



[12] 发明专利申请公开说明书

[21] 申请号 03138493.5

[43] 公开日 2004 年 1 月 21 日

[11] 公开号 CN 1469657A

[22] 申请日 2003.5.28 [21] 申请号 03138493.5

[30] 优先权

[32] 2002.5.31 [33] KR [31] 30555/2002

[71] 申请人 LG 电子株式会社

地址 韩国汉城

[72] 发明人 独孤世俊

[74] 专利代理机构 中原信达知识产权代理有限责
任公司

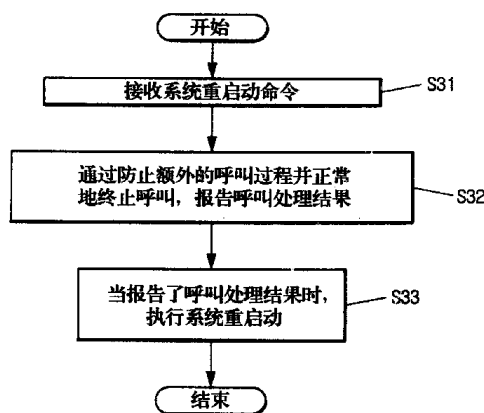
代理人 顾红霞 钟 强

权利要求书 5 页 说明书 14 页 附图 6 页

[54] 发明名称 用于管理 MSC 中的计费信息的方法和系统

[57] 摘要

一种用于管理移动交换中心(MSC)中的计费信息的方法和系统,它防止对进行中的呼叫的非正常终止及计费信息丢失,要防止的其中一种非正常终止包括当系统因为软件更换而从操作终端收到系统重新启动请求时发生的非正常终止。多个用户呼叫处理处理器中的每一个接收来自操作及维护处理器(OMP)的呼叫终止消息,且用户呼叫处理器中的主进程向用户呼叫处理器中正在处理呼叫的子进程发送该呼叫终止消息。主进程从子进程接收表明呼叫被正常终止且生成计费信息的对呼叫终止消息的响应消息,生成最终呼叫终止完成响应消息,并向OMP发送呼叫终止完成响应消息,从而实现了正常的系统重新启动程序。



1. 一种用于管理移动交换中心（MSC）中的计费信息的系统，
包括：

5 至少一个用户呼叫处理处理器，其防止额外的呼叫过程并正常地
终止进行中的呼叫；

 操作及维护处理器（OMP），通过在接收系统重启动命令时控制
用户呼叫处理处理器，正常地终止进行中的呼叫，并在接收到呼叫终
止完成响应消息时执行系统重启动程序，其中每一用户呼叫处理处理
器通过 OMP 的控制向 OMP 报告对呼叫的处理结果。

2. 如权利要求 1 的系统，其中用户呼叫处理处理器在接收到呼
叫终止消息之后切换至等待状态。

15 3. 如权利要求 1 的系统，其中用户呼叫处理处理器生成呼叫终
止完成响应消息，确定呼叫终止完成响应消息的发送点，并报告呼叫
的处理结果。

 4. 如权利要求 3 的系统，其中终止完成响应消息包括表明在用
户呼叫处理处理器中没有进行中的呼叫的信息。

 5. 如权利要求 3 的系统，其中终止完成响应消息包括表明当前
处理呼叫的进程执行了特定的计费信息管理程序的信息。

25 6. 如权利要求 3 的系统，其中呼叫终止完成响应消息的发送点
是经过一定的等待时间时的时间，该等待时间从用户呼叫处理处理器
中的主进程接收到呼叫终止消息之刻开始测量。

 7. 如权利要求 3 的系统，其中用户呼叫处理处理器中的主进程
以一定的时间间隔检查是否存在分配给当前处理呼叫的子进程的呼叫

30

寄存器，并且将不存在被分配的呼叫寄存器之刻确定为呼叫终止完成响应消息的发送点。

8. 一种用于管理 MSC 中的计费信息的方法，包括：

5 报告步骤，当接收到系统重新启动命令时，报告呼叫的处理结果；
 防止步骤，防止额外的呼叫过程，并正常终止进行中的呼叫；和
 执行步骤，当所有用户呼叫处理处理器报告了呼叫的处理结果
 时，执行系统重新启动程序。

10 9. 如权利要求 8 的方法，其中至少报告步骤和防止步骤之一包
 括：

 当接收到系统重新启动命令时，从操作和维护处理器（OMP）向用
 户呼叫处理处理器发送呼叫终止消息；

 15 确定在接收到呼叫终止消息的用户呼叫处理处理器中，是否存在
 正在处理呼叫的子进程；

 根据确定的结果引导正常的呼叫终止，并同时防止额外的呼叫过
 程；和

 在主进程中生成呼叫终止完成响应消息，确定呼叫终止完成响应
 消息的发送点，并向 OMP 发送呼叫终止完成响应消息。

20

 10. 如权利要求 8 的方法，，其中执行步骤包括：

 从多个用户呼叫处理处理器中的每一个向 OMP 发送呼叫终止完
 成响应消息；

 25 检查 OMP 是否从所有用户呼叫处理处理器接收到呼叫终止完成
 响应消息；和

 如果 OMP 从所有用户呼叫处理处理器接收到呼叫终止完成响应
 消息，则执行系统重新启动程序。

 11. 如权利要求 9 的方法，其中主进程在向子进程发送呼叫终止
30 消息之后切换至空闲状态。

12. 如权利要求 9 的方法，其中主进程确定是否存在正在处理呼叫的子进程，并根据确定结果生成呼叫终止完成响应消息。

5 13. 如权利要求 12 的方法，其中如果不存在正在处理呼叫的子进程，则主进程生成呼叫终止完成响应消息，该消息包括表明不存在进行中的呼叫的信息。

10 14. 如权利要求 12 的方法，其中如果根据确定结果，存在正在处理呼叫的子进程，则主进程生成呼叫终止完成响应消息，该消息包括表明执行了特定的计费信息管理程序的信息。

15 15. 如权利要求 9 的方法，其中呼叫终止完成响应消息包括以下消息的至少一条：表明用户呼叫处理处理器中不存在进行中的呼叫的消息，和表明正在处理呼叫的进程执行了特定的计费信息管理程序的信息。

20 16. 如权利要求 10 的方法，其中呼叫终止完成响应消息包括以下消息的至少一条：表明用户呼叫处理处理器中不存在进行中的呼叫的消息，和表明正在处理呼叫的过程执行了特定的计费信息管理程序的信息。

