

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04N 1/32 (2006.01)

G03G 21/14 (2006.01)



[12] 发明专利说明书

专利号 ZL 200610121463.9

[45] 授权公告日 2009年4月1日

[11] 授权公告号 CN 100474883C

[22] 申请日 2003.9.15

[21] 申请号 200610121463.9

分案原申请号 03164873.8

[30] 优先权

[32] 2002.9.13 [33] JP [31] 269280/02

[32] 2002.9.13 [33] JP [31] 269281/02

[32] 2002.9.13 [33] JP [31] 269282/02

[32] 2002.9.14 [33] JP [31] 307636/02

[32] 2002.9.14 [33] JP [31] 307637/02

[32] 2003.9.9 [33] JP [31] 317280/03

[32] 2003.9.9 [33] JP [31] 317281/03

[32] 2003.9.9 [33] JP [31] 317282/03

[32] 2003.9.9 [33] JP [31] 317283/03

[73] 专利权人 株式会社理光

地址 日本东京都

[72] 发明人 秋吉邦洋 田中浩行 安藤光男

[56] 参考文献

US 5678135A 1997.10.14

CN 1321933A 2001.11.14

CN 1236344A 1999.11.24

CN 1317768A 2001.10.17

CN 1038712A 1990.1.10

审查员 贾 允

[74] 专利代理机构 北京市柳沈律师事务所

代理人 郭定辉 吕晓章

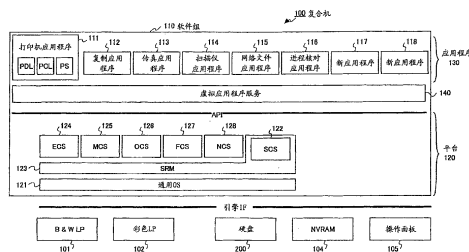
权利要求书 3 页 说明书 37 页 附图 80 页

[54] 发明名称

成像装置及其方法

[57] 摘要

一种成像装置及其方法和程序，该成像装置包括用于显示或输入操作信息的操作面板，所述成像装置包括：信息获得部件，用于获得有关将在所述成像装置中使用的应用程序的应用程序信息；以及显示部件，用于在所述操作面板上显示所述应用程序信息或与所述应用程序信息有关的信息。



1、一种成像装置，包括用于显示或输入操作信息的操作面板，所述成像装置包括：

 信息获得部件，用于获得有关将在所述成像装置中使用的应用程序的应用程序信息；

 显示部件，用于在所述操作面板上显示所述应用程序信息或与所述应用程序信息有关的信息；和

 确定部件，用于根据由所述信息获得部件获得的所述应用程序信息，确定是否可在所述成像装置中安装所述应用程序。

2、如权利要求1所述的成像装置，其特征在于，根据所述确定部件的确定结果，所述显示部件在所述操作面板上显示一个或多个可安装应用程序，其中能从所述一个或多个可安装应用程序中选择可安装应用程序。

3、如权利要求2所述的成像装置，其特征在于，所述显示部件显示用于所述一个或多个应用程序的包括应用程序名和应用程序概述的信息。

4、如权利要求1所述的成像装置，其特征在于，根据所述确定部件的确定结果，所述显示部件显示不可安装应用程序的原因。

5、如权利要求1所述的成像装置，其特征在于，当已经在所述成像装置中安装了被选择用于安装的应用程序时，所述显示部件显示警告。

6、如权利要求1所述的成像装置，其特征在于，在所述显示面板上显示的所述应用程序信息或与所述应用程序信息有关的信息包括用于确定是否可安装所述应用程序所需的信息。

7、如权利要求6所述的成像装置，其特征在于，用于确定是否可安装所述应用程序所需的所述信息包括所述应用程序的版本信息以及所述成像装置的版本信息。

8、如权利要求6所述的成像装置，其特征在于，用于确定是否可安装所述应用程序所需的所述信息包括所述应用程序的所需资源信息以及所述成像装置的可用资源信息。

9、如权利要求1所述的成像装置，其特征在于，所述显示部件通过使用所述应用程序信息显示安装操作屏或卸载操作屏，

 所述成像装置进一步包括：

用于根据所述安装操作屏或所述卸载操作屏上的操作，执行用于所述应用程序的安装进程或卸载进程的部件。

10、如权利要求 1 所述的成像装置，其特征在于，所述成像装置进一步包括用于使远程装置显示与所述操作面板上显示上的屏幕相同的屏幕的部件。

11、一种用于在成像装置中的操作面板上显示操作信息的方法，所述成像装置包括：

获得有关将在所述成像装置中使用的应用程序的应用程序信息的信息获得步骤；

在所述操作面板上显示所述应用程序信息或与所述应用程序信息有关的信息的显示步骤；以及

根据所述应用程序信息，确定是否可在所述成像装置中安装所述应用程序的确定步骤。

12、如权利要求 11 所述的方法，其特征在于，根据所述确定步骤的结果，所述成像装置在所述操作面板上显示一个或多个可安装应用程序，其中能从所述一个或多个可安装应用程序选择可安装应用程序。

13、如权利要求 12 所述的方法，其特征在于，所述成像装置显示包括用于所述一个或多个应用程序的应用程序名和应用程序概述的信息。

14、如权利要求 11 所述的方法，其特征在于，所述成像装置根据所述确定步骤的结果，显示不能安装应用程序的原因。

15、如权利要求 11 所述的方法，其特征在于，当在所述成像装置中已经安装了被选择用于安装的应用程序时，所述成像装置显示警告。

16、如权利要求 11 所述的方法，其特征在于，在所述显示面板上显示的所述应用程序信息或与所述应用程序信息有关的信息包括用于确定是否可安装所述应用程序所需的信息。

17、如权利要求 16 所述的方法，其特征在于，用于确定是否可安装所述应用程序所需的信息包括所述应用程序的版本信息以及所述成像装置的版本信息。

18、如权利要求 16 所述的方法，其特征在于，用于确定是否可安装所述应用程序所需的信息包括所述应用程序的所需资源信息以及所述成像装置的可用资源信息。

19、如权利要求 11 所述的方法，其特征在于，所述显示部件通过使用所述应用程序信息显示安装操作屏或卸载操作屏，

其中，根据所述安装操作屏或所述卸载操作屏上的操作，所述成像装置执行用于所述应用程序的安装进程或卸载进程。

20、如权利要求 11 所述的方法，其特征在于，所述成像装置使远程装置显示与所述操作面板上显示上的屏幕相同的屏幕。

成像装置及其方法

本申请是名称为“成象装置及用在成象装置中的方法”（申请号：03164873.8；申请日：2003年9月15日）的申请的分案申请。

技术领域

本发明涉及为用户提供有关成像进程诸如复制、打印、扫描、传真等等有关的服务的成像装置及其方法和程序。

背景技术

近年来，在机壳内包括打印机、复印机、传真机、扫描仪等等的功能的成像装置（以下称为复合机）是很公知的。复合机在机壳内包括显示部件、打印部件、图像拾取部件等等。在复合机中，提供分别对应于打印机、复印机和传真机的三种软件，以便通过切换软件，复合机分别充当打印机、复印机、扫描仪和传真机。

