键按压。

18. 如权利要求 2 所述的方法,其中用户设置在所述触摸表面上可改变,其中,所述用户设置被显示在多个 LED 指示器上。

19. 如权利要求 1 所述的方法,还包括:

感测所述触摸表面的移动;以及

如果感测的移动高于预定义的阈值,则中止所述表面的操作。

20. 如权利要求 1 所述的方法,还包括:

在所述断言之前,根据预定义的变化阈值来确定所记录的用户触摸是否变化,其中断言选择包括如果根据所述预定义的变化阈值确定用户触摸变化则断言一个键按压。

可清洁的触摸及轻敲敏感表面

[0001] 优先权

[0002] 本申请要求 2007 年 9 月 19 日递交的序列号为 No. 60/973, 691 的美国临时申请的优先权, 其通过引用被包含在本文中。

技术领域

[0003] 本发明涉及一种光滑的触摸敏感表面 (touch-sensitive surface), 其容易清洁, 并且允许用户把他们的手或者手指放在所述表面上而不导致事件致动 (event actuation)。更具体地, 所述触摸表面可以用作用于输入文本和命令的计算机键盘。

背景技术

[0004] 作为用于从人类向机器输入文本和数据的基本方法的现代键盘的起源回溯到 19 世纪的早期打字机。随着计算机的发展, 把用作用于输入文本和数据的基本方法的打字机键盘加以改装是一种自然演变。虽然打字机和随后的计算机键盘上的键的实施方案从机械的演变为电气的, 并且最终演变为电子的, 但是这些键本身的大小、位置和机械性质仍保持大体未改变。

[0005] 在跨很多行业的环境中, 计算机及其伴随的键盘已经变得司空见惯, 这些环境中的很多具有起初在计算机和键盘的设计中未被考虑到的严酷状况。例如, 计算机现在被使用在餐馆的厨房中、制造设施的生产平台上、以及石油钻机上。这些环境是传统键盘因极端的污染状况所致将不会无需清洁地保持运转很长时间的环境。

[0006] 计算机也被用在要求极端清洁的环境中。医院护理站、手术室、检查室、牙科治疗室和诊断设施现在一般都利用了计算机。这些是感染控制极端重要的环境。在实现抗菌环境时, 由于清洁键盘时出现的困难, 键盘变为特别难解决的问题。这被由位于 Hawaii, Honolulu 的 Tripler 陆军医学中心进行的研究所证明, 其发现将近 25% 的在医院中使用的键盘藏有对所有医源性疾病的 95% 负有责任的细菌类型。

[0007] 由于传统键盘上的键的机械性质, 它们包含很多导致裂缝和空腔的活动部分, 在所述裂缝和空腔处, 灰尘和污染物可能聚集。此外, 键盘反复地接触人手——污垢、细菌、病毒和其他感染原的最常见的输送者之一。清洁传统机械键盘带来了困难的挑战。在典型的键盘上, 存在超过 500 个要清洁的单独表面 (算上每一个键的顶部和侧面)。这些表面中的很多表面非常难以接近和有效地清洁, 特别是在不按下键时, 这意味着它们必须在键盘被禁用或者计算机被关闭时被清洁。

[0008] 一些以往的发明已经寻求通过在键盘上使用密封的橡胶键来解决清洁能力的问题。这些键盘由柔软、可曲折的橡胶 (通常是硅橡胶) 制成, 所述橡胶被模塑并密封在电气键触点上方, 提供防潮屏障。这使得例如能够把键盘放置在流动的水下来清洁它。键仍具有机械行程和触感, 这一般被认为有利于快速和有效的打字。

[0009] 虽然这种方法使得能够清洁键盘, 但是仍存在很多与使用橡胶化的键盘相关的问题。首先, 橡胶是多孔的, 这意味着污垢和其他污染物往往在微生物水平更容易聚集, 并更

难擦拭干净,经常要求更多的摩擦和 / 或溶剂来把污染物从橡胶分离。其次,在典型结构中,橡胶键从橡胶基底凸出,所以在键之间仍存在难以到达和通过擦拭来清洁的小间隙。

[0010] 某些橡胶结构通过在常规键盘的整个键盘区域上方伸展光滑的硅橡胶薄片,然后利用橡胶封盖 (cover) 上的图形图像标记键位置来避免键之间的间隙的问题。这种方法使得能够通过擦拭来清洁键盘,并且仍保持封盖下的机械键的行程和点击的触感 (尽管其多少受所述橡胶封盖的缓冲)。但是,橡胶的多孔性质的问题仍存在。最后,橡胶薄膜柔软并且易受裂缝、撕裂和深划痕影响。对于潜在的污染,这些可能导致进一步的问题区域。

[0011] 另一种可清洁结构是薄膜键盘 (或小键盘)。这些通常通过把传感器薄膜夹在硬的平整表面 (例如塑料) 和由乙烯树脂或者 **Mylar®** 制成的顶部封盖之间制成。这些单元被顶部封盖完全密封,允许它们被清洁。传感器薄膜通常由两个其上分别具有导电栅格的薄片组成,当薄膜被紧压在一起时,所述两个薄片相互接触。完成这个所要求的机械行程量非常小,并且实质上对于人类触摸来说无法察觉,意味着当按下键时没有触觉反馈。某些薄膜键盘是完全光滑和平整的,而其他的在顶部封盖中具有小凸起来指示每一个键的轮廓。

[0012] 薄膜键盘的优点是其平整,在键之间没有间隙,使其容易擦拭和清洁。薄膜键盘的主要缺点是其难以有效地在上面打字。经常难以感觉到键,导致用户必须看着键盘。缺少触觉反馈也使打字减慢。最后,致动薄膜键的力的量通常比正常键盘的高得多,导致用户更快地疲劳。

[0013] 解决清洁能力问题的另一种方法是利用在商业上可获得的塑料薄膜覆盖键盘,所述塑料薄膜被模塑以便配合在每一个键上方。所述塑料薄膜是柔软的,允许用户感到每一个键的移动,所以他们仍能够无需观看地打字。封盖可以被去除、洗涤,然后被放回在键盘上。在某些实例中,用户利用能伸展的塑料包装 (商业上叫做 Saran™ 包装) 覆盖其键盘以便使其更容易地可清洁。

[0014] 因为模塑的封盖具有针对每一个键的凹陷,所以当就位时其仍难以清洁;通常为了清洁封盖有必要将其移除。模塑的塑料薄膜也阻碍了键的移动和感觉,使得当其就位时多少更难打字。

[0015] 为了克服键盘的清洁能力的问题,看起来直观的是如果键盘表面自身可以是平整或者接近平整的平坦表面,则擦拭键盘来清洁它将容易得多。但是,这意味着将需要找到对键盘的物理机械或者薄膜键的替代。

[0016] 因此,需要以下述方式对用于键盘输入的上述方法进行改进,该方式容易清洁、允许用户感觉到键、允许用户把其手指放在键上、要求与在标准键盘上相同或者更少的力来按键、对人的触摸有响应,并允许用户与在标准键盘上一样快或者更快地打字。

发明内容

[0017] 本发明的方法的与众不同之处在于触摸电容和振动传感器被相互结合地使用的事实。这种方法使得能够让用户把其手指放在键上,允许他们如同在常规键盘上那样打字。在用户把其手指放在键上时,触摸电容传感器 (每个键一个) 向处理器报告被触摸的每个键的信号强度水平,但是系统不发出击键,直到检测到对应的“轻敲 (tap)” (即振动) 为止。当检测到轻敲时,本发明参考在发生轻敲的时刻之前、期间或者紧跟其后的触摸电容传感器的状态。在轻敲的时刻附近,键的触摸传感器或者从开启、然后关闭、然后再次开启循

环（用户把其手指放在其上的键）时，或者从关闭然后开启循环（用户未把其手指放在其上的键）时，键被确定为被断言。本发明也能够检测有意的键按压和当用户将其手指从一个键滑到下一个键时（与实际的键轻敲相反）之间的差别。例如，当用户能够达到键并利用食指在其上轻敲时，用户的中指可能从一行滑动到下一行。因此，本发明能够在无意的和有意的键致动之间进行区分。

[0018] 可能存在用户需要按压其手指已经放在上面的键的实例。他们可以以两种方式之一这么做：首先，他们可以抬起其手指，然后就像他们将要按任何其他键一样将手指往回向下轻敲在该手指先前所放在的同一键上。或者，用户可以在该键上更用力地按下而不将其手指完全抬起离开该键，在这种情况下，该键的触摸传感器发出比当手指较轻地放在该键上时更强的信号。所述更强的信号被检测，并且键按压（key press）被断言（assert）。

[0019] 本发明的方法相对于现有的平整键盘系统，例如由 Chien 和 Arnon 公开的那些，具有许多非常显著的优点。一个是键盘可以具有触觉键表面，所以用户能够保持其手在键盘上的位置（因而能够不用看着他们的手打字）。在优选实施例中，触觉表面由每一个键上的浅凹陷或者键“井”实现。键井被以仍容易擦拭干净但却深到足以用手指感觉到的方式逐渐倾斜。在另一个优选实施例中，在通常由食指触摸的“定位（home）”键（f 和 j 键）上存在小的凸起。在又一个优选实施例中，在定位键上以及小指通常所放在的键（“a”和“；”键）上键井略微更深。这允许用户更强地锚定键上一般用来将他们的手的位置锚定到键盘的点。

[0020] 与由 Chien 或者 Ziemkowski 描述的只有振动的系统相对比，本发明通过使用每一个键上的触摸电容传感器允许两个或更多个键之间的同时键按压而不要求额外的传感器模式（例如压力）。以类似的方式，键重复（key repeat）在被用户保持把键按下一段延长的时间段所引起时也被承认。

[0021] 本发明的方法相对于现有的电容键盘，例如由 Danish 公开的那种，具有显著的优点。一个这样的优点是用户可以把其手指放在键上而不导致发生键致动。另一个优点是用户可以通过触摸而不必看着键盘来打字。

[0022] 本发明的系统允许用户通过透明的顶部平坦表面观看键的虚拟图像。在优选实施例中，所述虚拟图像被印刷在位于可以具有（或可以不具有）键井的透明顶部平坦表面下面的平整标签上。

[0023] 本发明的系统可以适应无线电池供电实施方案。许多创新被用来节省电池寿命。一个创新是除了可以检测用户的手何时接近键盘的邻近传感器以外，整个系统可以被置于低功率模式（或者“休眠”模式）中。在检测到所述用户的手的存在时，邻近传感器“唤醒”系统的剩余部分。以这种方式，系统在不被使用时保持处于低功率模式。通过把触摸电容传感器置于低功率模式，然后周期性地将其唤醒以便快速地采样每一个键的状态获得了进一步的节电创新。这种方法的占空比使得触摸传感器大多数时间处于休眠模式 - 即使键盘正被使用时 - 因而延长了系统电池的寿命。

[0024] 因为本发明的系统可以是无线的，所以对于它来说在设备运行时被移动很常见。这可能导致无意的键断言（key assertion）。为了减轻这个问题，振动传感器（也叫做加速度计）可以检测设备何时正在移动，并在该时间期间暂时中止操作。一旦键盘停止移动，操作可以被自动恢复。

[0025] 本发明的系统包括扬声器,以便在用户致动键时提供可听的反馈。声音可以是点击、峰鸣、实际键点击的数字化录音,或者任何其他短声音。

[0026] 本发明的方法可以允许多种可被从键盘自身调整的用户设置。这些包括对触摸电容灵敏度、轻敲灵敏度和可听反馈的音量的调整。

[0027] 本发明的方法也检测键盘何时正被擦拭。存在一种双键组合,被分配用于“暂停”键盘以使其可以被清洁。但是,某些用户忘记这么做并且就开始擦拭键盘来清洁它,而没有首先暂停其操作。这可能导致非故意的击键。因此,本发明检测擦拭动作何时正在发生,并自动为用户暂停键盘的操作。恰在检测到擦拭之前被断言并且随后被认为是擦拭运动的一部分的任何击键可以由系统针对每一个无意的键断言发出退格或者删除键被自动反转。一旦用户完成清洁键盘,则键盘暂停灯闪烁,提醒用户利用双键组合重新使能键盘操作。