25 17. 如权利要求 9 的方法，其中所述在主进程生成呼叫终止完成响应消息的步骤包括：

 从主进程向正在处理呼叫的子进程发送所接收的呼叫终止消息；
 根据发送的呼叫终止消息，正常终止进行中的呼叫并生成关于呼叫的计费信息，并向主进程发送对呼叫终止消息的响应消息；和
 当主进程接收了来自所有子进程的对呼叫终止消息的响应消息时，在主进程生成呼叫终止完成响应消息，该消息包括表明完成了特定的计费信息管理程序的信息。

30

18. 如权利要求 17 的方法，其中如果呼叫是本地呼叫，则呼叫终止消息只被发送至以下子进程之一：处理呼入呼叫的子进程和处理呼出呼叫的子进程。

5

19. 如权利要求 17 的方法，其中所述从主进程向正在处理呼叫的子进程发送所接收的呼叫终止消息的步骤包括：

通过检索呼叫寄存器，提取正在处理呼叫的子进程的进程 ID；
和

10

根据提取的进程 ID，向子进程发送呼叫终止消息。

20. 如权利要求 9 的方法，其中的发送点是经过一定的等待时间时的时间，该等待时间从用户呼叫处理处理器中的主进程接收到呼叫终止消息之刻开始测量。

15

21. 如权利要求 9 的方法，其中用户呼叫处理处理器中的主进程以一定的时间间隔检查是否存在分配给当前处理呼叫的子进程的呼叫寄存器，并将不存在被分配的呼叫寄存器的时间确定为呼叫终止完成响应消息的发送点。

20

22. 如权利要求 9 的方法，其中所述向 OMP 发送呼叫终止完成响应消息的步骤包括：

接收来自正在处理呼叫的子进程的响应消息，该消息包括表明完成了特定的计费信息管理程序的信息；和

25

比较接收的响应消息的数目和向子进程发送的呼叫终止消息的数目，并且在两数相等的情况下，从主进程向 OMP 发送呼叫终止完成响应消息，该消息包括指示执行了特定的计费信息管理程序的信息。

30

23. 如权利要求 22 的方法，其中当比较结果为响应消息的数目与呼叫终止消息的数目不等时，主进程在一定的等待时间后向 OMP

发送呼叫终止完成响应消息，该消息表明完成了特定的计费信息管理程序。

用于管理 MSC 中的计费信息的方法和系统

5 技术领域

本发明涉及用于管理移动交换中心（MSC）中的计费信息的方法和系统。

背景技术

10 图 1 显示了移动交换系统的结构，其中运行现有技术的管理 MSC 计费信息的方法。该移动交换系统包括：用户呼叫处理处理器 1、干线匹配处理器 2、呼叫过程集中处理器 3、访问位置寄存器（VLR）处理处理器 4 和操作及维护处理器（OMP）5。

15 用户呼叫处理处理器 1 和干线匹配处理器 2 是主要的处理器，它们与呼叫过程集中处理器 3 和 VLR 处理处理器 4 合作，实现呼叫处理功能和呼叫连接功能。当用户呼叫处理处理器 1 和干线匹配处理器 2 接收到来自 OMP 5 的对于进行中的呼叫的呼叫终止请求时，用户呼叫处理处理器 1 和干线匹配处理器 2 生成计费信息，将计费信息发送
20 至 OMP 5，释放资源，并终止呼叫。

呼叫过程集中处理器 3 实现通用的呼叫处理功能，如数字翻译、s-交换连接和断开等，而不考虑用户呼叫处理处理器 1 执行的用户呼叫过程、干线匹配处理器 2 执行的干线匹配等。

25

VLR 处理处理器 4 通过与不包含在移动交换系统中的归属位置寄存器（HLR）合作，请求特定区域内用户的用户位置寄存器信息及用户信息，并临时存储从 HLR 接收的用户位置寄存器信息和用户信息。

30

OMP 5 执行移动交换系统中的特定操作和维护、系统启动/条件管理等，接收用户呼叫处理处理器 1 和干线匹配处理器 2 发送的计费信息，将计费信息存入硬盘 6 中，并控制是否提取硬盘 6 中存储的计费信息。

5

当操作者因为软件更换等原因，请求移动交换系统重新启动时，操作者终端向 OMP 5 发送重新启动命令。OMP 5 完成 OMP 5 的重新启动，并控制其它处理器（用户呼叫处理处理器 1、干线匹配处理器 2、呼叫过程集中处理器 3、VLR 处理处理器 4 等）实现自身的重新启动。对于处理器的重新启动，软件模块，如处理器加载程序（PL）、系统启动加载程序（SSL）等，执行存在于处理器中的各处理器的重新启动。

10

PL 从硬盘 6 依次接收要操作的应用软件模块，并被载入除 OMP 5 以外的其它处理器。

15

SSL 从载入其它处理器的 PL 依次接收应用软件模块请求，并通过依次从硬盘 6 接收应用软件模块而被载入 OMP 5。

20

接下来，将描述重新启动系统在移动交换中心中的管理计费信息的现有方法。尽管操作者既可能命令整个系统重新启动又命令特定的软件模块重新启动，但这里假设本发明中操作者可能命令整个系统重新启动。

25

从操作者终端接收到重新启动命令的 OMP 5 生成呼叫终止消息，并通过向各处理器发送呼叫终止消息，命令各处理器终止所有进行中的操作。各处理器在接收到来自 OMP 5 的呼叫终止消息后，终止所有进行中的操作。而且，操作系统使用引导程序执行重新启动。当重新启动完成时，PL 被载入各个处理器，且 SSL 被同时载入 OMP 5。也就是说，当系统重新启动时，PL 通过依次接收来自硬盘 6 的应用软件模块，被载入各处理器；SSL 通过依次接收来自硬盘 6 的应用软件模块，被

30

载入 OMP 5。

5 由于 OMP 5 与硬盘 6 直接相连，且其它处理器通过 OMP 5 与硬盘 6 相连，所以可以通过 OMP 5 的重启动，在 SSL 被载入 OMP 5 之后，执行由引导程序对系统的重启动。也就是说，当重启动请求发生时，各处理器终止所有进行中的操作，尝试由引导程序进行的重启动程序，通过向 OMP 5 的 SSL 发送特定消息接收软件模块，并将 PL 载入各处理器。