根据这种传统的复合机，运行对应于每种能力，诸如打印、复制和扫描的应用程序。至于现有的应用程序，由于预先了解由应用程序使用的资源量、以及应用程序和复合机系统间的版本一致性等等，应用程序能简单地安装到该复合机中并能运行。因此，例如，在安装应用程序之前或在运行应用程序之前，不必核对由应用程序使用的资源量。

由于传统的复合机具有单独地用于打印机、复印机、扫描仪和传真机的每个软件，所以需要大量时间来开发软件。因此，申请人开发了包括硬件资源、多个应用程序和包括在应用程序和硬件资源间提供的各种控制服务的平台的成像装置（复合机）。硬件资源包括显示部件、打印部件和图像拾取部件。应用程序执行用于打印机、复印机和传真机等等的用户服务的内部进程。平台包括当执行用户服务时，执行各种控制服务，该各种控制服务用于至少两个应用程序共同所需的硬件资源的管理、执行应用程序的执行控制以及执行成像进程。

根据复合机，由于与应用程序分开提供控制服务，所以能将由用户或第

三方卖主提供的新应用程序安装在复合机中。因此，存在下述问题。

至于现有的应用程序，诸如复制应用程序，由于预先知道版本和所需资源量，当安装和运行应用程序时，不可能存在由于版本不一致或缺少资源的故障。然而，关于由第三厂商或用户开发的新应用程序，当开发该新应用程序时，不考虑复合机的状态（版本和可用资源等等）。因此，存在当运行该新应用程序时，复合机的操作变得不稳定的可能性。更具体地说，例如，当由新应用程序所使用的所需资源量大于复合机中的可用资源量时，由于缺少资源，会停止复合机的操作。

另外，至于用于复制、打印、传真和扫描的现有应用程序，可将这些应用程序设计成即使在复合机中同时执行它们时，也能稳定操作。然而，至于由第三方开发的新应用程序，开发新应用程序不一定考虑到与其他应用程序的兼容性。因此，在新应用程序与另一应用程序不是相互兼容的情况下，如果与另一应用程序同时执行新应用程序，该新应用程序异常操作或不操作。为解决该问题，可考虑为应用程序分配优先级。然而，每次安装新应用程序时，都应当更新优先级。另外，对普遍用户来说，很难确定应用程序的优先级。

另外，在由第三厂商开发的新应用程序中，存在应当与另一应用程序一起执行的新应用程序。如果用户仅执行需要另一应用程序的新应用程序，该新应用程序不能提供其全部能力，或该应用程序会停止。

发明内容

本发明的第一目的是提供用于获得用于确定是否可安装应用程序的应用程序信息的成象装置和方法。本发明的第二目的是提供用于在操作面板上显示应用程序信息的成象装置和方法。本发明的第三目的是提供用于根据有关资源的信息，执行运行限制进程的成象装置和方法。本发明的第四目的是提供用于执行不兼容或不可缺少的应用程序的运行控制的成象装置和方法。

由成象装置实现上述目的，能在所述成象装置中安装多个应用程序，其中成象装置包括：应用程序信息获得部件，用于获得应用程序信息，所述应用程序信息用于确定是否能在所述成象装置中安装应用程序。

根据本发明，可确定是否可安装应用程序。

也可由包括操作面板的成象装置实现上述目的，操作面板用于显示或输

入操作信息，其中成象装置包括：

信息获得部件，用于获得有关将在所述成象装置中使用的应用程序的应用程序信息；以及

显示部件，用于在所述操作面板上显示所述应用程序信息或与所述应用程序信息有关的信息。

根据本发明，由于在操作面板上显示应用程序信息，用户能确定是否可安装应用程序。

上述目的也可由其中能安装多个应用程序的成象装置实现，其中成象装置包括：

信息获得部件，用于在所述成象装置中获得执行一个应用程序的所需资源信息和可用资源信息；以及

运行限制部件，用于将所需资源信息与所述可用资源信息进行比较，并根据所述所需资源信息与所述可用资源信息间的比较结果，在所述应用程序上执行运行限制进程。

根据本发明，成象装置能防止缺少资源的情况下错误运行应用程序。因此，能稳定地操作成象装置。

也可通过其中能安装多个应用程序的成象装置来实现上述目的，其中，成象装置包括：

用于引用数据库的部件，所述数据库包括用于每个应用程序的有关应用程序的特性组合的信息；以及

运行控制部件，用于根据有关特性的信息，在应用程序上执行运行控制进程。

根据本发明，能根据有关应用程序特性组合的信息运行适当的应用程序。

附图说明

从结合附图阅读下述的详细描述，本发明的其他目的、特点和优点将更显而易见，其中：

图 1 是根据本发明的第一实施例的复合机的框图；

图 2 是根据第一实施例的复合机的硬件结构的框图；

图 3 是表示根据第一实施例的 VAS 140 的结构，和在 VAS 140、每个应用程序、控制服务层 150 和通用 OS 121 间的关系的框图；

图 4 表示 VAS 140 与用于存储应用程序信息、应用程序信息表 106 和 NVRAM 104 的硬盘 200 间的关系;

图 5 表示用于在复合机 100 中执行安装进程或卸载进程的整个过程;

图 6 是表示在图 5 的步骤 S401 中的应用程序信息获得进程的子例程的流程图;

图 7 表示包括临时运行函数的一个应用程序过程描述(主函数)的例子;

图 8 表示 VAS 和应用程序间的顺序;

图 9 表示根据第二实施例的初始设置屏的验证屏;

图 10-16 表示用于根据第二实施例的文件复制操作屏;

图 17-28 表示用于根据第二实施例的安装操作屏;

图 29-36 表示用于根据第二实施例的卸载操作屏;

图 37 表示根据第二实施例的网络文件应用程序的结构;