附图说明

[0028] 下面参考附图详细描述本发明的优选和替换例子:

[0029] 图 1 是示出具体实施如图 2 中所示的本发明的系统的典型硬件部件的硬件框图;

[0030] 图 2 是本发明的平整表面键盘的实施例的示意图;

[0031] 图 3 是本发明的平整表面键盘的实施例的剖视图,示出了组件的各个层;

[0032] 图 4A 到图 4H 示出了实施本发明的方法以便检测有效键按压和擦拭姿态的软件算法的实施例;和

[0033] 图 5A 到图 5E 示出了实施本发明的方法以便检测设备表面的污染和清洁状况的软件算法的实施例。

具体实施方式

[0034] 图 1 示出了触摸/轻敲敏感可清洁键盘 100 的实施例的硬件部件的简化框图。设备 100 包括容纳邻近传感器 120、电容触摸传感器 130、振动传感器 140、音频扬声器 150 和 LED 视觉指示器 160 的平整的平坦表面。传感器部件 120、130 和 140 向 CPU(处理器)110 提供输入,当用户的手接近或者触摸键盘表面时向 CPU 110 通知接触事件,这通常由硬件控制器协调,所述硬件控制器解释从传感器接收的原始信号,并使用已知的通信协议通过可用的数据端口把信息传递到 CPU110。类似地,CPU 110 与硬件控制器通信,用于开启 LED 指示器并把适当的听觉信号输出到扬声器。处理器 110 可访问存储器 170,存储器 170 可以包括临时和/或永久存储设备的组合,以及只读和可写存储器(随机存取存储器或 RAM)、只读存储器(ROM)、例如快闪存储器的可写非易失存储器、硬驱动器,等等。存储器 170 包括程序存储器 180,其包含所有的程序和软件,例如操作系统 181、触摸-轻敲检测软件 182 以及任何其他应用程序 183。存储器 170 也包括数据存储器 190,其包括每一个键的触摸电容传感器状态的数据阵列(dataarray)、用于保持用户选项和偏好的记录的存储设备 192,以及设备 100 的任何元件所需的任何其他数据 193。设备 100 使用已知的通信协议通过可用的数据连接(有线或者无线的)把对应于用户输入的功能传递到主计算机终端 194。

[0035] 为了最大化电池寿命并节省功率,设备 100 在其不被使用时具有低功率操作模式很重要。在典型的使用场景中,当用户使其手与设备接触时,邻近传感器 120 检测这一情况并发送信号唤醒 CPU 110(其可以已处于或未处于低功率或“休眠”模式),CPU 110 进而唤

醒系统 100 的剩余部分。

[0036] 当用户的手指与平整的平坦表面接触时,电容触摸传感器 130 被断言。CPU 110 周期性地采样每一个触摸传感器的状态,并把结果存储在数据存储器 191 中。在操作系统软件 181 的支持下,CPU 确定每一个被断言的触摸传感器的位置,并将其位置映射到功能(通常是键盘键)。CPU 110 和触摸/轻敲检测软件 182 独立地监视振动传感器 140 的状态。当用户在平坦表面上轻敲时,通常是击打一个键,对应振动被穿过系统发射,该振动由振动传感器 140 检测。当检测到有效轻敲时,由软件 182 支持的 CPU 110 执行对包含在存储器 191 中的传感器数据的算法分析以确定平坦表面的哪个区域被轻敲。如果算法确定有效的轻敲-触摸(见图 4A 到图 4H),则该位置的映射的功能被发送到主计算机终端 194。

[0037] 在一个实施例中,振动传感器也是能够检测运动的加速度计。这被用来检测设备何时正被移动,允许使设备的操作暂时中止直到其停止移动为止的选项。该方案帮助避免键盘正被移动时发生无意的击键,并且也使得用户能够在其移动键盘时触摸键盘表面上的任何地方。

[0038] 图 2 示出了代表典型的主计算机终端 205 和本发明 200 的示意图。平整或者接近平整的表面 200 被呈现给用户,用户在该表面 200 上利用他们的手或者手指做出选择。在一个实施例中,平坦表面 200 呈现键盘键 210、230、240 等以及具有对应的左 280 和右 270 鼠标按钮的鼠标触摸板 260 的图像。提供特殊功能键 290,当该键 290 与其他键同时被按压时作为在键盘上设置用户偏好的手段。例如,功能键 290 被与暂停键 240 同时按压以便暂时中止键盘的操作(通常使其可被清洁)。特定键上的其他 LED 指示器给用户提供了指示大写锁、数字锁、暂停状态的反馈。

[0039] 在又一个实施例中,可以在键盘上直接设置其他的用户设置。使用单独的一行 LED 215 来指示 0 到 5 的通用水平。为了进入设置模式,用户同时按压功能键 290 与多个其他键 235 中的一个,以便指示其希望轮询或者改变的设置。LED 指示器在被选择的设置键 235 上方亮起,同时,在通用水平 LED 0 到 5 上指示设置的当前水平。然后,用户可以通过按压通用水平 LED 下方的期望的键来改变设置的水平。为了退出设置模式,用户再次同时按功能键和设置键,或者等待预定的时间量,此后系统自动退出到正常模式。在该实施例中,用户做出轮询并设置下列设置:音量、触摸灵敏度、轻敲灵敏度、清洁水平、无线信号强度、电池强度,以及邻近传感器的灵敏度。

[0040] 在又一个实施例中,设备的顶部平坦表面具有浅的凹槽,或者“键井”,用于提供指示键的位置的触感。键井足够深以允许人的手指感觉到它们,但又足够浅以便仍允许轻松擦拭和清洁表面。一般被称作定位键(在英文键盘上通常是“f”和“j”键)230 的两个键上可以具有额外的触觉标记,例如像一般在标准键盘上找得到的小凸起。在又一个实施例中,在定位键 230 上以及小指通常所放在的键 210(“a”和“;”键)上键井略微更深。这允许用户更强地锚定研究表明用户通常将其手指放在其上的键上的点。

[0041] 图 3 示出了构成本发明的实施例的顶部层的部件的剖视图。顶部透明平坦层 300 是用户触摸的最外面的层。浅凹陷或键井 340 位于每一个键上。通过印刷、绘制、雕刻或者粘附标签,把键盘键的图像 310 呈现在顶部平坦表面 300 的下侧。印刷电路板 320 被附着于平坦表面 300 的下侧,其具有构成触摸传感器的导电迹线 330。其他的电子部件,例如 LED 和振动传感器 350 位于印刷电路板 320 的下侧。

[0042] 图 4A 到图 4I 示出了实施可清洁触摸和轻敲敏感键盘的软件 182 的实施例的流程图。图 4A 示出了可清洁触摸和轻敲敏感键盘软件 182 的主处理例程 4100 的实施例的流程图。在块 4105, 当过程初始启动时, 各种系统变量被初始化, 包括触摸传感器的数字状态 (Q) 和触摸传感器的模拟水平 (A)。在块 4110, 过程等待预定轮询周期, 然后在块 4115 中捕捉所有触摸电容传感器的值。在块 4120, 调用图 4B 的键释放检测例程, 以确定先前被断言的键是否已经被释放。在块 4125, 调用图 4C 的键按压检测例程以确定先前释放的键是否已经被断言。在块 4130 调用图 4F 的擦拭检测例程以确定在键盘上是否已检测到擦拭运动, 并且如果检测到, 则自动地中止键盘的操作。擦拭检测例程也可以用来确定键盘是否已被充分地清洁。

[0043] 图 4B 示出了键释放检测例程 4200 的实施例的流程图。在块 4205, 获取针对当前采样 (N) 存储的每一个键传感器状态, 并且在块 4210 中将其与相同键的来自前一采样 (N-1) 的状态进行比较。如果在前一采样中键被断言 (即值 = 1), 但是在当前采样中未被断言 (即值 = 0), 则在块 4215 中确定该键刚刚经历了有效的键释放状态。在块 4220 中, 所有的键被这样评估, 然后在块 4225 中, 例程返回。

[0044] 图 4C 示出了键按压检测例程 4300 的实施例的流程图。这个例程通过在块 4305 中确定是否已发生由振动传感器 140 确定的有效轻敲开始。如果未检测到轻敲, 则在块 4310 中例程继续到图 4I 中的按压检测例程。如果检测到有效轻敲, 则过程通过清除用于候选键 (即已经将其状态从未被断言改变到被断言的键) 和有效键 (即已经被确定为被轻敲的键) 的两个数据容器开始。因为轻敲传感器和触摸传感器可能未精确地在同一时刻被断言, 在块 4320 和块 4325 中, 每一个键的状态的比较被当前采样以及过去的两个 (或更多个) 采样括在一起 (bracket)。在块 4330 中, 例程确定正被检查的当前键是否已从未被断言改变到被断言, 并且如果是, 则在块 4335 中其被标记为候选键。过程在块 4340 中继续, 直到所有键已经被检查为止。如果没有候选键被识别, 则过程在块 4350 中继续到图 4D。如果存在候选键, 则过程在块 4355 中继续, 直到队列中的所有键已经被检查为止, 并且在块 4360 中例程返回。

[0045] 图 4D 示出块 4400 中图 4C 的键按压检测例程的继续的实施例的流程图。在例程的这个点上, 至少一个 (或更多个) 键已经被识别为候选键 (即键的触摸传感器状态已经从未被断言改变到被断言)。触摸电容键盘的常见问题是“幻象 (phantom)”击键, 这是当用户正在利用另一个手指打字时, 用户将其手指从一行滑动到相邻行的结果。当出现意外的滑动时, 当手指在两个键之间滑动时, 通常存在两个相邻键被同时断言的时刻。在块 4405 中例程检查每一个候选键, 并在块 4410 中将其状态与不同行上与其相邻的键进行比较。在块 4415 中, 如果相邻键在与被断言候选键的当前采样相同的时间被断言, 或者如果相邻键恰在被断言的候选键的当前采样之前被断言, 则这种情形被视为是手指滑动的结果, 并且候选键被忽略。块 4416 示出了采样数据集合表, 其中, 由于候选键和相邻行键被在同一时刻断言, 所以确定了键滑动。在表中, 可以看到虽然候选的被断言状态从 0 变到 1 (给予其作为候选键的资格), 相邻行键从被断言转换到未被断言, 具有至少一个采样集合 n-1, 其中候选键和相邻行键都被同时断言。块 4417 示出了采样数据集合表, 其中, 由于相邻行键恰在候选键被断言的时刻之前被断言, 所以确定了键滑动。由于在第一手指位于候选键上的同时把第二手指放在相邻行键上非常不自然并且很困难, 所以假设在 4415 中发现为真

的情形最可能作为用户把同一手指从一个键滑动到相邻行上的键的结果而发生。由于触摸-轻敲可清洁键盘的前提是:为了使键有效,其必须被轻敲或者按压(而不仅仅是滑动到该键),所以在 4415 中发现为真的情形被排除出有效键按压。在块 4420 中,这个过程针对每一个相邻行键继续。如果发现候选键并非作为滑动运动的结果而发生,则其在块 4425 中被添加到有效键集合。在块 4430 中,这个过程针对每一个候选键继续。一旦已在每一个候选键上执行过滑动分析,则过程继续到图 4E 中所示的流程图。

[0046] 图 4E 示出块 4500 中图 4D 的键按压检测例程的继续的实施例的流程图。块 4505 检查有效键的总数目,并确定该数目是否超过可接受阈值。如果是,则键按压被忽略。例如,用户可能把其整个张开的手往下拍在键盘上,这将导致多个有效键,尽管用户很显然地并未在打字。如果情况如此,则例程在块 4515 中返回。否则,例程确定有效键被按压,并且在块 4520 中把命令发送到主计算机终端 194。过程在块 4525 中针对每一个有效键继续,然后在块 4530 中返回。