10 移动交换系统继续进行许多呼出/呼入尝试、为支持呼叫连接状态而分配并释放资源、请求/响应/更新相关信息等。

15 如果在移动交换系统的持续操作期间，用户输入由于软件更换的需要而引起的重启动命令，则移动交换系统立即切换至初始状态，在这种状态中忽视进行中的呼叫。因此，当前在不同状态中进行的呼叫被非正常结束，且资源丢失。

20 当这种情况发生时，存在这样的问题：所有关于呼叫的记录因非正常结束而丢失。而这些记录包括重要信息，不仅涉及有关运营商利润的计费信息，而且涉及民众要求的用途。

发明内容

本发明的目标是解决现有技术的系统和方法的上述问题。

25 本发明的另一个目标是防止由于软件更换而引起的进行中的呼叫非正常终止以及计费信息丢失，具体而言，通过稳定地重启系统而防止非正常终止。

30 为了达到本发明的这个和其它目标及优点，提供用于管理移动交换中心中的计费信息的方法和系统，其通过正常终止适当地释放资

源，生成并记录计费信息，并切换至空闲状态，从而不处理额外的呼叫呼出/呼入请求。

5 根据一个实施例，用于管理移动交换中心（MSC）中的计费信息的系统包括：操作及维护处理器（OMP），通过在接收到系统重新启动命令时控制各用户呼叫处理处理器并在接收到呼叫终止完成响应消息时实现系统重新启动程序，正常地终止进行中的呼叫；和若干用户呼叫处理处理器，通过 OMP 的控制，防止额外的呼叫过程，正常地终止进行中的呼叫，并向 OMP 报告呼叫的处理结果。

10

优选地，用于管理 MSC 中的计费信息的系统的用户呼叫处理处理器在接收到呼叫终止消息后，切换至等待状态。

15 优选地，用于管理 MSC 中的计费信息的系统的用户呼叫处理处理器生成呼叫终止完成响应消息，确定终止呼叫终止完成响应消息的发送点，并报告呼叫的处理结果。

优选地，用于管理 MSC 中的计费信息的系统的终止完成响应消息包括表明用户呼叫处理处理器中没有正在进行的呼叫的信息。

20

优选地，用于管理 MSC 中的计费信息的系统的终止完成响应消息包括表明特定的计费信息管理程序由当前处理呼叫的进程执行的信息。

25

优选地，在用于管理 MSC 中的计费信息的系统处，呼叫终止完成响应消息的发送点是经过一定的等待时间时的时间，该等待时间从用户呼叫处理处理器中的主进程接收到呼叫终止消息之刻开始测量。

30 优选地，用户呼叫处理处理器中的主进程以一定的时间间隔检查分配给当前处理呼叫的子进程的呼叫寄存器是否存在，从而将用于管

理 MSC 中的计费信息的系统的呼叫终止完成响应消息的发送点确定为被分配的呼叫寄存器不存在的时间。

根据本发明的优选实施例的管理 MSC 中计费信息的方法包括：

- 5 报告步骤，在通过防止额外呼叫处理并正常终止进行中的呼叫而接收系统重启命令时，报告呼叫的处理结果；和执行步骤，当所有用户呼叫处理处理器报告了呼叫的处理结果时，执行系统重启动程序。

10 优选地，所述的在通过防止额外呼叫处理并正常终止进行中的呼叫而接收系统重启命令时报告呼叫处理结果的报告步骤包括：当接收到系统重启命令时，从操作及维护处理器（OMP）向各用户呼叫处理处理器发送呼叫终止消息；由接收到呼叫终止消息的用户呼叫处理处理器中的主进程确定是否存在当前处理呼叫的子进程，根据所确定的结果引导正常的呼叫终止并同时防止额外的呼叫过程；以及在主进程中生成呼叫终止完成响应消息，确定该呼叫终止完成响应消息的发送点，并将该呼叫终止完成响应消息发送至 OMP。

20 优选地，所述当所有用户呼叫处理处理器报告了呼叫的处理结果时执行系统重启动程序的执行步骤包括：从各用户呼叫处理处理器向 OMP 发送呼叫终止完成响应消息；以及检查 OMP 是否接收到来自所有用户呼叫处理处理器的呼叫终止完成响应消息，并在 OMP 接收到来自所有用户呼叫处理处理器的呼叫终止完成响应消息的情况下执行系统重启动程序。

25 优选地，主进程在向子进程发送呼叫终止消息之后切换至空闲状态。

优选地，主进程确定是否存在正在处理呼叫的子进程，以根据所确定的结果生成呼叫终止完成响应消息。

30

优选地，在不存在正在处理呼叫的子进程的情况下，主进程生成的呼叫终止完成报告消息包含指示不存在正在进行的呼叫的信息。

5 优选地，在根据所确定的结果，存在正在处理呼叫的子进程的情况下，主进程生成的呼叫终止完成报告消息包含指示执行了特定的计费信息管理程序的信息。

10 优选地，呼叫终止完成响应消息包括以下至少一条信息：指示用户呼叫处理处理器中不存在进行中的呼叫的信息、指示在当前处理呼叫的进程中执行了特定的计费信息管理程序的信息。

15 优选地，所述在主进程中生成呼叫终止完成响应消息的步骤包括：从主进程向当前处理呼叫的子进程发送接收的呼叫终止消息；根据发送的呼叫终止消息，正常地终止进行中的呼叫并生产有关呼叫的计费信息，并将对呼叫终止消息的响应消息发送至主进程；和生成呼叫终止完成响应消息，该消息包括指示当主进程从所有子进程接收到对呼叫终止消息的响应消息时，主进程执行了特定的计费信息管理程序的信息。

20 优选地，如果呼叫是本地呼叫，则呼叫终止消息只发送至处理呼入呼叫的子进程和处理呼出呼叫的子进程之一。

25 优选地，所述从主进程向当前处理呼叫的子进程发送接收的呼叫终止消息的步骤包括：通过检索呼叫寄存器，提取当前处理呼叫的子进程的进程 IDs；并将呼叫终止消息发送至对应于所提取的进程 IDs 的子进程。

30 优选地，发送点是当特定的等待时间经过时的时间，该等待时间是从用户呼叫处理处理器中的主进程接收到呼叫终止消息的时间开始测量的。

5 优选地，用户呼叫处理处理器中的主进程以一定的时间间隔检查是否存在分配给当前处理呼叫的子进程的呼叫寄存器，从而将根据用于管理移动交换中心中计费信息的方法的呼叫终止完成响应消息的发送点确定为不存在已分配的呼叫寄存器的时刻。