图 38 是表示根据第三实施例的 VAS 140 的结构,以及在 VAS 140、应用程序、控制服务层 150 和通用 OS 121 间的关系的框图;

图 39 表示资源使用信息文件 1201 的例子;

图 40 表示由资源使用信息获得线程 1141 引用的进程结构 211 的例子;

图 41 表示在操作面板上显示的屏幕图形,其中通过使用直方图表示运行限制级;

图 42 表示运行设置屏,其中已经执行了用于“simple scanner”和“simple print”的运行设置;

图 43 表示运行设置屏,其中选择“简单打印”;

图 44 是用于建议选择另一应用程序的屏幕;

图 45 是即使在复合机中不包括足够的存储器的情况下,也能执行运行设置的屏幕的例子;

图 46 是当选择没有由任何键所指定的应用程序时显示的屏幕;

图 47 是表示运行设置完成的屏幕;

图 48 是表示在结束运行设置后显示的屏幕;

图 49-53 表示执行运行设置释放的屏幕;

图 54 表示根据第三实施例,当执行运行设置时,由 VAS 140 执行的进程;

图 55 是根据第三实施例,当运行应用程序时,执行运行限制进程的流程图;

图 56 表示用于当产生资源使用信息文件时，用于解释 VAS 140 的操作的流程图；

图 57 是第四实施例的复合机的框图；

图 58 是表示第四实施例的复合机 800 的 VAS841-848 的结构图；

图 59 表示第五实施例的 VAS 140 的结构，并表示在 VAS 140、每个应用程序、控制服务层 150 和通用 OS 121 间的关系；

图 60 表示应用程序兼容性信息文件 2201 的例子；

图 61 表示根据第五实施例的另一文件；

图 62 表示根据第五实施例，用于不能同时运行的应用程序的用户选择屏的例子；

图 63 表示根据第五实施例，由 VAS 的应用程序运行限制进程的过程的流程图；

图 64 表示根据第五实施例，步骤 S2505 的进程的过程的流程图；

图 65 表示根据第五实施例，运行限制进程的过程的流程图；

图 66 是表示第六实施例的复合机 100 的 VAS941-948 的结构，以及在 VAS941-948、每个应用程序、控制服务层 150 和通用 OS 121 间的关系的图；

图 67 表示第七实施例的 VAS 140 的结构，并表示在 VAS 140、每个应用程序、控制服务层 150 和通用 OS 121 间的关系；

图 68 是不可缺少的应用程序信息文件 2202 的例子；

图 69 是用于提示用户运行用于运行所要求的应用程序（浏览程序）的不可缺少的应用程序（插件程序）的屏幕的例子；

图 70 表示根据第七实施例，VAS 140 的应用程序运行控制进程的过程的流程图；

图 71 表示根据第七实施例，步骤 S2804 的进程的过程的流程图；

图 72 表示根据第七实施例，运行控制进程的过程的流程图；

图 73 是表示复合机 100 的 VAS941-948 的结构，以及在 VAS941-948、每个应用程序、控制服务层 150 和通用 OS 121 之间的关系的结构的图；

图 74 表示第八实施例的数据库；

图 75 表示根据第八实施例，复合机 100 的运行控制的过程；

图 76-81 表示当根据第八实施例执行运行设置时所显示的屏幕。

具体实施方式

在下文中，将参考图来描述成象装置及方法的实施例。

(第一实施例)

图 1 是根据本发明的第一实施例，成象装置（在下文中称为复合机）的框图。如图 1 所示，复合机 100 包括硬件资源和软件组 110。硬件资源包括黑白行式打印机（B&W LP）101、彩色激光打印机 102、和硬件资源 103，该硬件资源诸如扫描仪、传真机、硬盘、存储器（RAM、NV-RAM、ROM 等等）以及网络接口。软件组 110 包括平台 120、应用程序 130 和虚拟应用服务 140（以下称为 VAS）。

VAS 140 作为用于控制服务的客户机进程工作并且操作为用于应用程序的服务器进程工作。VAS 140 在执行安装进程前，获得应用程序信息，其中执行安装进程以便使应用程序在复合机中可运行。然后，VAS 140 在硬盘 200 中产生应用程序信息文件。例如，所获得的信息是产品信息。产品信息包括产品 ID、厂商名称、应用名、版本、联系地址（电话号码）、所需资源、有关安装的确切以及相应的应用程序信息。产品信息能包括在应用程序的执行文件中，或可通过经例如，网络读取文件获得。另外，当已经将产品信息存储在硬盘 200、NVRAM 104、应用程序信息表 106 等等中时，能通过读取它们中的一个来获得产品信息。

能通过临时运行应用程序，由 VAS 140 获得包括在应用程序执行文件中的应用程序信息以便启动 VAS 140 和应用程序间的进程间通信。

通过临时运行应用程序，利用 VAS 140 能通过使用 VAS 140 和应用程序间的进程间通信获得应用程序的执行文件中的应用程序信息。

例如，能将应用程序从 PC 卡、IC 卡、SD 卡等等复制到硬盘上。然后，对应用程序执行安装进程，其中，将应用程序设置成可运行的。在本实施例中，在执行安装进程前，VAS 获得有关应用程序的应用程序信息，以便 VAS 能确定是否能将应用程序安装在复合机中。是否能安装应用程序是根据由于缺少资源或版本不一致性，应用程序是否使复合机不稳定操作或停止其操作而定。在安装应用程序之前，用于核对应用程序的原因在于安装不能运行的应用程序是无意义的，以及如果错误地运行应用程序，该应用程序可不利地影响复合机的操作。

平台 120 包括用于解释来自应用程序的进程请求并发布用于硬件资源的

查询请求的控制服务，用于管理一个或多个硬件资源和判优来自控制服务的查询请求的系统资源管理器 (SRM) 123 和通用 OS 121。

控制服务包括多个服务模块，它们是系统控制服务 (SCS) 122、引擎控制服务 (ECS) 124、存储器控制服务 (MCS) 125、操作面板控制服务 (OCS) 126、传真控制服务 (FCS) 127、网络控制服务 (NCS) 128。另外，平台 120 具有能通过使用预定功能，从应用程序 130 接收进程请求的应用程序接口 (API)。

通用 OS 121 是通用操作系统，诸如 UNIX，并能将平台 120 和应用程序 130 的每个软件部分作为进程执行。

SRM 123 的进程是用于执行系统控制和用于执行具有 SCS 122 的资源管理。SRM 123 的进程对来自使用包括引擎，诸如扫描仪部件和打印机部件、存储器、HDD 文件、主 I/Os (Centronics I/F、网络 I/F、IEEE1394 I/F、RS232C I/F 等等) 的硬件资源的上层的请求，完成判优和执行控制。