[0047] 图 4F 示出了擦拭检测例程 4600 的实施例的流程图。对于用户来说忘记在清洁键盘之前禁用(或“暂停”)键盘的操作很常见。因此期望检测在键盘表面上何时发生擦拭运动,并自动为用户禁用键盘以避免无意的击键被发送到主计算机终端 194。在块 4605 调用图 4H 的传感器足迹检测例程来确定传感器的当前“足迹(footprint)”的大小(术语“足迹”用来描述一组被同时断言的相邻的键传感器)。块 4610 清除所有足迹记录的更新标志以便为将只检测新形成的足迹或者已经移动的先前足迹的分析做准备。块 4615、4620 和 4625 检索每一个被检测的足迹和每一个现存的足迹,并比较其大小和位置。如果足迹在大小和位置上类似,则块 4630 确定它们之间的大小变化和移动向量,在块 4635 中更新 F,并在 4650 中将其更新标志设置为真。如果 DF 和 F 足迹在大小上不类似,则在块 4640 中过程继续检查其他的足迹记录。一旦所有足迹已被比较,则在块 4645 中过程生成新的足迹记录,并将其更新标志设置为真。在块 4655 中,过程针对其他的检测足迹继续。一旦所有被检测的足迹已被处理,则在块 4660 中去除所有其更新标志为假的足迹记录,并且在块 4665 中过程继续到图 4G 中所示的流程图,沿着新形成的足迹和已经移动过的预先存在的足迹通过。

[0048] 图 4G 示出了继续来自块 4700 中的图 4F 的擦拭检测例程的实施例的流程图。在块 4705 中,每一个足迹记录(F)被检查。块 4710 确定是否存在足够的足迹记录来继续进行擦拭检测流程。如果是,则块 4715 确定足迹的移动是否沿相同的方向,以及足迹的大小是否相对恒定。如果是,则在块 4720 中声明有效的擦拭,并且过程在块 4725 中终止。如果被分析的足迹不具有作为擦拭的资格,则在块 4730 中例程继续检查所有的剩余足迹,然后在块 4735 中返回。

[0049] 图 4H 示出了传感器足迹检测例程 4800 的实施例的流程图。这个例程的目的是找到许多相邻的触摸传感器被同时断言的区域(即“足迹”)。例程通过在块 4805 中检查触摸电容传感器历史阵列 191 中的每一个键传感器开始。每一个键(k)在块 4810 中被处理并在块 4815 中查看其目前是否被断言。如果是,则在块 4820 中生成新的潜在足迹。块 4825 检查与键(k)相邻的每一个键(j),并在块 4830 中检查其是否被断言。如果相邻键(j)被断言,则块 4835 查看其是否已被包括在潜在足迹中。如果没有,则块 4840 把相邻键(j)添加到键(k)的潜在足迹。块 4845 确保键(k)和相邻键(j)相互参考。块 4850 把相邻键(j)标记为已处理。过程在块 4855 中继续,直到所有潜在的相邻键已经被检查为止,此后块

4860 把键 (k) 标记为已处理。整个过程在块 4865 中重复,直到所有的键已经被检查为止。

[0050] 图 4I 示出了静止后按压检测 (Press-after-Rest Detection) 例程的实施例的流程图。过程通过在块 4810 中清除有效键集合 (V) 开始。块 4820 检查从当前采样 (N) 往下到确定的先前采样 (在这个例子中示为 N-2) 的队列 (e) 中的每一个条目。块 4830 针对每一个键 (ke) 检索触摸传感器的模拟信号值 (A)。块 4840 确定模拟信号值 ke 是否高于静止阈值 (即典型的静止手指的传感器读数)。如果模拟信号值 ke 不高于静止阈值,则过程跳到块 4870,在块 4870 处理下一个键。如果模拟信号值 ke 高于静止阈值,则块 4850 确定 ke 和前一采样 ke-1 之间的模拟信号值的变化是否超过按压阈值 (被示为 SLOPE(ke, ke-1))。块 4850 也确定恰在 ke-1 之前的变化是否是轻轻抬起,跟着是按压 (考虑到刚好在断言按压运动之前用户可能在键上略微抬起其手指而没有完全使其离开该键时的实例)。如果存在这些状况的任何一个,则在块 4860 中过程把 ke 添加到有效键集合 (V)。如果这些状况中的任何一个都不存在,则过程通过在块 4870 中检查下一个键 A(e) 继续。在块 4880 中,一旦所有键被考虑,则过程在块 4890 中退回到图 4E 中所示的流程图。

[0051] 存在这里所公开的键盘的许多可选择的有益方面。可选择的有益方面的例子集合在下面逐条列举:使用硬化丙烯酸、玻璃、退火玻璃或其他抗刻划透明材料用于顶部表面;使用抗微生物固态表面材料作为外壳的基底;使用铝(成形的、机加工的等)作为外壳的基底;把透明顶部表面附着于基底的方法;密封外壳以对抗液体或者灰尘侵入到外壳的内部;使用氧化铟锡(ITO)提供用于触摸传感器的导电透明迹线;和/或通过使用 ITO 技术,通过刻蚀或者其他方式把键图像附着到玻璃以及 ITO 迹线的下侧而使键盘完全透明的能力;和使用可更换的搭锁式透明封盖用于顶部表面。

[0052] 键盘理想地适于轻松的清洁和消毒。如上所述,键盘的操作可以被暂时中止,或者通过键组合,或者通过擦拭检测,然后被清洁或消毒。在另一方面,系统能够确定触摸表面何时有可能已污染到其应该被清洁的程度。系统通过追踪自从上次清洁以来在触摸表面上发生的累积活动来实现它。通过组合以下因素中的一个或更多个来确定活动水平:表面上触摸的数量(使用触摸传感器 130)、在表面上方滑动的数量(使用触摸传感器 130)、在表面上轻敲的数量(使用振动传感器 140)、键盘移动的量(使用振动传感器 140),以及邻近传感器被激活的次数(使用邻近传感器 120)。当活动水平到达用户可设置的阈值时,则激活污染警报(其可以是灯、一系列灯、可听声音,或者到主计算机终端的电子通知)。

[0053] 在一个实施例中,主计算机终端是计算机网络的一部分,管理员通过计算机网络可以监视连接到主计算机终端的每一个键盘的污染状态。例如,如果任何键盘超过了污染阈值,则医院感染控制人员可以被通知,并可以采取步骤快速地使那些设备被消毒,从而保持感染受控环境的完整性。

[0054] 在另一方面,系统能够确定触摸表面何时已被充分擦拭足以清除污染警报。当擦拭动作发生时,“足迹”或者多个被激活的相邻触摸传感器横穿表面/键盘的平面。足迹的路径被系统观测,从而使得能够检测擦拭活动是否在整个键盘表面上发生,以及发生了多少来回的擦拭。一旦擦拭活动达到用户可定义的阈值,则污染警报被解除激活,如下面在图 5A 到图 5E 中所说明的那样。

[0055] 图 5A 到图 5E 示出实施污染和清洁检测过程的软件 182 的实施例的过程流程图。图 5A 示出可清洁触摸和轻敲敏感键盘软件 182 的污染和清洁检测例程的主处理例程 5100

的实施例的流程图。块 5110 初始化“活动水平”变量 AL。这个变量用来追踪键盘表面上可能对其污染有贡献的活动的量。块 5120 查看是否已经超过污染阈值（见图 5B）。如果未超过阈值，则例程在循环中继续，直到超过污染阈值为止，在超过污染阈值后，在块 5130 中，其激活污染警报。块 5140 确定键盘用户设置是否被设置成使用清洁擦拭检测例程。如果不是，则过程在块 5150 中等待键盘暂停和取消暂停，然后清除污染警报。如果键盘用户设置被设置成使用清洁擦拭检测例程，则过程在块 5160 等待，直到已超过清洁阈值为止（见图 5C）。一旦已经达到清洁阈值，则在块 5170 中过程清除污染警报，然后返回以便在块 5110 从起始处再次开始所述过程。

[0056] 图 5B 示出了可清洁触摸和轻敲敏感键盘的污染检测例程 5200 的实施例的流程图。块 5205 检查每一个触摸传感器，并在 5210 中确定其是否被断言。如果其被断言，则在块 5220 中，活动水平计数器被增加（increment），并返回以便在块 5230 中检查其他键。块 5235 确定设备是否正被移动（通过振动传感器，也叫做加速度计）。如果是，则在块 5250 中活动水平计数器被增加。块 5255 确定用户的手是否已接近键盘（通过邻近传感器 120）。如果是，则活动水平计数器在块 5260 中被增加。在块 5265 中计算所有活动触发的加权以确定给定时间片 n 的污染水平。

[0057] 图 5C 示出了可清洁触摸和轻敲敏感表面设备的清洁检测例程 5300 的实施例的流程图。块 5305 确定如果有的话存在什么样的“足迹”（见图 5E）。在这个图中的流程图的剩余部分确定识别的足迹是否是有效的移动足迹，和针对图 4F 描述的相同（参考上面对图 4F 的描述）。然后，所述过程通过块 5365 继续到图 5D。

[0058] 图 5D 示出了继续可清洁触摸和轻敲敏感表面设备的清洁检测例程 5400 的实施例的流程图。块 5405 检查从图 5C 中的分析确定的每一个足迹记录 (F)。块 5410 确定足迹是否在触摸表面的边缘或者靠近所述边缘。如果不是，则过程继续查看其他足迹。块 5415 确保构成足迹的所有传感器在相同的大致方向上移动并且足迹的大小保持大致恒定。如果不是，则其被确定不是有效的移动足迹，并且在块 5420 中“F”被从候选足迹去除。否则，在块 5425 中增加清洁水平计数器（因为擦拭已经横穿到触摸表面的边缘）。在块 5430 中这针对所有足迹记录继续，然后在块 5435 中返回。存储在存储器 192 中的用户设置确定清洁水平阈值。一旦已超过清洁水平阈值，则清除污染警报，并且表面被视为已被充分地清洁。

[0059] 在另一方面，系统能够确定触摸表面何时已被充分擦拭足以清除污染警报。当擦拭动作发生时，“足迹”或者多个被激活的相邻触摸传感器横穿键盘的平面。足迹的路径被系统观测，从而使得能够检测擦拭活动是否在整个键盘表面上发生，以及发生了多少来回的擦拭。一旦擦拭活动达到用户可定义的阈值，则污染警报被解除激活。

[0060] 在另一方面中，键盘可以被放置在消毒设备（通常称为“高压灭菌器”）中。

[0061] 在另一方面中，动态显示器（例如 LCD 或者“电子墨水”显示器）可以被放置在顶部平坦表面下方，以使每一个键的功能分配以及相关联的视觉显示可以被动态地改变，此后这被称为“软键”。

[0062] 在另一方面中，动态键分配可以由用户手工确定，或者由系统根据主计算机上的上下文变化自动确定。例如，一个软键集合可以为当前应用程序显示，但是随后当新的应用程序变为激活时自动地变化。各个键也可以在应用程序自身内变化，依赖于所遵循的工作流。这个特征的实施方案可以利用由主计算终端上的操作系统提供的某些工具（例如

Windows 操作系统中提供的窗体小部件 (Widgets) 和辅助显示 (Sideshow))。

[0063] 期望给用户提提供某种对触摸和键的致动的触觉反馈。这可以以很多方式实现,其包括但不限于“击针 (striker)”,击针被遍及键散布,其从下方击打玻璃表面并对键按压给予触觉响应(即用户无论何时触摸键,其都将感觉到击针打中玻璃的明显振动)。

[0064] 在另一个实施例中,键盘将在键被致动时振动(与蜂窝电话或者寻呼机振动类似)。

[0065] 在又一个实施例中,触觉致动器被置于键盘的触摸表面和支撑基底之间。当键被按压时,整个顶部表面对着触觉致动器略微降低,这产生了触觉响应(这与简单的“敲击”开关的感觉和构造类似)。