10 优选地，所述向 OMP 发送呼叫终止完成响应消息的步骤包括：从当前处理呼叫的子进程接收响应消息，该消息包括指示执行了特定的计费信息管理程序的信息；和比较接收到的响应消息的数目与向子进程发送的呼叫终止消息的数目，并且在两数相等的情况下，从主进程向 OMP 发送呼叫终止完成响应消息，该消息包括指示执行了特定的计费信息管理程序的信息。

15 优选地，在比较结果显示响应消息的数目与呼叫终止消息的数目不等的情况下，在一定等待时间后向 OMP 发送呼叫终止完成响应消息，该消息指示执行了特定的计费信息管理程序。

20 本发明其它的优点、目标和特征将在后面的说明书中部分介绍，并且对于本领域技术人员，通过本发明的实际操作可以进一步了解到它们。本发明的目标和优点可以以所附权利要求中特别指出的方式实现并得到。

附图简述

25 下面参考附图详细介绍本发明，其中相似的数字代表类似的元件，附图中：

图 1 显示了应用现有的管理 MSC 中计费信息的方法的 MSC 的结构；

图 2 显示了根据本发明的一个优选实施例，管理 MSC 中计费信息的系统的结构；

30 图 3 是显示根据本发明的一个优选实施例的管理 MSC 中计费信

息的方法的流程图；

图 4 是显示根据本发明的一个优选实施例的 OMP 的操作的流程图；

5 图 5 是显示根据本发明的一个优选实施例，用户呼叫处理处理器的操作的流程图；

图 6 是显示根据本发明的一个优选实施例，向子进程发送呼叫终止消息的过程的流程图；

图 7 是显示根据本发明的一个优选实施例，发送呼叫终止完成响应消息的过程的流程图。

10

具体实施方式

根据本发明的一个实施例涉及一个预处理程序，其生成并记录由于非正常终止而丢失的呼叫的计费信息。当发生这种情况时，优选地执行向空闲状态的切换，通过在重启动程序前执行该切换而防止额外的呼入呼叫。下面介绍本发明的实施例。

15

如图 2 所示，根据本发明的一个优选实施例的系统包括用户呼叫处理处理器 10 和 OMP 50。处理器中的软件模块优选地用来操作该系统。

20

每一个用户呼叫处理处理器 10 执行主要的呼叫处理功能，如呼叫连接、资源管理、计费信息生成等。如图 3 所示，如果用户呼叫处理处理器 10 接收到来自 OMP 50 的系统重启动命令 (S31)，则用户呼叫处理处理器向用户呼叫处理处理器的主进程发送呼叫终止消息。然后主进程将呼叫终止消息发送至一个或多个子进程。

25

如果主进程接收到来自子进程的对于呼叫终止消息的响应消息，该消息表明所有进行中的呼叫被终止且呼叫的计费信息已生成，则主进程生成呼叫终止完成响应消息，确定呼叫终止完成响应消息的发送点，并将呼叫终止完成响应消息发送至 OMP 50。也就是说，从 OMP 50

30

接收到系统重新启动命令的用户呼叫处理处理器 10 阻止额外的呼叫过程，正常地终止进行中的呼叫，并再次向 OMP 50 报告对进行中的呼叫的处理结果（S32）。

5 根据一个实施例，只有在所有用户呼叫处理处理器 10 向 OMP 50 报告了处理呼叫的结果时，OMP 50 才执行系统重新启动程序（S33）。也就是说，如果 OMP 50 接收到来自操作者终端的系统重新启动命令，则 OMP 50 向用户呼叫处理处理器 10 发送呼叫终止消息。而且，OMP 50 通过接收来自用户呼叫处理处理器 10 的对呼叫终止消息的呼叫终止完成响应消息，执行计费信息管理及系统重新启动程序。

15 此时，用户呼叫处理处理器 10 通常涉及至少在呼出/呼入侧之一的所有呼叫，但是干线匹配处理器涉及对应呼叫分类的一些呼叫。因此，为所有呼叫而操作的用户呼叫处理处理器 10 控制呼叫处理程序模块，并正常地终止许多进行中的呼叫。

 首先解释移动用户的呼出/呼入呼叫处理顺序，然后解释根据系统重新启动方法，在 OMP 50 和用户呼叫处理处理器 10 之间的操作。

20 如果发生了移动用户呼出呼叫，则用户呼叫处理处理器 10 中的主进程生成子进程，并立即切换至等待状态，等待另一呼叫。向子进程分配一个呼叫寄存器，且子进程继续进行呼入呼叫连接的程序，诸如翻译呼叫用户输入的呼入数字、地址检索（retrieval）等。

25 如果子进程接收到呼入响应消息，则子进程切换至忙状态，并存储收费开始时间。而且，如果接收到主叫用户终止消息或被叫用户终止消息，则子进程返回资源和呼叫寄存器，并生成计费信息。

30 呼叫寄存器是用于存储在呼叫处理程序期间发生的各种类型信息的空间。呼叫寄存器存在于公共存储区中，公共存储区在请求呼叫时

被分配且在终止呼叫时被返回，且用户呼叫处理处理器 10 访问呼叫寄存器。呼叫寄存器存储如子进程的数目、正在处理呼叫的子进程的 ID 列表、空闲子进程的 ID 列表等信息。

5 如果发生了对于移动用户的呼入呼叫，则消息处理模块接收到呼入呼叫请求消息后，执行呼入寻呼操作。在从移动用户的手机发送了呼入响应消息之后，消息处理模块请求一个主进程和一个服务模块，以执行呼入呼叫过程。

10 然后，主进程为呼入呼叫过程生成子进程，并切换至等待状态。子进程将信息存储入分配的呼叫寄存器中，并继续进行与呼出处理器呼叫连接的程序。

15 如果从被叫用户或主叫用户的手机发送呼叫终止信号，则主进程释放所有资源，根据运营商选择性地生成计费信息，向 OMP 50 发送计费信息，并终止呼叫。

20 主进程接收在用户呼叫处理处理器 10 中的呼出和呼入呼叫请求，并令子进程继续各自独立进行呼叫过程。各呼叫处理程序的呼叫处理信息（如交换信息、被叫用户数、主叫用户数、位置信息、呼叫处理程序的当前状态等）由子进程记录到呼叫寄存器中。因此，主进程通过检索呼叫寄存器、公共存储区，可以得到如用户呼叫处理处理器中的子进程数目、子进程地址、每个呼叫的当前状态等信息。