更具体地说，SRM 123 确定所请求的硬件资源是否可用 (是否未由另一请求使用)，以及当所请求的硬件资源可用时，告知上层所请求的硬件资源可用。另外，SRM 123 对根据来自上层的请求所使用的硬件资源执行调度，并直接执行对应于请求的进程 (例如，打印机引擎的纸传送和成象、分配存储空间、文件生成等等)。

SCS 122 的进程执行应用程序管理、控制操作部分、显示系统屏、LED 显示器、资源管理和中断应用程序控制。

ECS 124 的进程控制硬件资源包括黑白激光打印机 (B&W LP) 101、彩色激光打印机 (Color LP) 102、扫描仪、传真机等等的引擎。MCS125 的进程获得和释放图象存储器的区域，使用硬盘装置 (HDD)，并压缩和扩展图象数据。

FCS 127 的进程提供 API，该 API 用于通过使用 PSTN/ISDN 网络，从每个应用程序发送和接收传真、由 BKM (备份 SRAM) 管理的各种类型的传真数据的注册/查阅、传真读取、传真接收和打印以及混合发送和接收。

NCS128 是用于提供通常用于需要网络 I/O 的应用程序的业务的进程。NCS128 将按协议从网络接收的数据分配给相应的应用程序，并且当将数据发送给网络时，充当应用程序和网络间的媒介。更具体地说，NCS128 的进程包括协议的服务器数据自适应鉴定监视器 (server daemon)，诸如 ftpd、httpd、

lpd、snmpd、telnetd、smtpd 和客户机函数。

OCS 126 进程控制用于在操作员（用户）和机器的控制部件间传送信息的装置的操作面板。在本发明的复合机 100 中，OCS 126 包括 OCS 进程部分和 OCS 函数库部分。OCS 进程部分获得密钥事件，其表示从操作面板按压密钥，以及将对应于密钥事件的密钥事件函数发送给 SCS 122。OCS 函数库注册制图函数和用于控制操作面板的其他函数，其中制图函数用于根据来自应用程序或来自控制服务的请求，将不同图象输出到操作面板上。OCS 函数库动态地链接到应用程序和控制服务的每个模块。能将所有 OCS 126 构造为进程，或能构造为 OCS 库。

应用程序 130 包括打印机应用程序 111（该打印机应用程序是用于打印机的应用程序，具有页面描述语言（PDL）和 PCL 以及附录（PS））、复制应用程序 112、传真应用程序 113、用于扫描仪的扫描仪应用程序 114、网络文件应用程序 115、进程检验应用程序 116 和新应用程序 117 和 118。这些应用程序是专用于复合机 110 的应用程序。当运行应用程序时，应用程序将具有进程 ID 的应用程序注册请求消息发送给 VAS 140。然后，VAS 140 执行用于该应用程序的注册进程。

在应用程序 130 的进程和控制服务的进程间执行进程间通信，其中调用函数，发送返回值，以及发送和接收消息。通过使用进程间通信，能实现用于成象进程，诸如复制、打印、扫描和发送传真的用户服务。

如上所述，第一实施例的复合机 100 包括多个应用程序 130 以及多个控制服务，以及这些中的每一个充当一个进程。在每个进程中，生成一个或多个线程以及并行执行线程。控制服务为应用程序 130 提供公用服务。提供有关图象形成诸如复制、打印、扫描和发送传真的用户服务同时并行执行这些进程，并且执行这些线程，以及执行进程间通信。第三方卖主能开发用于复合机 100 的应用程序，以及能执行复合机 100 中控制服务层上的应用层中的应用程序。如图 1 中所示的新应用程序 117 和 118 是例子。能逐个增加或删除新应用程序。

图 2 表示复合机 100 的硬件结构的例子。

复合机 100 包括控制器 160、操作面板 175、传真控制单元（FCU）176、以及专用于成象处理的诸如打印机的硬件资源的引擎部件 177。控制器 160 包括 CPU161、系统存储器 162、北桥（NB）163、南桥（SB）164、ASIC166、

局部存储器 167、HDD168、网络接口卡 (NIC) 169、SD 卡槽 170、USB 设备 171、IEEE1394 设备 172、以及 Centronics173。存储器 162、167 可包括例如 RAMs 和/或 ROMs。FCU176 以及引擎部件 177 经 PCI 总线 178 连接到控制器中的 ASIC166。CPU161 执行应用程序并通过从 RAM 读取数据，执行安装在复合机 100 中的控制服务等。

图 3 是表示 VAS 140 的结构，以及在 VAS 140、每个应用程序、控制服务层 150 和通用 OS 121 之间的关系框图。

VAS 140 的进程包括调度程序 145、控制线程 144 和应用程序信息获得线程 141 和屏幕显示线程 142 以及应用程序检验线程 143。屏幕显示线程 142 在操作面板上产生操作屏。应用程序核对线程 143 根据所获得的应用程序信息，核对是否能安装应用程序。

调度程序 145 监视来自应用程序 130 或来自控制服务的消息接收。调度程序 145 根据所接收的消息，将进程请求发送给控制线程 144、应用程序信息获得线程 141、屏幕显示线程 142 和应用程序核对线程 143。

控制线程 144 请求线程 141-143 以便执行应用程序信息获得进程、屏幕显示进程，以及应用程序核对进程。在应用程序核对进程中，VAS 核对如果运行该应用程序，该应用程序是否不利地影响复合机 100。例如，VAS 140 获得请求安装哪个应用程序的应用程序信息以及获得已经安装完的应用程序的应用程序信息。然后，VAS 能核对应用程序的资源或版本以便核对应用程序是否不利地影响复合机。那时，屏幕显示线程 142 通过使用存储在硬盘 200 中的应用程序信息文件中的应用程序信息，在操作面板 105 上显示应用程序的应用程序信息。因此，用户能了解将安装的应用程序的状态。因此，即使当将应用程序确定为不可安装时，用户也能理解确定的原因。

另外，控制线程 144 控制安装进程和卸载进程，其中，当确定该应用程序是可安装的时，执行安装进程，以及当该应用程序变为不必要时，执行卸载进程。

当应用程序信息获得线程 141 从控制线程 144 接收到进程请求时，应用程序信息获得线程 141 临时运行未安装的应用程序以便启动 VAS 和应用程序间的进程间通信，并从该应用程序获得应用程序信息。关于安装的应用程序，通过运行该应用程序，通过进程间通信获得应用程序信息。将所获得的应用程序信息存储在硬盘 200 中，作为应用程序信息文件 201。将应用程序信息