[0066] 因为设备的表面是连续的,各个键不具有其自身的机械移动结构(如在传统键盘上那样)。因此,用户有可能以较小的误差余量错过在期望的键上轻敲(例如在两个键之间)。在这种情况下,系统仍将检测到轻敲,但是轻敲的键位置将是不明确的。为了帮助提高准确性,基于词典的数据库可以被存储在系统中,包括常见单词、常见字母对和常见的下一词的数据库。当发生不明确的键激活时,算法可以通过参考数据库找到所有候选不明确字母的最可能的字母消除选择的歧义。这种歧义消除逐字母地完成,而不通过显示机制向用户展示任何替代选择(这使其区别于现有技术)。

[0067] 因为表面是触摸敏感的,并且允许用户把其手指放在键的表面上,所以很难确定当用户打算一个键被断言为被保持按下时和当他们只是在键上轻敲后把其手指放在该键上时之间的差别。通过实施用于初始化键重复(或者键按压和保持)的特殊姿态来解决这个问题。用户首先在期望的键上轻敲,从该键快速地抬起其手指,然后快速接着再次轻敲并保持其手指在该键上。从那一刻起,只要用户保持其手指在该键上,则该键被视为被断言(或被按压)。为了结束键断言,用户从该键抬起其手指。

[0068] 在又一个实施例中,可以使用光学方法确定触摸表面的污染水平。例如,透明顶部表面形成了光通道,红外或者可见光通过所述光通道从一侧传送到另一侧,在所述另一侧光电二极管检测传播的光的水平。当在触摸表面的外部上出现例如油性指纹的污染时,一定量的光从所述通道漏出(采用斯内耳定律)。在又一个实施例中,可以使用激光扫描触摸表面来检测污染。

[0069] 虽然这里所描述的实施例的焦点是用于键盘应用,但是本领域技术人员将看到所述系统也可以被成功地应用于任何类型的触摸屏设备。

[0070] 虽然如上所述已经示出并描述了本发明的优选实施例,但是不偏离本发明的精神和范围可以做出很多变化。因此,本发明的范围不受优选实施例的公开限制。相反,本发明应该完全通过参考下面的权利要求来确定。