25 图 4 和图 5 显示了根据本发明的一个优选实施例的用于管理 MSC 中的计费信息的方法。这些图分别显示了 OMP 以及用户呼叫处理处理器的操作。

30 OMP 50 接收来自操作者终端的重启动命令（S41），通过解释重启动命令而生成呼叫终止消息，并向每个用户呼叫处理处理器 10 发

送呼叫终止消息（S42）。

OMP 50 切换至等待状态，直至从用户呼叫处理处理器 10 接收到对呼叫终止消息的呼叫终止完成响应消息（S43）。OMP 处于等待状态的原因是防止除进行中的呼叫外的另一呼叫呼入。

如果接收到了来自所有用户呼叫处理处理器 10 的呼叫终止完成响应消息，则系统重新启动（S44，S45）。

另一方面，用户呼叫处理处理器 10 中从 OMP 50 接收到呼叫终止消息的主进程切换至等待状态（S52），并等待来自子进程的响应，而不执行额外的呼叫过程。

主进程检索呼叫寄存器（S53），并确定是否存在正在处理呼叫的子进程（S54）。

如果不存在正在处理呼叫的子进程，则意味着不存在进行中的呼叫，因此主进程向 OMP 50 发送表明不存在进行中的呼叫的呼叫终止完成响应消息（S59）。

另一方面，如果检索呼叫寄存器的结果是存在正在处理呼叫的子进程，则主进程从呼叫寄存器中提取正在处理呼叫的子进程的 IDs，并根据提取的进程 ID 向子进程发送呼叫终止消息（S55）。

由于向子进程发送消息需要子进程的进程 ID，所以主进程要通过在呼叫寄存器中搜索目标进程 ID，向子进程发送呼叫终止消息。

而且子进程在接收到呼叫终止消息之后终止进行中的呼叫，生成计费消息，并生成对呼叫终止消息的响应消息。切换至空闲状态的主进程接收响应消息（S56）。

主进程确定是否从所有子进程接收了对呼叫终止消息的响应消息 (S57)。在从所有子进程接收了响应消息的情况下，主进程生成包括表明完成了特定计费信息管理程序的信息的呼叫终止完成响应消息，
5 确定呼叫终止完成响应消息的发送点，并向 OMP 50 发送呼叫终止完成响应消息 (S58, S59)。

主进程可以根据多种方法确定呼叫终止完成响应消息的发送点，该呼叫终止完成响应消息表明完成了特定计费信息管理程序。确定呼
10 叫终止完成响应消息的方法例如：如果从主进程接收呼叫终止消息的时刻开始，经过了一定的等待时间，则发送呼叫终止完成响应消息；以一定的时间间隔检查是否存在分配给正在处理呼叫的子进程的呼叫寄存器，在结果为不存在被分配的呼叫寄存器时发送呼叫终止完成响应消息。
15

通过用实验方法计算所有子进程完成特定的计费信息管理程序通常使用的时间，来确定特定的等待时间。通过考虑所有子进程完成特定的计费信息管理程序通常使用的时间，来确定该特定时间的间隔。

20 当各子进程接收到呼叫终止消息之后正常终止呼叫时发生的计费信息也被发送到 OMP 50 的计费处理模块，从而存储有关被终止的呼叫的记录。

等待状态中的 OMP 50 检查是否从所有用户呼叫处理处理器 10
25 接收到呼叫终止完成响应消息 (S44)。如果从所有用户呼叫处理处理器 10 接收了呼叫终止完成响应消息，则系统重新启动 (S45)。

参考图 6 和图 7 详细解释在主进程和子进程之间发送消息的操作。

30

从 OMP 50 接收到呼叫终止消息的用户呼叫处理处理器 10 中的主进程检索呼叫寄存器 (S61)、提取当前处理呼叫的子进程的进程 ID (S62)，并向子进程发送呼叫终止消息 (S63)。

5 主进程可以通过呼叫寄存器信息检索呼叫类型。由于如果呼叫是本地呼叫，则处理呼入呼叫的子进程和处理呼出呼叫的子进程都存在于用户呼叫处理处理器 10 中，所以呼叫终止消息只被发送到处理呼出呼叫的子进程，以避免重复。

10 从子进程接收到对呼叫终止消息的响应消息的主进程将接收的响应消息数和向子进程发送的呼叫终止消息数进行比较 (S71, S72)。

如果比较的结果是响应消息数与呼叫终止消息数相等，则确定所有子进程被正常终止，从而主进程生成表明完成了特定计费信息管理程序的呼叫终止完成响应消息，并向 OMP 50 发送呼叫终止完成响应消息 (S73, S74)。

相反，如果比较结果是响应消息数与呼叫终止消息数不等，则主进程在一定的等待时间后，向 OMP 50 发送表明完成了特定的计费信息管理程序的呼叫终止完成响应消息 (S75)。

在所述一定的等待时间期间，只要主进程接收到来自子进程的对呼叫终止消息的响应消息，主进程就重复执行上述过程 (S71~S73)。

25 主进程等待特定的等待时间，以从子进程接收对呼叫终止消息的响应消息，而且主进程通过在一一定的等待时间后，发送最终呼叫终止完成响应消息，强制终止仍不正常完成的呼叫。通过用实验方法计算在子进程发送对呼叫终止消息的响应消息通常所用的时间，来确定等待时间。

30

5 如上所述，由于移动交换系统中同时以多种状态（例如对于许多用户
提供处于忙状态的呼叫，用于提供呼叫连接的程序在进行中，或在呼
10 叫完成之后执行呼叫终止程序）执行呼叫处理程序，所以现有技术
的方法强制重新启动系统，造成许多强制终止的呼叫的计费信息丢
失，而该计费信息是对运营商和用户都很重要的信息。本发明的用于
管理移动交换中心中的计费信息的方法和系统存储大量可能作为计费
信息而丢失的信息，并防止额外的呼叫处理程序，从而对所有进行中
的呼叫发送不受非正常终止程序支配而根据正常终止程序的消息。因
15 此，本发明的用于管理移动交换中心中的计费信息的方法和系统可以
适当地释放资源并将非正常终止最小化。进一步，本发明的用于管理
移动交换中心中的计费信息的方法和系统可以有效地用于发展更多种
的服务及改进操作中的问题的应用软件的应用。