记录为用于每个应用程序的记录。

能将 VAS 140 的程序提供为 SDK（软件开发工具包）的一部分或全部。能将程序提供为记录介质，诸如 CD-ROM 或 FD 中的可执行文件或可安装文件。另外，能经网络提供 VAS 140 的程序文件。

图 4 表示 VAS 140 和存储应用程序信息的硬盘 200、应用程序表 106 和 NVRAM 104 间的关系。如图 4 所示，为每个应用程序管理存储在硬盘 200 中的产品信息。通过指定产品 ID，能指定对应于产品 ID 的应用程序信息。另外，硬盘 200 存储在复合机 100 中复制的应用程序 A 和 C 的数据。从 IC 卡安装应用程序 B。用虚线表示的“F”表示存在于补丁文件（patch）中的执行文件组。

NVRAM 104 是即使断电，也能保存存储器内容的非易失性存储器。NVRAM 104 为每个应用程序存储产品 ID、运行信息、另外的信息。当安装应用程序时，将有关应用程序的信息记录在 NVRAM 104 中。另外，运行信息表示当接通复合机的电源时，是否将运行相应的应用程序。

应用程序信息表将有关应用程序的信息存储为表数据。通过指定应用程序 A 的产品 ID，能从该表获得应用程序名、版本和安装属性，以及能在操作面板上显示信息。

图 5 表示用于在复合机 100 中执行安装进程或卸载进程的整个过程。图 6 是表示在图 5 的步骤 S411 中的应用程序信息获得进程的子例程的流程图。

在图 5 的步骤 S401 中，调度程序 145 请求应用程序信息获得线程 141 以便经控制线程 144 获得应用程序的应用程序信息。应用程序信息获得线程 141 通过执行如图 6 所示的应用程序信息获得进程生成应用程序信息文件 201。

然后，控制线程 144 从应用程序信息文件 201 读取应用程序信息，并在步骤 S402 产生操作屏以便在操作面板上显示该操作屏。显示屏显示用于每个应用程序的应用程序名、版本、资源信息等等。另外，显示屏表示是否能安装应用程序。通过显示该屏幕，用户能理解应用程序的状态，以及用户能选择屏幕中的项以便进入安装进程。

在用户选择一些项后，复合机 100 在步骤 S403 等等用户输入，并在步骤 S404 确定是否存在用户请求。用户请求是应用程序的安装请求或应用程序的卸载请求。如果请求安装应用程序并允许该请求，在步骤 S405，将信息注册到如图 4 所示的 NVRAM 104。如果该请求是用于卸载该应用程序，在步骤 S406，

从图 4 所示的 NVRAM 104 删除应用程序信息。

图 6 是表示用于获得应用程序信息的操作的流程图。由于如果未运行应用程序则不能获得应用程序信息，所以 VAS 140 在步骤 S501 中临时运行该应用程序。然后，VAS 140 能通过使用进程间通信，从应用程序请求所需信息获得应用程序信息。根据本实施例，VAS 140 在步骤 S502 获得产品 ID，并访问 NVRAM 104，以便 VAS 140 能在步骤 S503 确定是否已经注册该应用程序。

如果在 NVRAM 104 未注册应用程序，则 VAS 140 在步骤 S504 确定是否存在产品信息文件。如果不存在产品信息文件，VAS 140 在步骤 S505 通过使用进程间通信，从该应用程序获得应用程序信息。然后，在步骤 S506 生成产品信息文件，以及 VAS 140 在步骤 S507 确定是否能安装应用程序。在步骤 S508，VAS 140 将确定结果反映到产品信息上。例如，通过将从应用程序获得的、将由该应用程序使用的资源信息与复合机的系统的可用资源信息进行比较，执行确定。

在步骤 S509，根据比较结果，确定是否能安装应用程序。当在步骤 S504 存在产品信息文件时，在步骤 S510，从产品信息文件获得应用程序信息。在步骤 S511 获得用于确定是否能安装应用程序的信息后，在步骤 S509 确定是否能安装该应用程序。

如果在步骤 S509 能安装该应用程序，在步骤 S512 设置可安装标志，以及在步骤 S516，将该标志写入应用程序信息表中。该应用程序不可安装时，在步骤 S513 设置不可安装标志，以及在步骤 S516 将该标志写入应用程序信息表中。如果在 S503 中，产品信息已经注册在 NVRAM 104 中，在步骤 S514，从 NVRAM 104 获得应用程序信息，以及在步骤 S515 设置安装标志并在步骤 S516 中，将该标志写入应用程序信息表。

如果对每个应用程序来说，未完成上述进程，步骤返回步骤 S501，并重复上述进程。

根据该实施例的复合机，由于获得用于所有应用程序的应用程序信息，所以能了解应用程序的状态。

接着，将描述临时运行应用程序。临时运行不同于应用程序的正常运行，其中使用复合机的独立资源（产生由该应用程序所需的函数(function)的运行称为“正常运行”）。在临时运行中，应用程序不获得资源，诸如当正常操作应用程序时所需的存储区。在临时运行中，应用程序仅执行与 VAS 140 进

程序间的通信，其中应用程序向 VAS 提供有关该应用程序的信息。当应用程序与 VAS 间的通信终止时，终止临时运行的应用程序的进程。另外，应用程序的临时运行的函数与应用程序的正常函数无关，并且对本实施例的复合机的应用程序是公用的。因此，例如，通过将包括临时运行的函数的程序模板提供给卖主，卖主能通过使用该程序模板，轻易地开发可应用于复合机的应用程序。为将应用程序信息包括在该应用程序中，例如，卖主产生将应用程序信息包括在其中的包含文件，并且当编译该应用程序时，将该包含文件包括到该应用程序中。

通过为应用程序提供临时运行函数，应用程序能将应用程序信息提供给 VAS 140，而不用正常运行，以便 VAS 140 能核对该应用程序。

图 7 表示包括临时运行函数的应用程序的程序描述（主函数）的例子。该描述能提供为程序模板。如图 7 所示，由自变量（-v）确定临时运行还是正常运行该应用程序。因此，当 VAS 140 运行该应用程序时，能轻易地选择正常运行和临时运行中的一个。即，当通过使用自变量（-v）选择临时运行时，临时运行该应用程序以便告知该应用程序信息。如果未选择临时运行，正常运行该应用程序以便能执行该应用程序的内在操作。