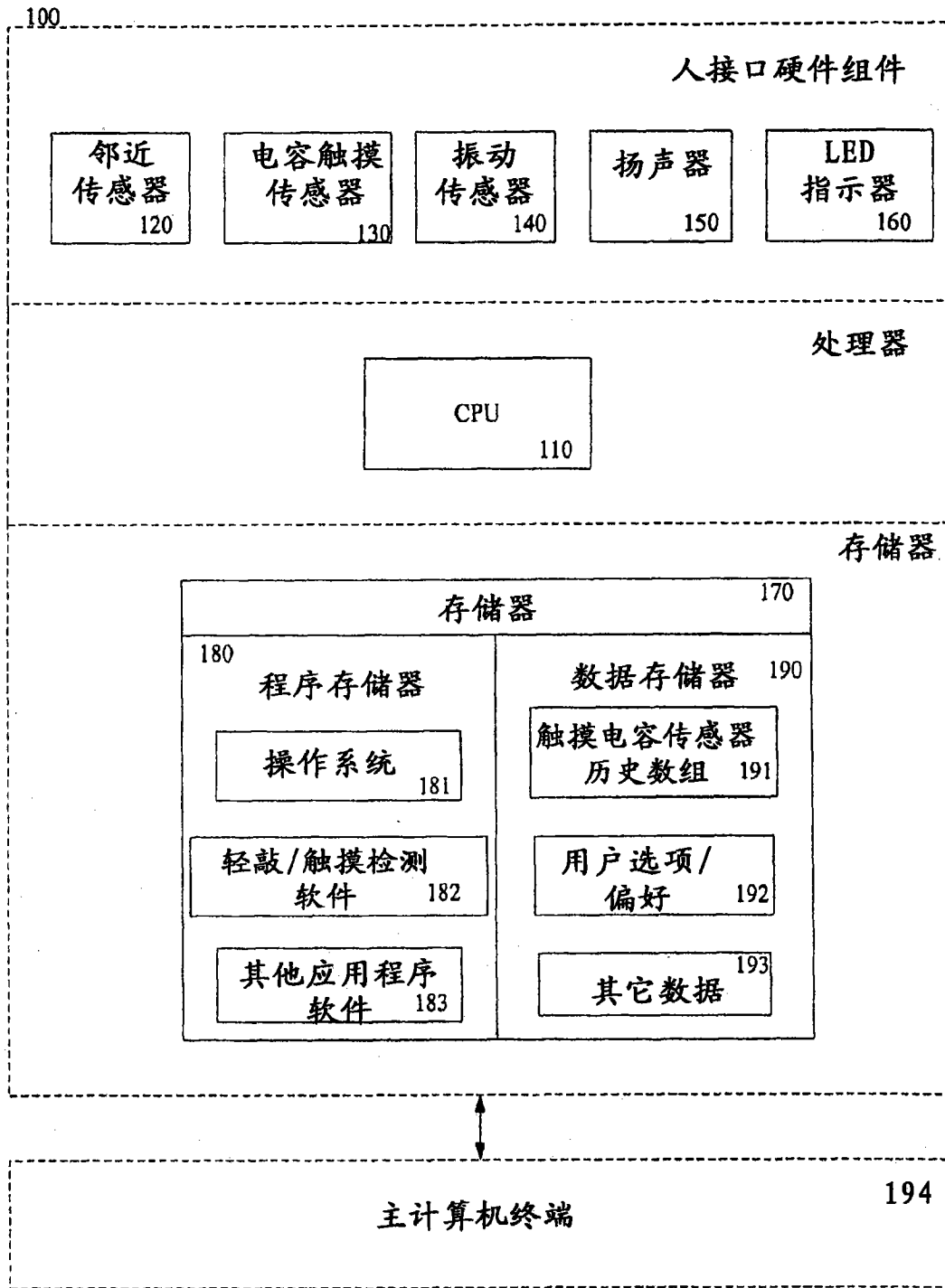


图 1

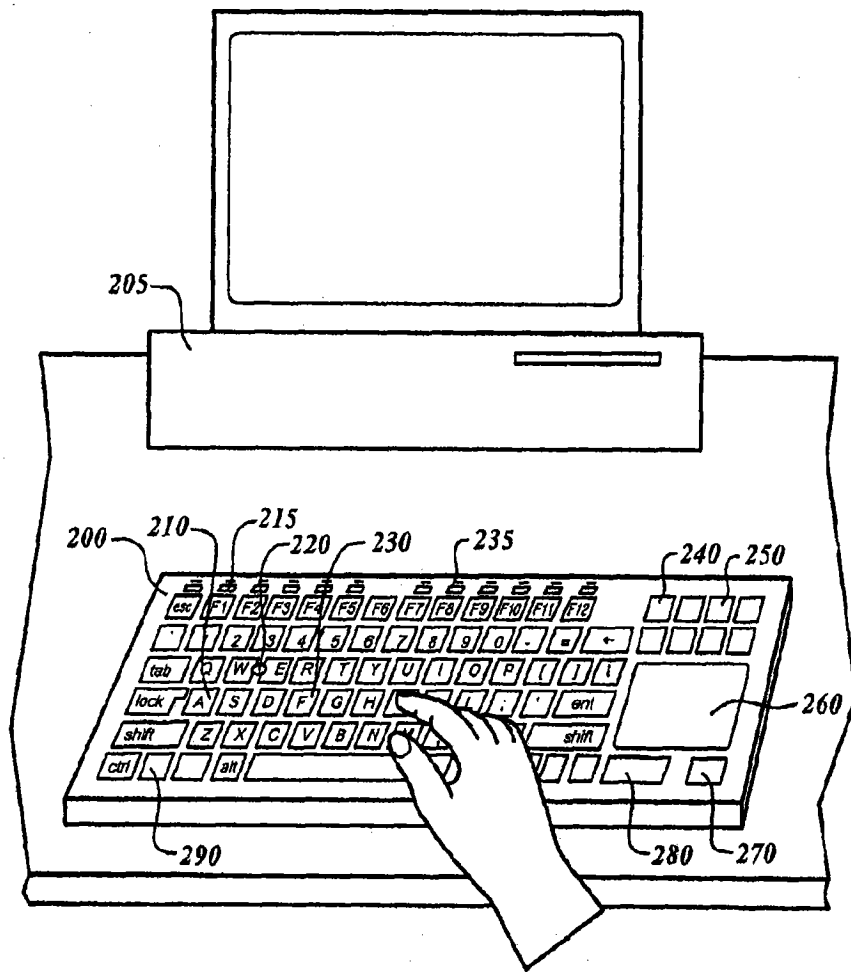


图 2

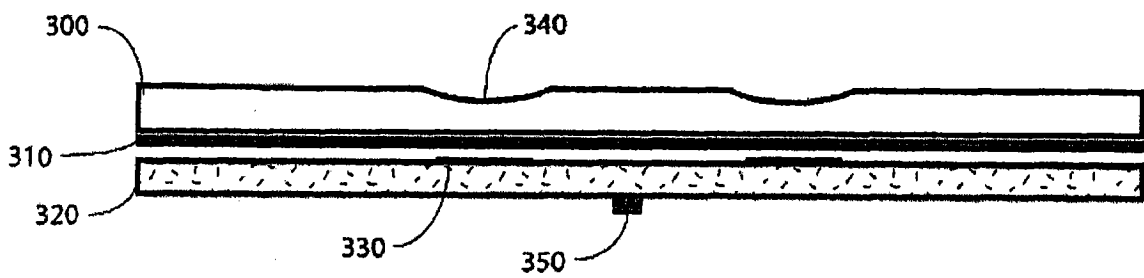


图 3

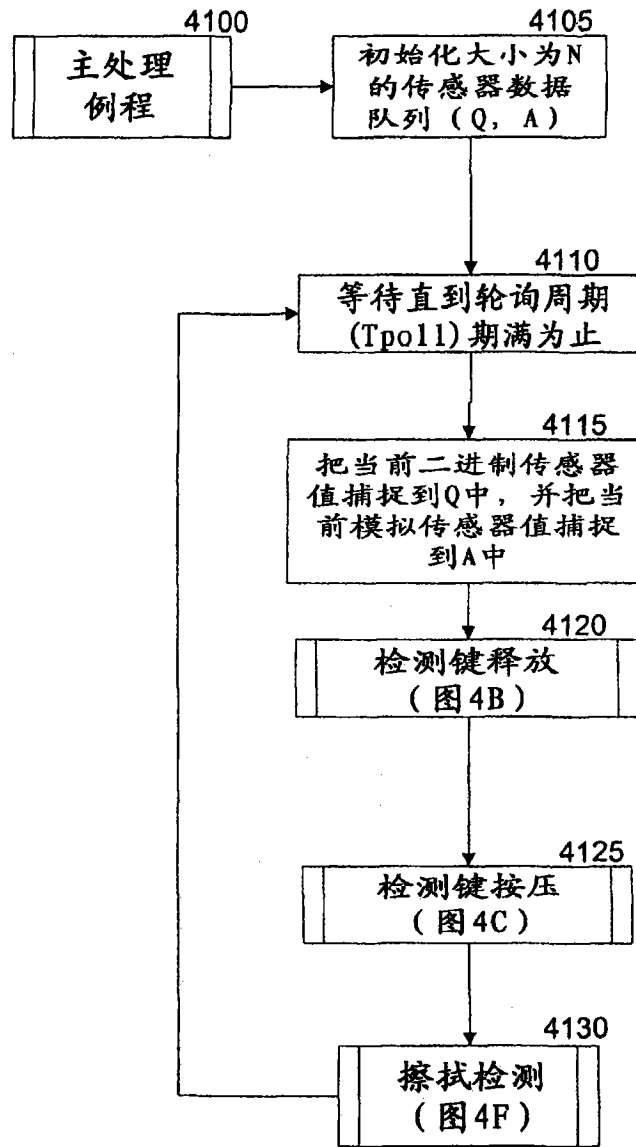


图 4A