15 前面的实施例和优点只是示例性的，而不是构造为本发明的限定。
提出的指导可以容易地应用于其它类型的装置。本发明的说明书是
描述性的，不限制权利要求的范围。本领域技术人员容易理解它的
许多改进、修改及变化。

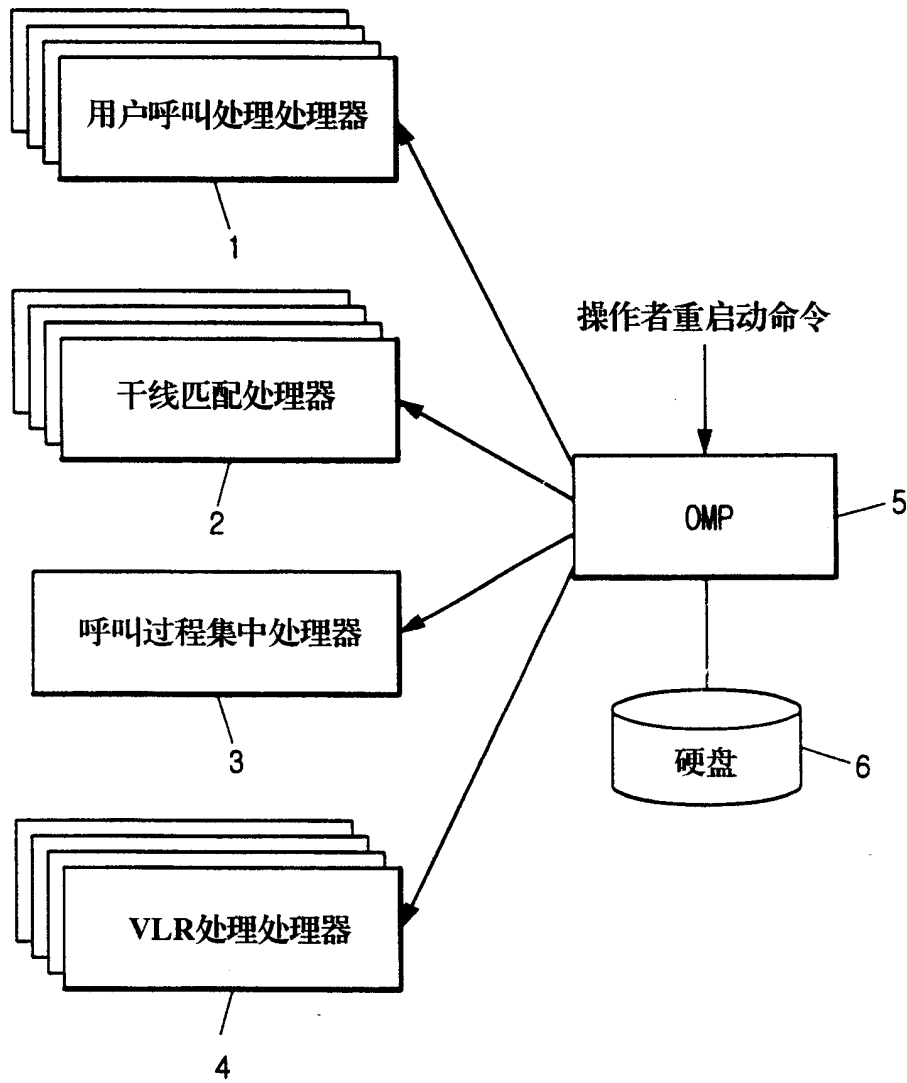