VAS 140 从应用程序获得信息并确定在该实施例中是否能安装应用程序。代替该过程，能采用如图 8 所示的过程。在图 8 所示的过程中，在 VAS 140 确定可安装应用程序时，VAS 140 在步骤 S601 询问该应用程序是否能真正在安装该应用程序。如果在步骤 S602，应用程序告知 VAS 140 能安装该应用程序时，在步骤 S603，VAS 140 执行用于该应用程序的安装过程。

在步骤 S601 和步骤 S602 之间，应用程序能从 VAS 140 获得系统信息（资源数据或配置数据），以及应用程序能根据系统信息确定是否能安装应用程序本身。

在本实施例中，VAS 140 从应用程序的执行文件获得应用程序信息。代替该方法，例如，VAS 140 也可从包括该应用程序信息的文件获得应用程序信息。通过将该文件存储在服务器中，VAS 140 能经网络获得该应用程序信息。

另外，能分别通过除 VAS 140 外的程序执行在操作面板上显示屏幕和核对应用程序。

（第二实施例）

在该实施例中，能更详细地描述由屏幕显示线程 142 显示的屏幕的例子。

图 9-36 表示在操作面板 105 上显示的操作屏幕的状态转换。图 9 表示初始设置屏幕的验证屏。图 10-16 表示用于文件复制操作的屏幕。图 17-28 表示用于安装操作的屏幕。图 29-36 表示用于卸载操作的屏幕。

在图 9 所示的验证屏幕中，输入用户 ID 和密码。在输入它们并按压 OK 按钮 210 后，能选择所需模式。在该实施例中，在安装应用程序前，在该复合机 100 中复制应用程序。如图 10A 所示，显示文件复制屏，以及选择将复制的文件的存储位置。在该实施例中，能从存储卡 211、SD 卡 212、和经由网络 213 中选择目的地。如图 10B 所示，当选择经由网络 213 并按压 OK 按钮 216 时，显示如图 11A 所示的下载屏或如图 11B 所示的复制屏。下载屏用于通过使用 Web 下载应用程序。复制屏用于通过使用 LAN 网络复制应用程序。当在图 10B 中按压取消按钮 215 时，屏幕返回到图 10A。当按压 EXIT 按钮 214 时，屏幕返回到未示出验证屏的图 9。

图 12A 表示选择存储卡 211 的情形。图 12B 表示选择 SD 卡 212 的情形。当按压 OK 按钮时，屏幕改变到显示文件列表产生等待屏的图 13A。当按压 CANCEL 按钮 221 时，屏幕返回到图 10A。图 13B 表示文件列表 222。

图 14A 表示从文件列表选择 SimpleCopy225 的屏幕。图 14B 表示另外选择 SimplePrint223 的屏幕。当在图 14A 中按压 OK 按钮 216 时，屏幕改变到图 15A 所示的等待屏。如果有相同的文件，显示如图 15B 所示的警告屏，其中显示用于覆盖的确认。当文件复制结束时，显示如图 16A 所示的屏幕。此时，完成将应用程序复制到复合机中。

图 17 表示安装操作屏的例子。当选择安装标志时，获得上述应用程序信息。

图 17A 表示诸如“允许注册”、“不允许注册”和“注册”。“注册”表示通过执行安装过程，将应用程序信息记录在 NVRAM 中。当 VAS 140 访问 NVRAM 时，VAS 140 能确定是否已经注册该应用程序。

当 VAS 140 确定由于某些原因，不能将应用程序安装在复合机 100 中时，显示“不允许注册”。例如，当在复合机 100 上不提供由应用程序使用的资源时，确定不能安装该应用程序。

“未注册（允许注册）”表示尽管可安装该应用程序，但还没有执行安装过程。

如图 17B 所示, 当指定可安装程序 SimpleCopy235 时, 应用程序的字段闪烁。另一方面, 如图 19A 所示, 当指定不允许注册应用程序 234 时, 屏幕改变到图 19B, 以及表示应用程序的信息不能安装并且显示原因。当 VAS 140 获得应用程序信息时, 将用于显示该消息的信息存储在复合机中。

在图 17A 所示的屏幕中, 当选择用于安装的应用程序时, 可如图 18 所示显示来源的介质 (从其中复制该应用程序) 以及目的介质 (其中在安装应用程序之后该应用程序所存在的介质)。在这种情况下, 能选择目的地。例如, 如图 18 所示, 能将复制在 SD 卡中的应用程序安装在 SD 卡中。

当选择安装应用程序时, 屏幕从图 20A 改变到图 20B, 以及显示警告屏。当选择可安装应用程序时, 在显示如图 21A 所示的注册等待屏 240 后, 显示如图 21B 所示的注册结束屏。然后, 结束应用程序注册 (应用程序安装)。因此, 如图 22 所示, 增加所安装的应用程序的数量。

图 23A 表示示出所安装的应用程序和不可安装应用程序的另一例子。从图 23B 的情形来看, 当不选择任何一项时, 屏幕返回到图 23A。如图 24A 所示, 能将多个应用程序选择为将要安装的应用程序。如图 24B 所示, 即使选择应用程序, 也能取消选择。

如图 25A 所示, 当选择不可安装应用程序时, 如图 25B 所示, 显示警告屏。如图 26A 所示, 当选择已安装应用程序时, 显示如图 26B 所示的警告屏, 其中显示表示将初始化设置值的警告。如图 27A 所示, 当正在注册应用程序时, 屏幕改变为等待屏。然后, 在图 27B 中显示完成结果。然后, 如图 28 所示, 显示所安装的应用程序名。

关于卸载进程, 显示如图 29A 所示的屏幕。图 29B 表示选择在将卸载的应用程序的屏幕。当选择不允许卸载的应用程序时, 显示如图 30A-30B 所示的警告屏。当允许卸载时, 显示图 31A 和 31B, 以及如图 32 所示显示卸载结果。

图 33 和 34 是当已经为该应用程序分配密钥时的卸载屏的例子。如图 34A 所示, 当选择 SimpleScan 时, 显示如图 34B 所示的屏幕, 其中请求是否释放密钥指定和卸载应用程序。当按压 OK 时, 显示如图 35A 和 35B 所示的屏幕, 以及如图 36 所示显示卸载结果。