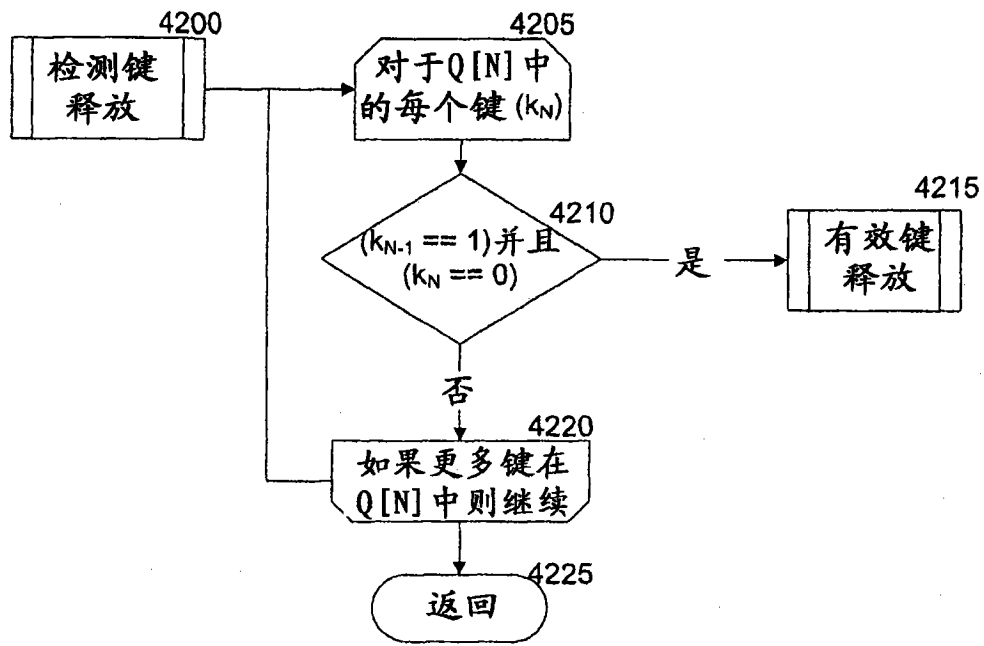


图 4B

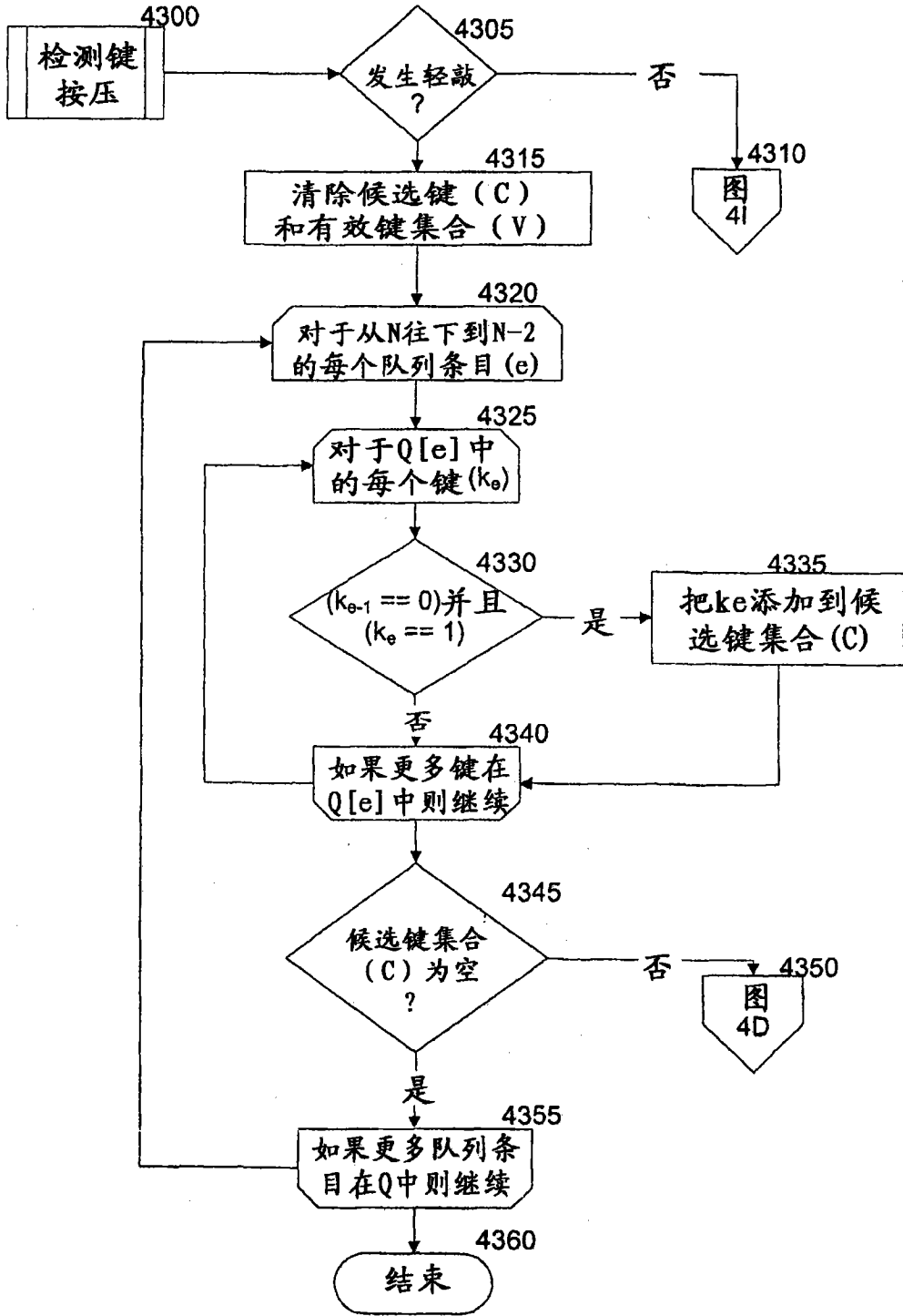


图 4C

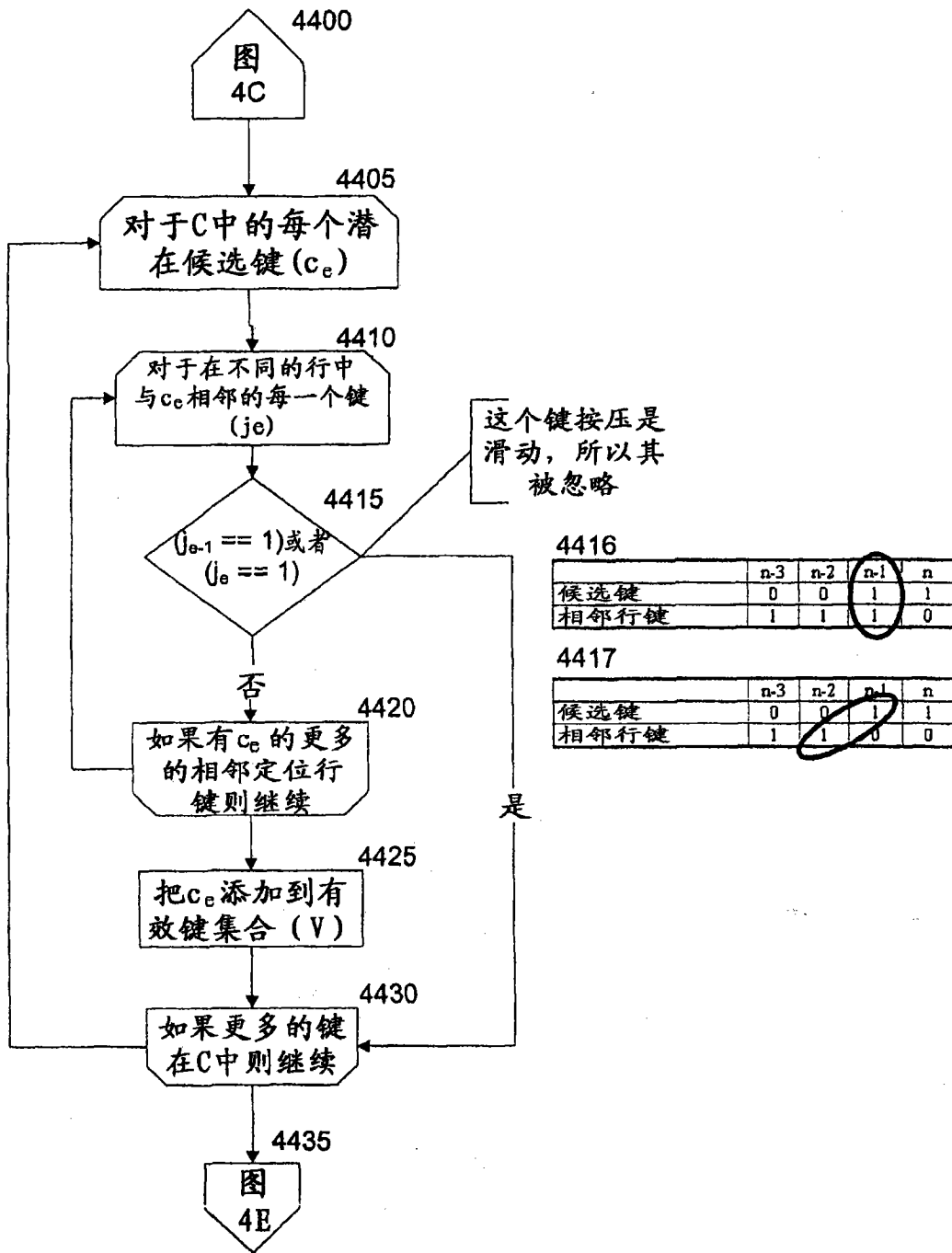


图 4D

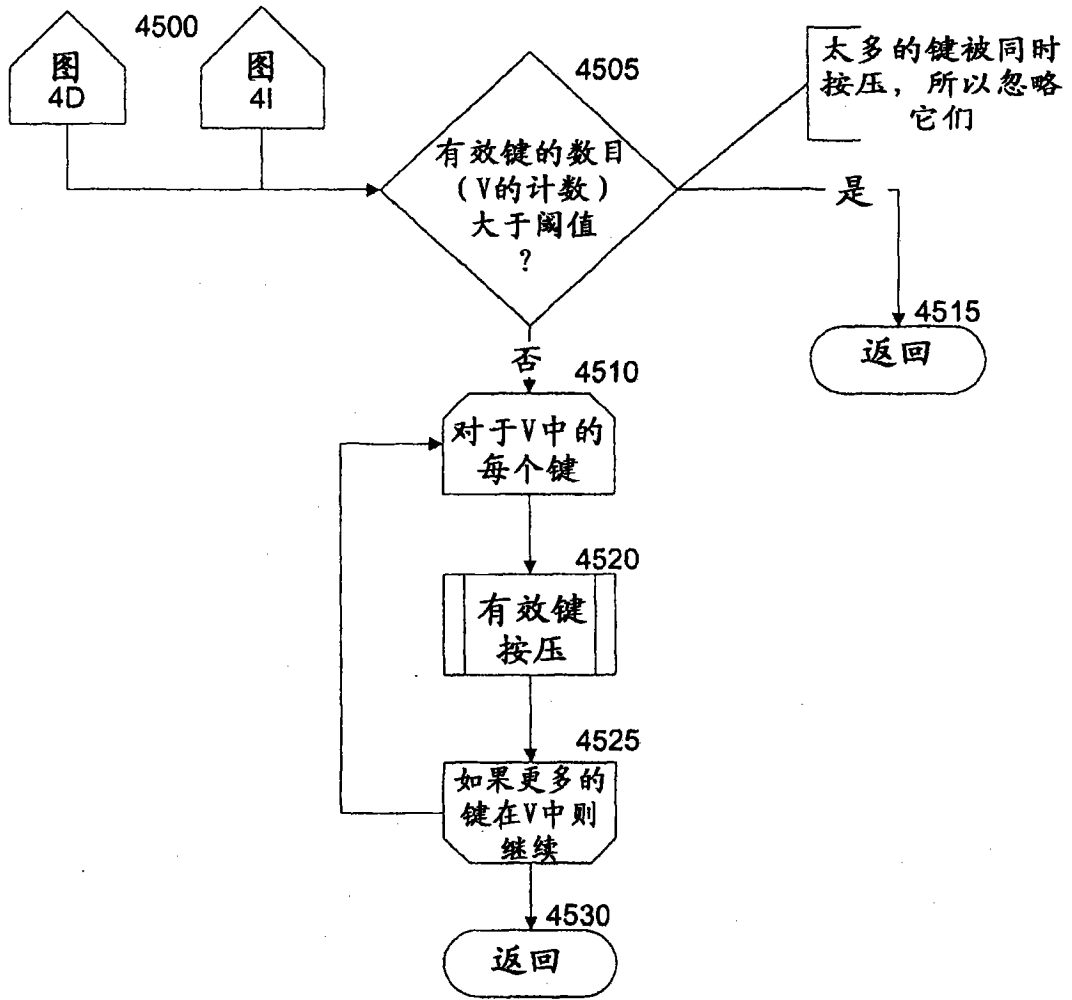


图 4E

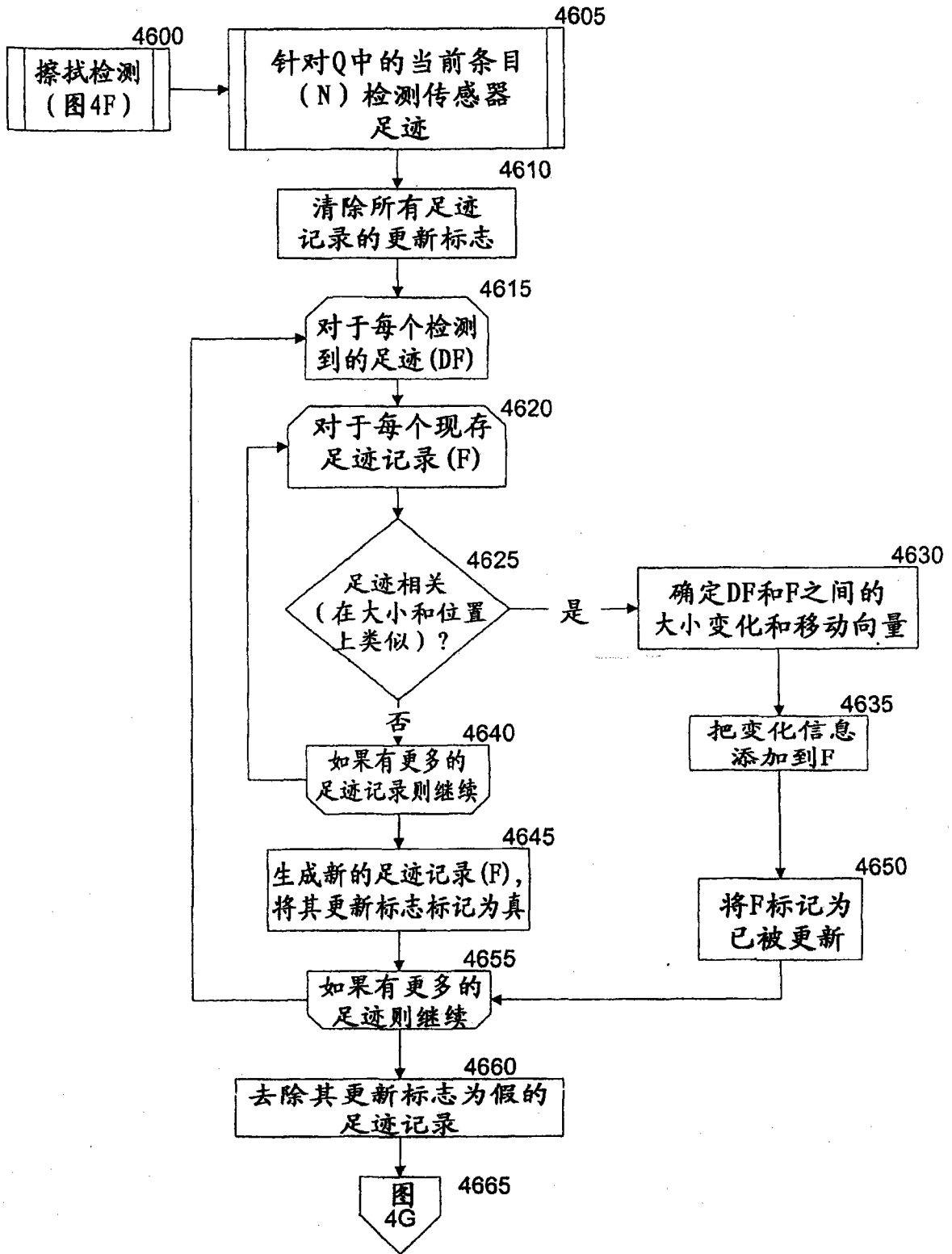