图 1

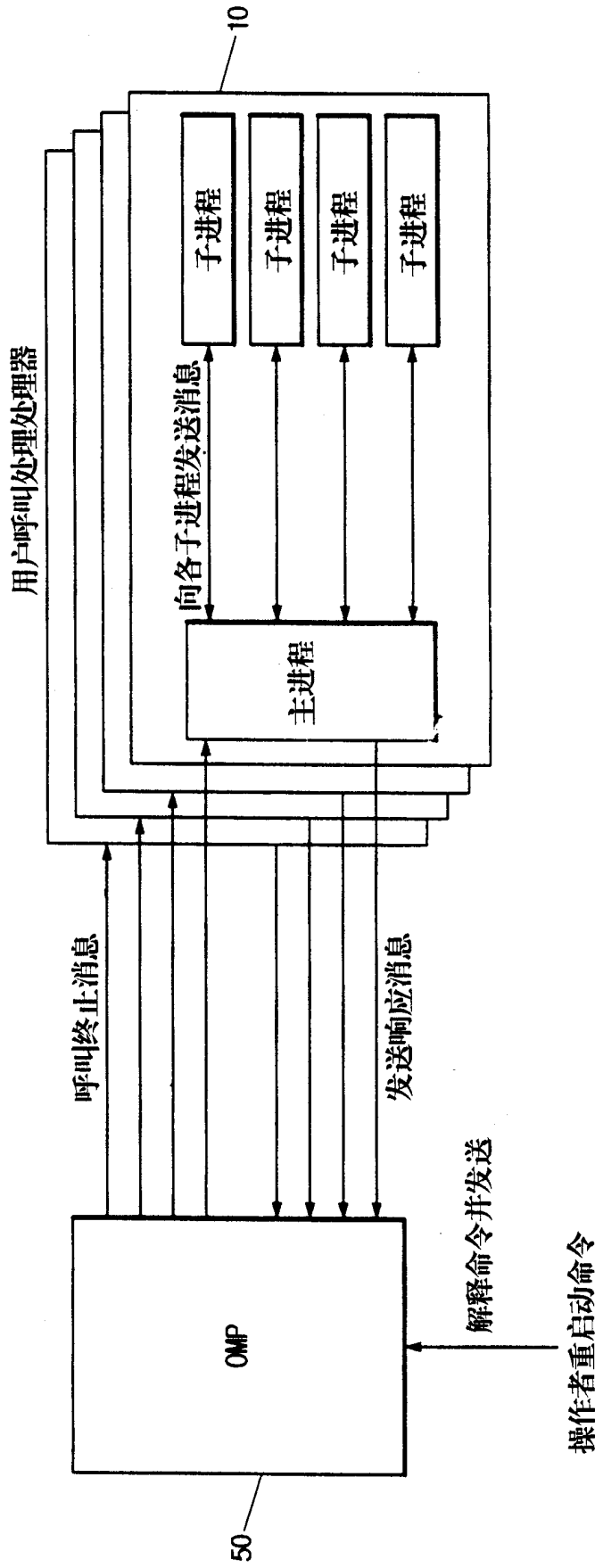


图 2

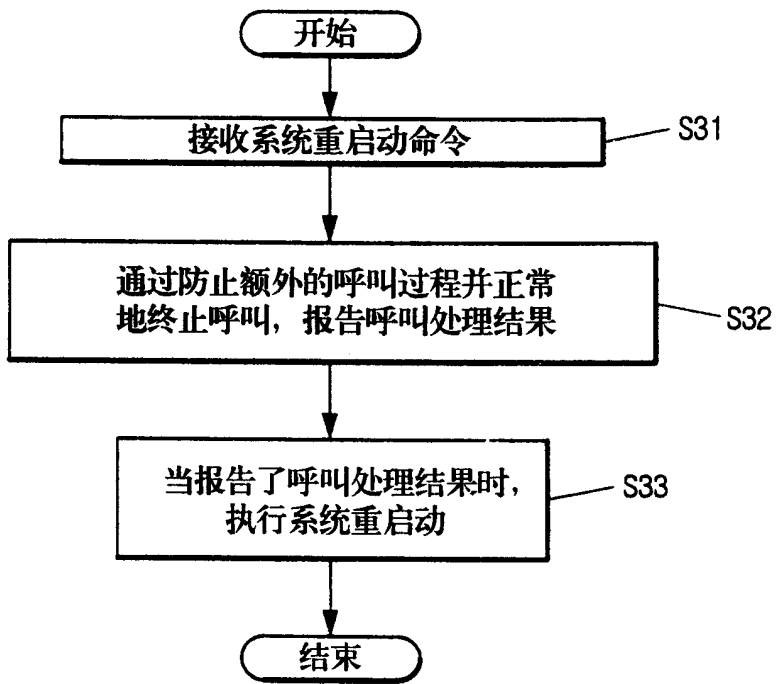


图 3

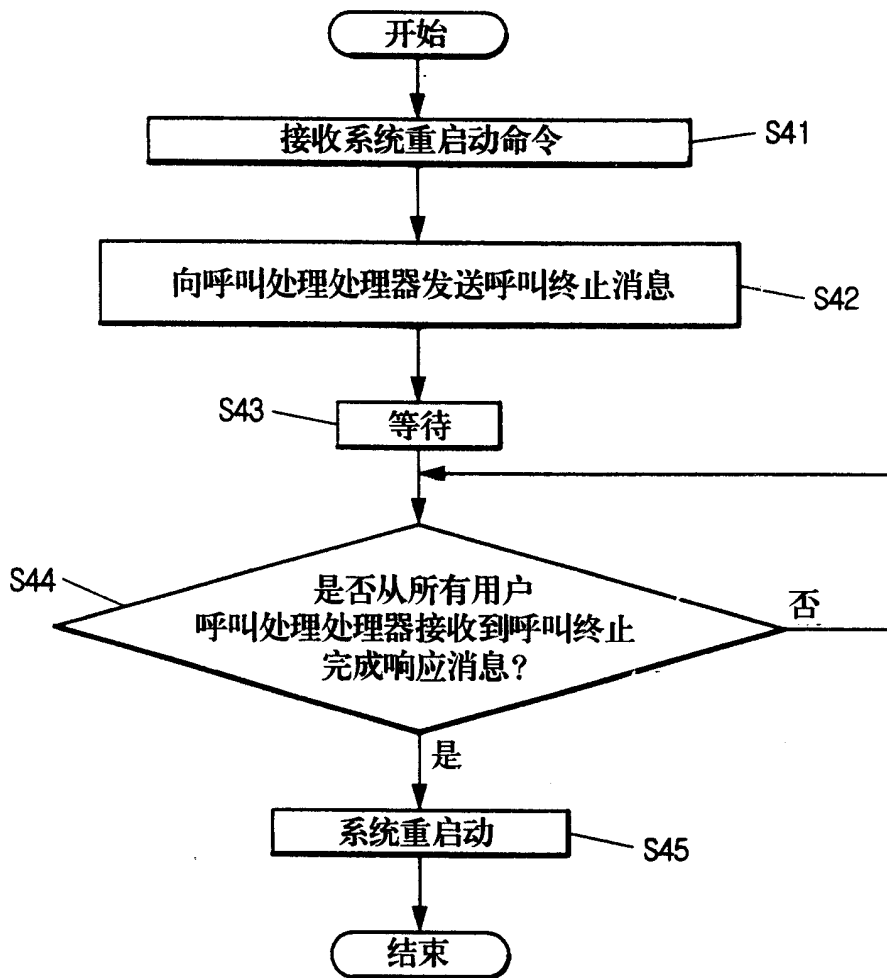