如所提到的, 根据该实施例的复合机 100, 通过使用如图 9-36 所示的操作屏, 能容易操作复制过程、安装过程和卸载过程。由于在操作面板上显示

各个应用程序信息，用户能根据所显示的应用程序信息，顺序地选择屏幕。因此，能实现容易操作。

在图 9-36 所示的例子中，复合机 100 确定是否可安装应用程序，以及在操作面板上显示确定结果。然而，复合机 100 也能构造成复合机 100 简单地显示用于确定是否可安装应用程序所需的信息以使用户执行确定。在这种情况下，例如，显示用在应用程序中的 APIs 的版本信息以及实际上用在复合机中的 APIs 的版本信息。另外，可显示将由该应用程序使用的资源量和实际上在复合机中使用的可用资源量以便能比较它们。

能在装置，诸如经网络连接到复合机的 PC 上显示上述一系列屏幕，以及能由该装置执行安装进程等等。

在这种情况下，如图 1 所示的网络文件应用程序 115 具有 Web 服务器能力。作为网络文件应用程序 115 的一部分的 Java 程序生成一连串的画面并通过使用所谓的“小服务程序”方法，执行用于复合机 100 的安装进程等等。PC 包括 web 浏览程序。

在图 37 中示出了网络文件应用程序 115 的结构。如图 37 所示，网络文件应用程序 115 包括 web 服务器 1151、JVM (Java 虚拟机) 1152 和 Java 程序 1153。网络文件应用程序 115 从 PC 接收请求。然后，网络文件应用程序 115 执行安装进程并生成 HTML 文件以及将该 HTML 返回给 PC。

在 PC 上显示画面的同时，在复合机的操作面板上显示相同的画面，或仅在 PC 上显示画面。当在 PC 上显示画面的同时，在操作面板上显示相同的画面时，例如，网络文件应用程序 115 从 OCS 获得制图信息，并将该制图信息转换成 HTML 并发送 HTML。

(第三实施例)

在下文中，将描述第三实施例。第三实施例的复合机 100 的功能结构与第一实施例相同。即，第三实施例的复合机 100 的功能结构如图 1 所示。

关于通用个人计算机，如果逐个运行多个应用程序，存在由于缺少资源，使 PC 挂起的可能性。另一方面，关于本发明的复合机 100，不能挂起预安装的应用程序，诸如复制应用程序。

然而，由于复合机 100 能安装由第三卖主或用户开发的新应用程序，存在当如果已经将多个应用程序设定为运行，执行用于新应用程序的运行设置时会出现资源缺少的情形。

当电复合机 100 供电时，执行将运行的已安装应用程序的“运行设置”。即，在该实施例的复合机 100 中，仅通过安装该应用程序，不运行应用程序。在应用程序上执行用于运行该应用程序的运行设置是必要的。通过执行运行设置，表示将运行应用程序的信息存储在存储器中。然后，当给复合机 100 供电时，参考该信息以便运行该应用程序。

VAS 140 从该应用程序获得有关将由该应用程序使用的资源的信息（也称为所需资源信息），并通过使用硬盘（HD 200）中的信息生成资源使用信息文件。文本存储区大小、堆积区大小和堆栈区大小是有关将使用的资源的信息的例子。文本存储区是加载每个应用程序的程序的存储区。堆积区是动态保存每个应用程序的存储区。堆栈区是保留用于存储用于由每个应用程序调用外部模块的自变量的区。

当开发应用程序时，将有关所需资源量的信息包括在应用程序的执行文件中。然后，VAS 140 临时运行该应用程序以便通过使用进程间通信获得信息。

另外，通过将有关由该应用程序使用的资源的信息存储在文件中，通过引用该文件，VAS 140 能获得所需资源信息。

另外，如果实际上已经在复合机 100 上执行过该应用程序，VAS 140 能通过获得该应用程序的资源使用记录来获得所需资源量。

VAS 140 还通过从复合机 100 的总的资源量减去已经由应用程序和系统使用的资源量，获得复合机 100 中的可用资源量。对资源量已经由应用程序和系统程序使用的资源量，可使用记录在复合机中的使用记录。能将已经由应用程序和系统程序使用的资源量记录在资源使用信息文件中，以及 VAS 140 能引用它们。

当执行用于应用程序的运行设置时，VAS 140 将用于该应用程序的所需资源量与可用资源量进行比较。当在复合机中缺少资源量时，限制运行，即，不执行用于该应用程序的运行设置，除此之外，如果应用程序具有有限的执行模式，VAS 140 可将该应用程序设置成按有限执行方式运行，其中有限执行模式是例如，当存储区不够时，通过使用硬盘区代替存储区运行该应用程序的模式。

不仅当执行运行设置时，而且当实际上正好运行该应用程序时，能执行资源核对和运行限制。

图 38 是表示该实施例的 VAS 140 的结构框图，以及 VAS 140、应用程序、控制服务层 150 和通用 OS 121 中的关系。

在 VAS 140 的进程中，调度程序 144、控制线程 143、资源使用信息获得线程 1141、以及应用程序运行限制线程 1142 操作。

调度程序 144 监视从应用程序或控制服务接收的消息，并根据该消息，发送进程请求。

在实际上运行应用程序的情况下，当调度程序 144 接收来自控制服务的应用程序运行请求消息时，调度程序 144 将所接收的应用程序运行请求消息发送给控制线程 143。控制线程 143 将进程请求发送给资源使用信息获得线程 1141 和应用程序运行限制线程 1142。

在执行运行设置的情况下，例如，调度程序 144 将进程请求消息发送给控制线程 143，以及控制线程 143 将进程请求发送给资源使用信息获得线程 1141 和应用程序运行限制线程 1142。

控制线程 143 能通过引用资源使用信息文件 1201，确定是否记录用于所安装的应用程序的资源使用信息。

然后，控制线程 143 从资源使用信息获得线程获得资源使用信息，并将该资源使用信息发送给应用程序运行限制线程 1142。应用程序运行限制线程 1142 可引用资源使用信息文件 1201。应用程序运行限制线程 1142 将可用资源量和所需资源量进行比较。如果所需资源量大于可用资源量，应用程序运行限制线程 1142 执行运行限制进程。

当资源使用信息获得线程 1141 从控制线程 143 接收进程请求时，资源使用信息获得线程 1141 引用通用 OS 121 的过程结构（或 u 区 212）来获得由应用程序使用的文本存储区大小、堆积区大小和堆栈区大小。然后，资源使用信息获得线程 1141 在硬盘（HD）200 中产生资源使用信息文件 1201。将对于每个应用程序的资源使用信息记录为一条记录。其代替者，如前所述，资源使用信息获得线程 1141 可从包含在应用程序的执行文件中的应用程序获得所需资源信息，并将资源信息记录在资源使用信息文件 1201 中。