图 4F

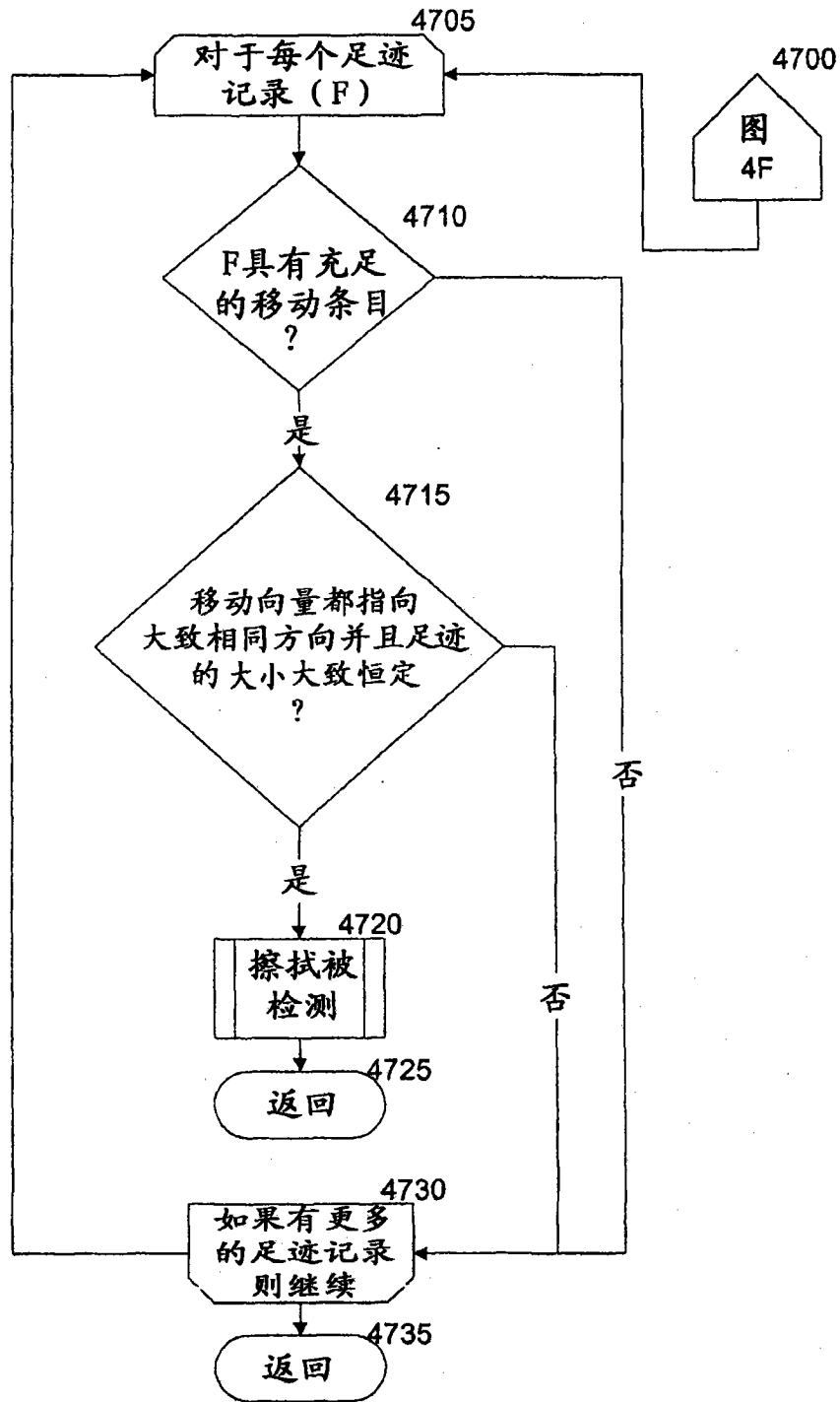


图 4G

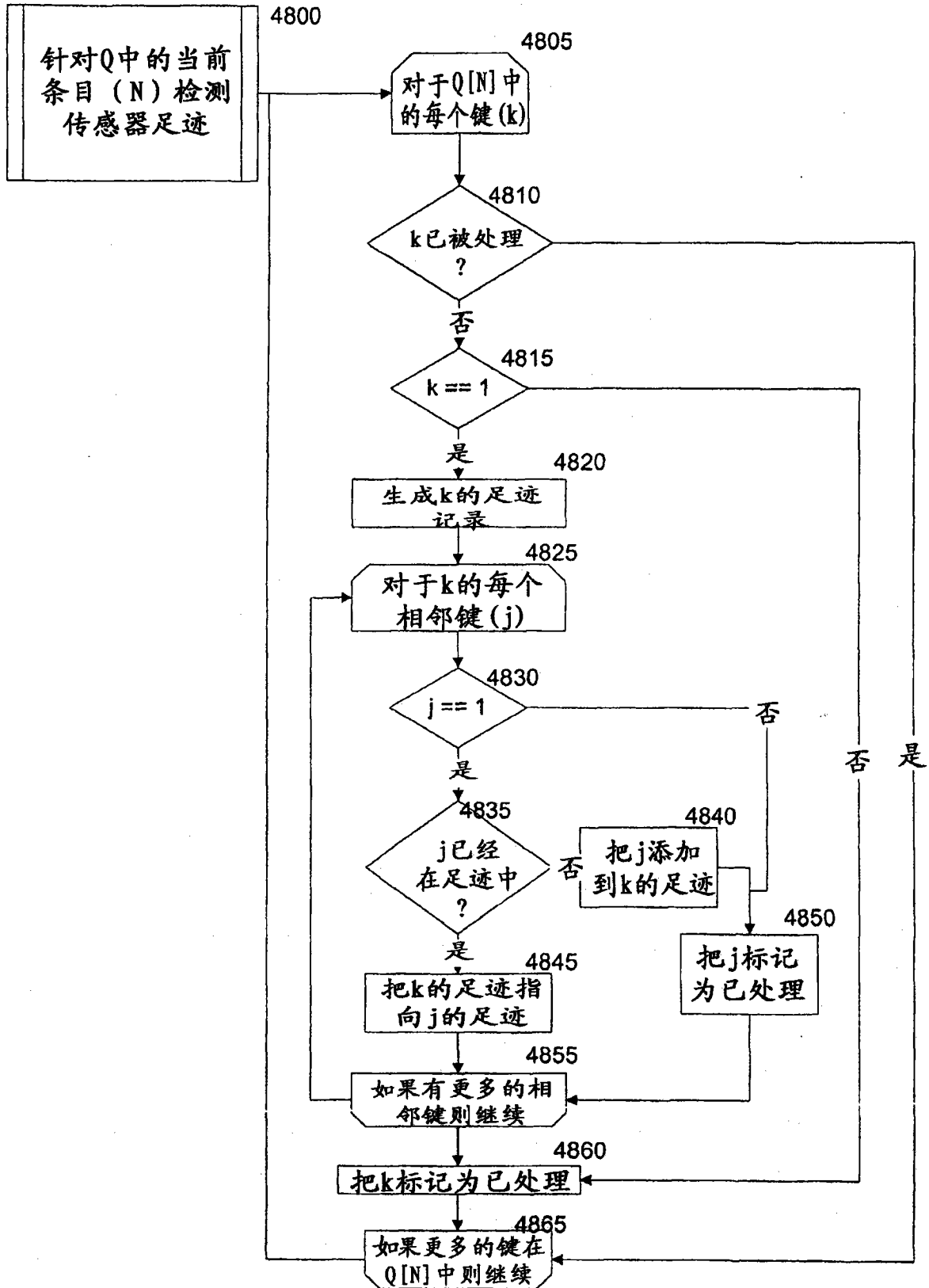


图 4H

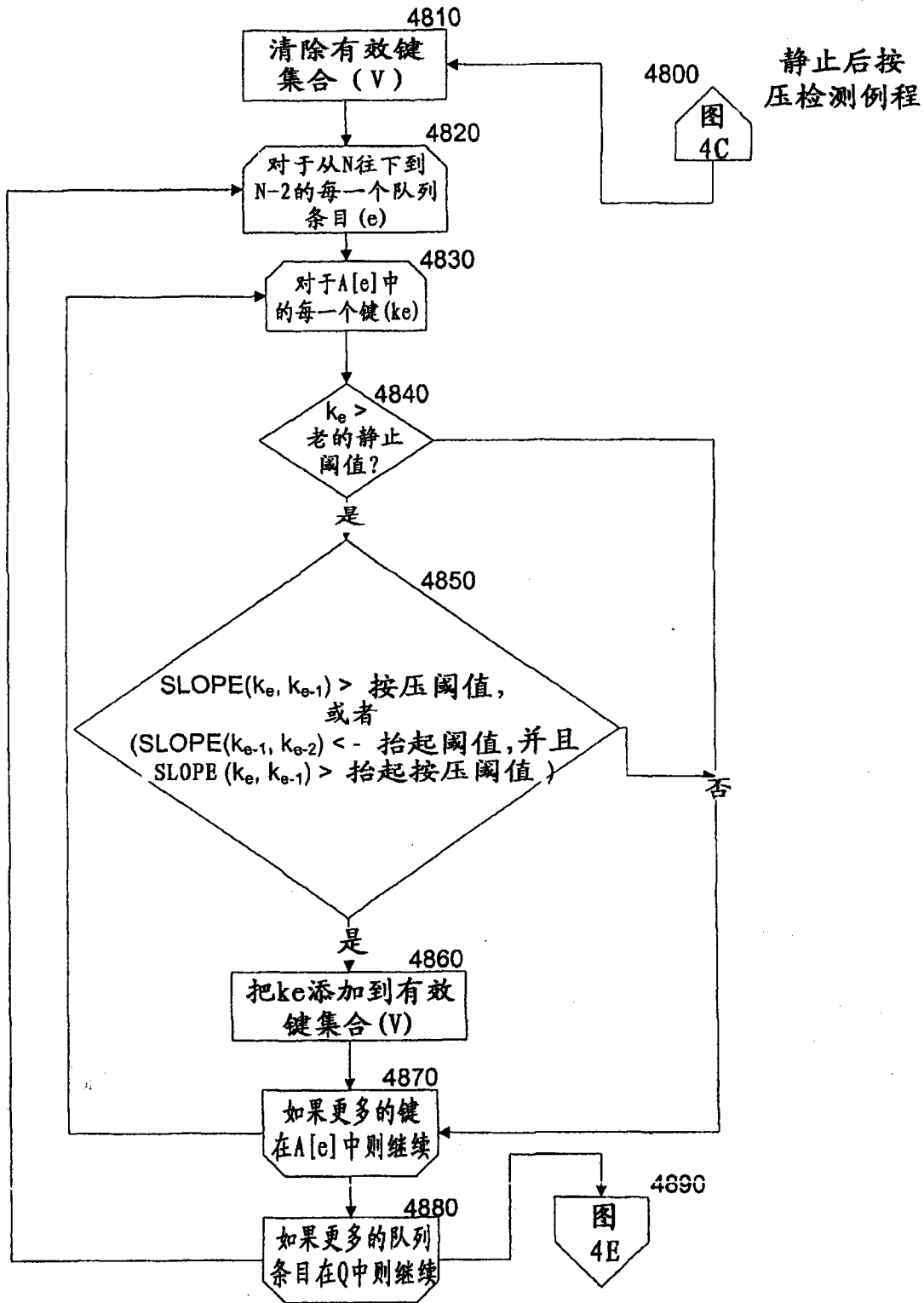


图 4I

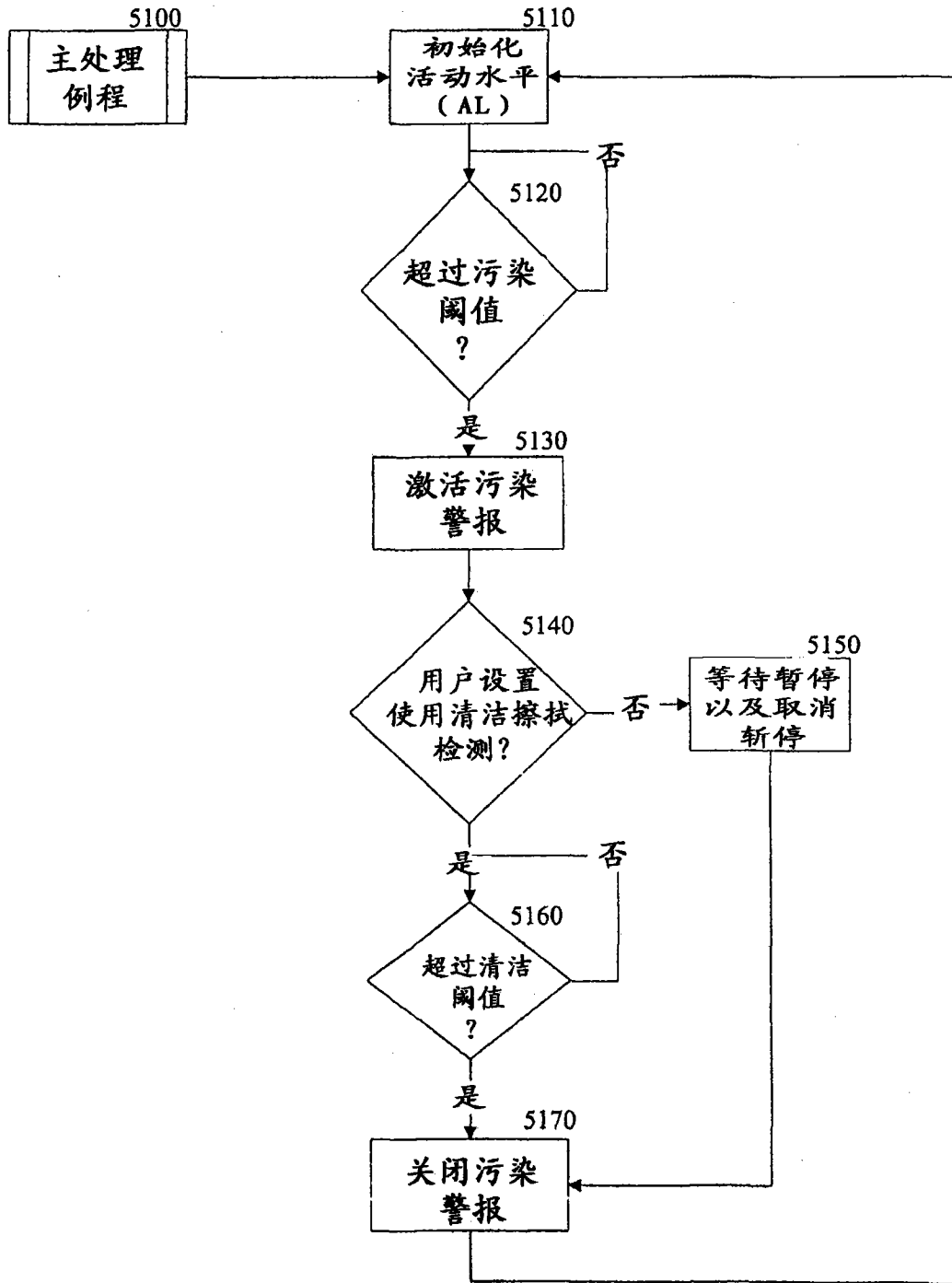