图 4

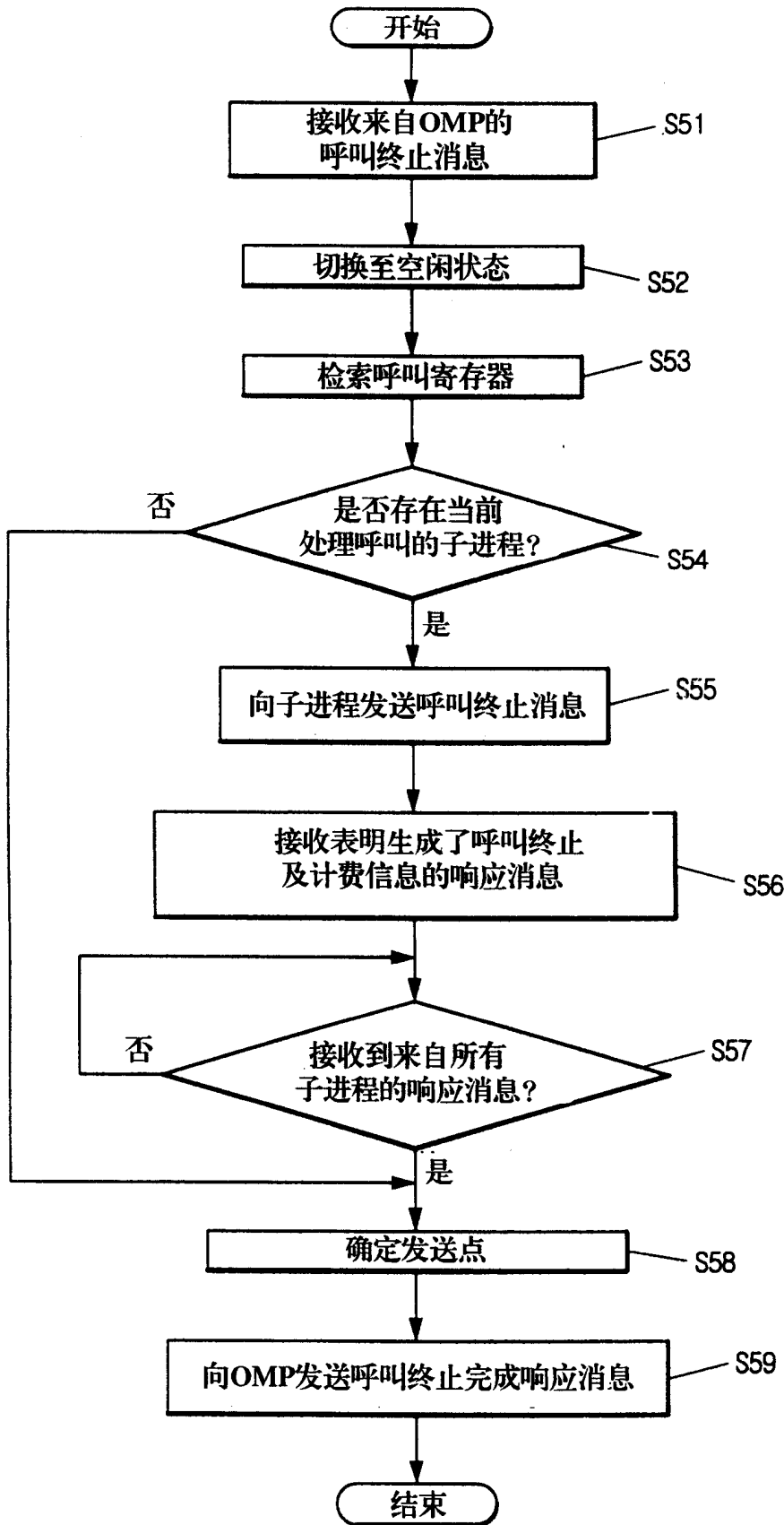


图 5

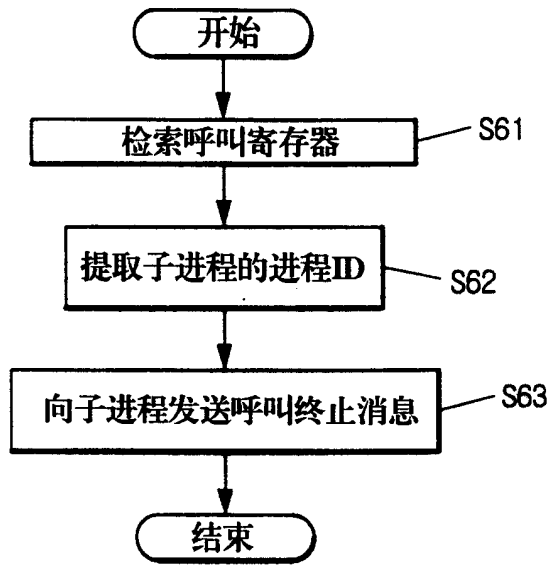


图 6

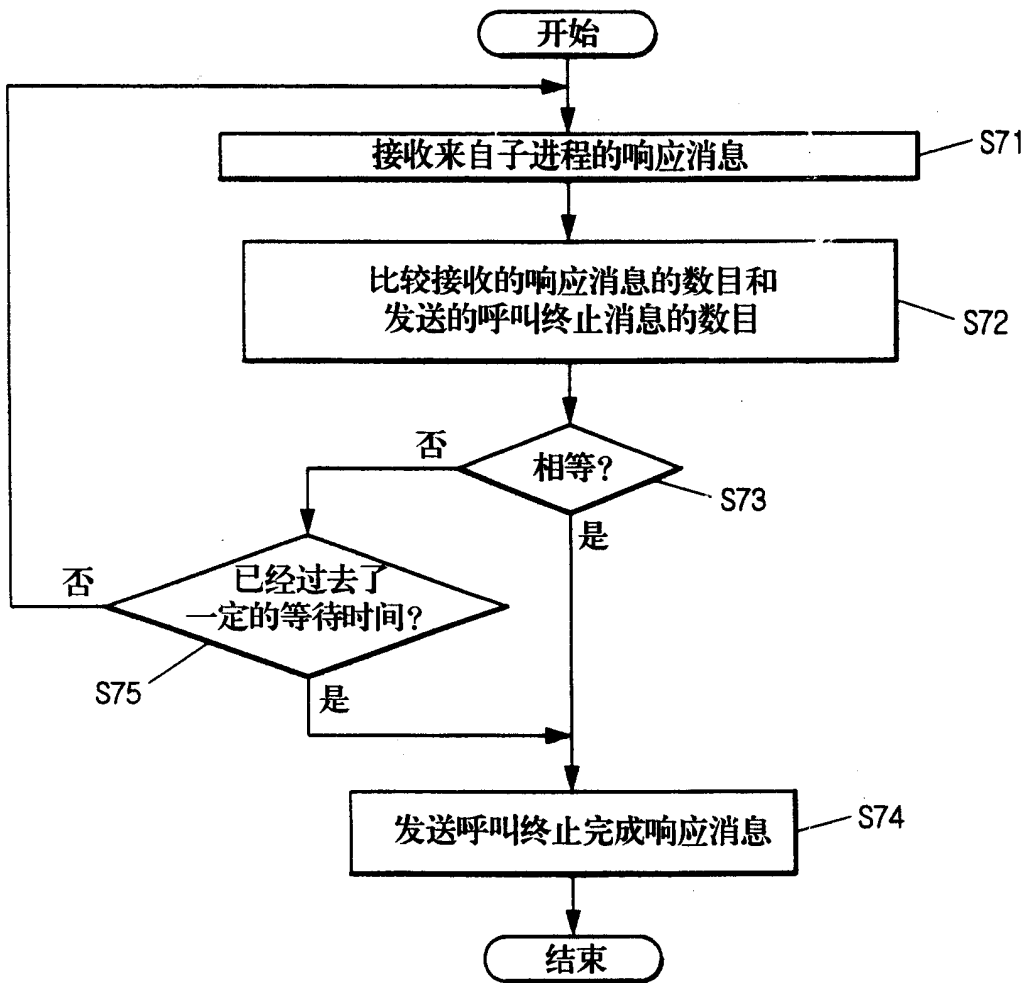


图 7