图 5A

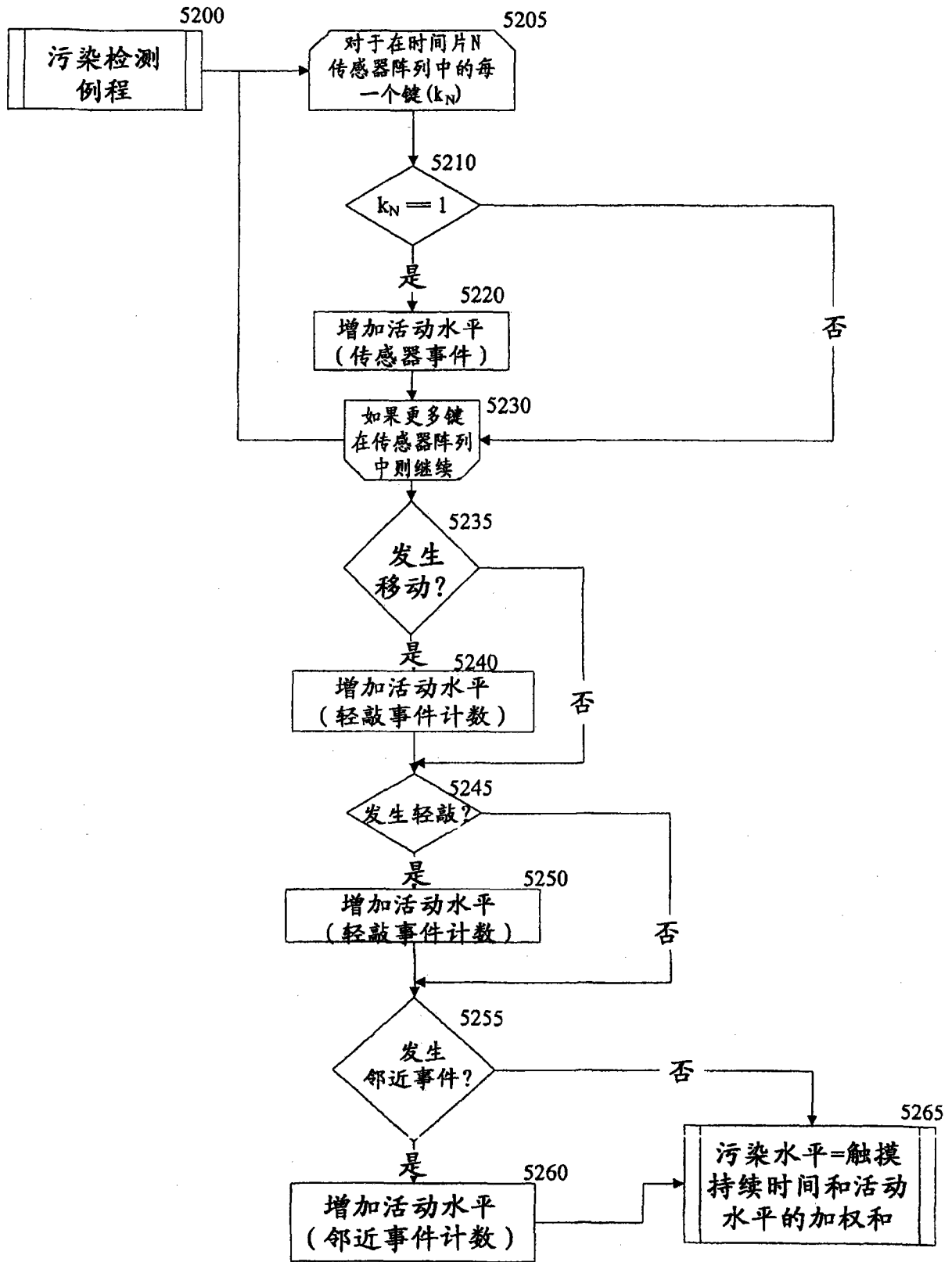


图 5B

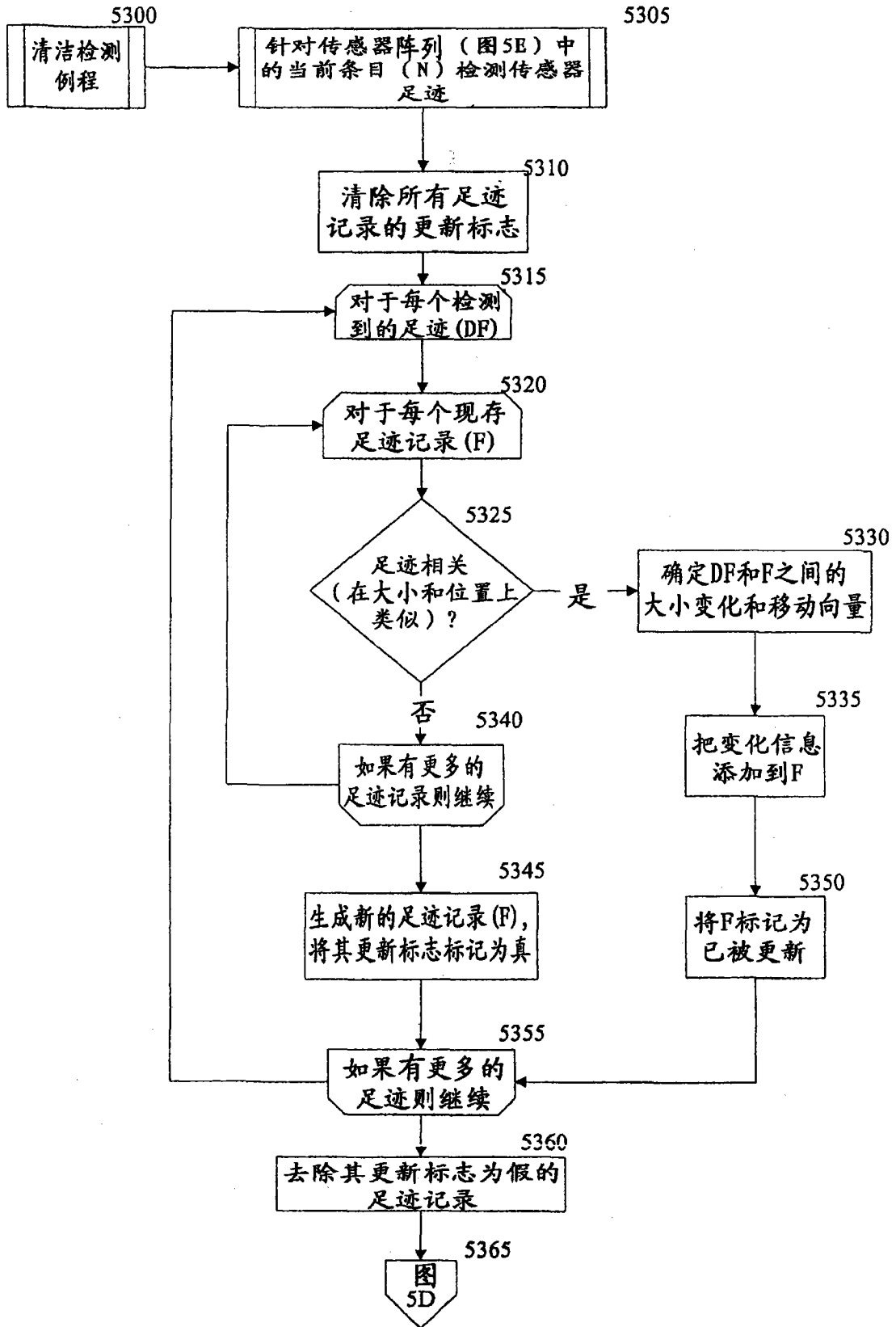


图 5C

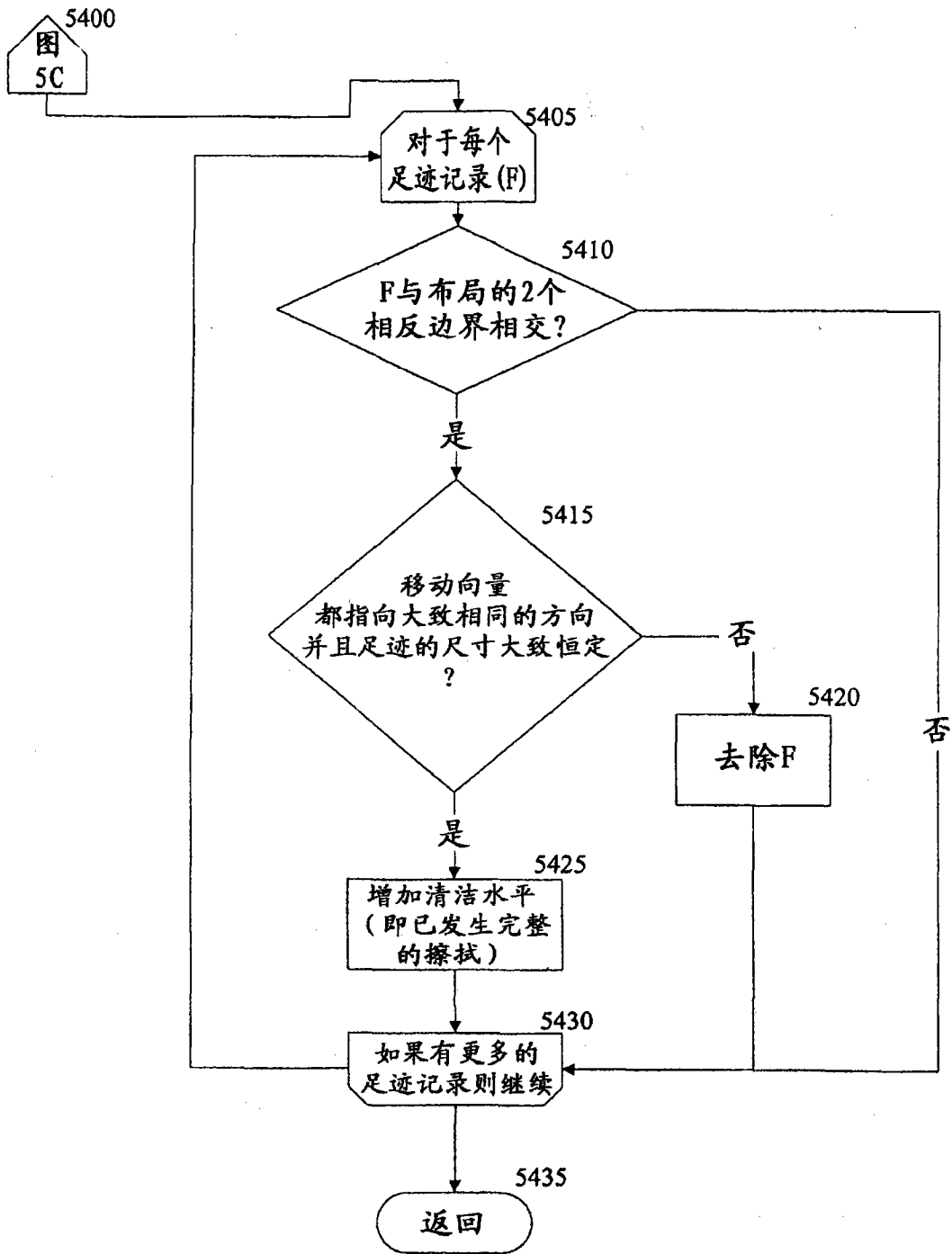


图 5D

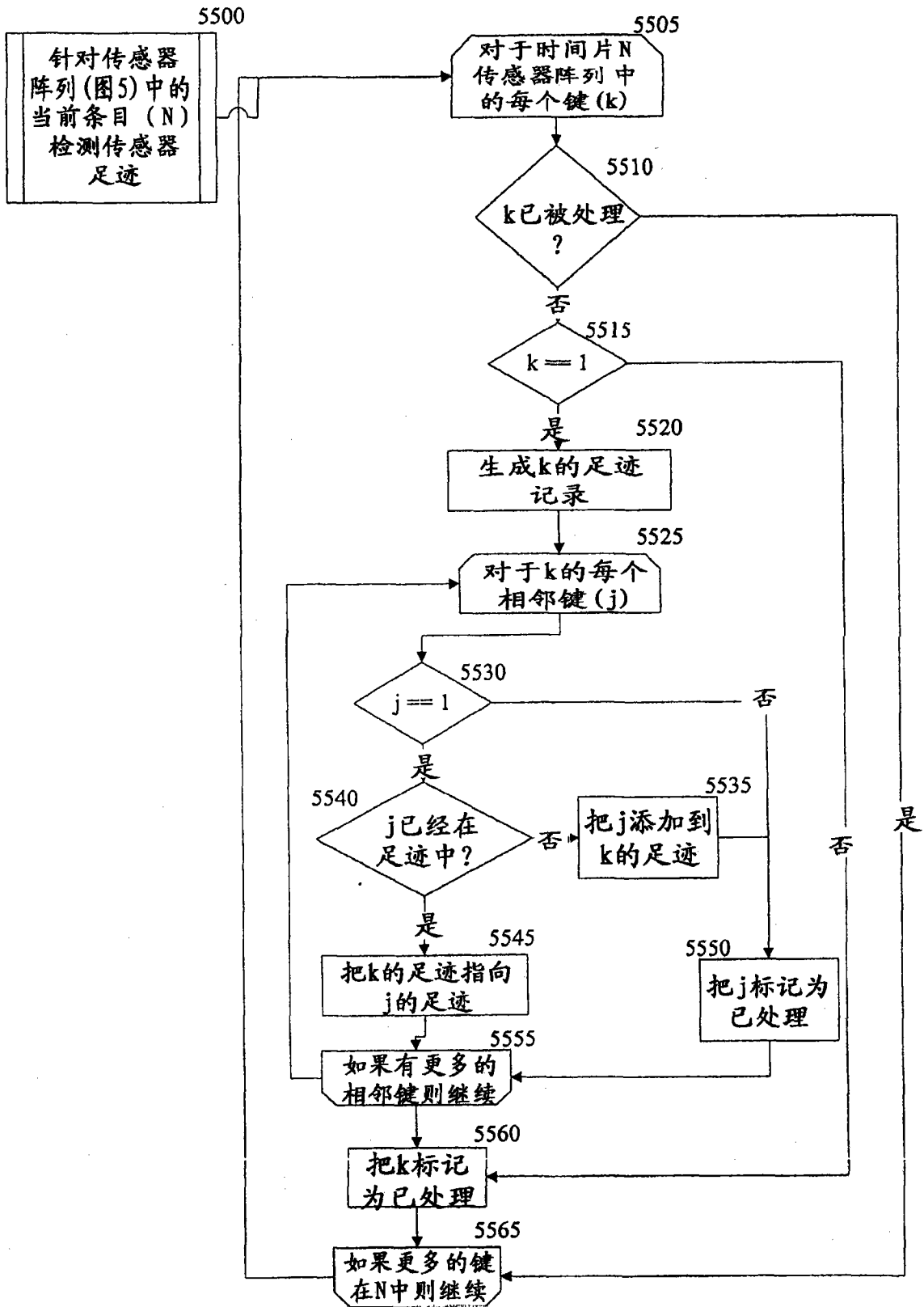


图 5E