另外，资源使用信息获得线程 1141 能从过程结构 211 获得由当前进程使用的资源量，其中当前使用的资源量用于计算可用资源量。

代替通过使用线程执行上述处理，可将 VAS 140 构造为一个进程。

图 39 表示存储在 HDD200 中的资源使用信息文件 1201 的例子。如图 39

所示，资源使用信息文件 1201 存储用于每个应用程序 ID（也称为产品 ID）的文本存储区大小、堆积区大小和堆栈区大小。

该文件可存储复合机 100 的结构数据，例如 ADF。通过这样做，根据该结构数据，能执行运行限制。例如，如果复合机 100 不具有 ADF，限制运行使用该 ADF 的应用程序。

图 40 表示由资源使用信息获得线程 1141 引用的过程结构 211 的例子。如图 40 所示，过程结构 211 包括用于每个进程的进程 ID (P_pid)、文本存储区大小、堆积区大小和堆栈区大小。当执行进程时，当进程结束时，以及当进程改变状态时，由通用 OS 更新过程结构。

另外，资源使用信息获得线程 1141 获得已经由过程结构 211 设置成将运行的应用程序和系统程序使用的资源量，并通过从复合机的总资源量减去当前使用的资源量，计算可用资源量。

当应用程序运行限制线程 1142 接收用于新应用程序的运行请求，或接收运行设置请求时，应用程序运行限制线程 1142 引用资源使用信息文件 1201，并获得有关由新应用程序使用的资源的信息。另外，应用程序运行限制线程 1142 从例如，资源使用信息获得线程 1141 获得可用资源量。然后，应用程序运行限制线程 1142 比较它们并确定是否需要限制运行该新应用程序。

在运行设置的情况下，当应用程序运行限制线程 1142 确定应用程序运行限制是必要的，应用程序运行限制线程 1142 显示下述警告屏。

在实际运行的情况下，当应用程序限制线程 1142 确定应用程序运行限制是必要的，应用程序运行限制线程 1142 将运行限制请求消息发送给发送应用程序运行请求的应用程序。当应用程序运行限制线程 1142 确定应用程序运行限制是不必要的，应用程序运行限制线程 1142 将运行请求消息发送给应用程序以便执行正常的运行进程。

图 41 表示在操作面板上显示的屏幕的图，其中通过使用直方图显示运行限制级。当执行运行设置时，显示该屏幕。如图 41 所示的直方图表示通过从总的资源量减去由复合机中预安装的应用程序和系统程序使用的资源而获得的可用资源量。即，该图真实地表示如何在新应用程序中划分剩余资源。通过使用该屏幕，用户可真实地确定如何在新应用程序中划分剩余的资源。该直方图表示将由首先设置成将运行的用户应用程序 221 使用的资源量，以及其次设置成将运行的用户应用程序 222 使用的资源量。至于还未执行运行设

置的用户应用程序 223，示出了预测的资源量。在图 41 的例子中，如果执行用于用户应用程序 223 的运行设置，资源量达到危险级。因此，不能对用户应用程序 223 执行运行设置。在该屏幕中，通过使用颜色，例如，安全（绿）、警告（黄）和危险（红）来表示运行限制级。另外，在危险级的位置，直方图闪烁，或可输出语音引导或警告声。

在下文中，将示出在操作面板上显示的屏幕的例子。例如，由 VAS 140 显示每个屏幕。

图 42 表示运行设置屏，其中已经执行用于“simple scanner”和“simple print”的运行设置。如该图所示，由“simple scanner”和“simple print”使用的存储量如直方图 251 所示。

接着，当选择“simple print”用于执行运行设置，如图 43 所示时，“简单打印”部件闪烁几秒。然后，直方图的值变为 100%且直方图闪烁。

在此之后，显示如图 44 和 45 所示的屏幕。图 44 是用于建议选择另一应用程序的屏幕。通过按压 OK 按钮，屏幕返回图 42。图 45 是即使在复合机中不包括足够的存储器时，也能执行运行设置的屏幕的例子。当选择未指定任何密钥的应用程序时，显示如图 46 所示的屏幕。

在图 42 的屏幕中，当选择“simple scanner”时以及当结束运行设置时，在图 47 的屏幕后，显示如图 48 所示的屏幕。如图 48 所示，使直方图的值增加由“simple scanner”所使用的存储量。

当释放用于应用程序的运行设置时，显示如图 49-53 所示的屏幕。在图 49 的情况中，当选择用于释放运行设置的“simple scanner”时，如图 50 所示，“simple scanner”闪烁。如果按压 OK，显示如图 51 所示的屏幕。然后，如果按压 OK 按钮，经图 52 显示如图 53 所示的屏幕。如图 53 所示，从直方图减少简单扫描仪的存储量。

将参考图 54 的流程图，描述由复合机 100 的 VAS 140 执行的应用程序运行限制进程。图 54 表示当执行运行设置时，由 VAS 140 执行的过程。

当 VAS 140 在步骤 S1601 接收用于应用程序的运行设置请求时，VAS 140 在步骤 S1602，通过参考 HD 200，核对是否有用于该应用程序的资源使用信息。

当存在该资源使用信息文件时，VAS 140 在步骤 S1603，从资源使用信息文件 1201 获得资源使用信息。如果在步骤 S1602 不存在资源使用信息文件，

VAS 140 在步骤 S1604 产生资源使用信息文件。例如，控制线程 143 请求资源使用信息获得线程 1141 产生资源使用信息文件，以便资源使用信息获得线程 1141 产生资源使用信息文件。

稍后，将参考图 56 来描述用于产生资源使用信息文件的过程。

接着，在步骤 S1605，VAS 140 获得可用资源量。为计算可用资源量，通过使用系统调用或服务函数调用，能从过程结构获得已经对其执行了运行设置的应用程序的资源量。由通用 OS 提供系统调用，以及由控制服务层 150 提供服务函数。另外，如果在资源使用信息文件 1201 中包括资源量，能从资源使用信息文件 1201 获得该资源量。

接着，VAS 140 在 S1606，将请求运行设置的应用程序所使用的所需资源量与可用资源量进行比较。应用程序运行限制线程 1142 在步骤 S1607，确定所需资源量是否大于可用资源量。如果所需资源量大于可用资源量，在步骤 S1608 显示如图 44 所示的屏幕。如果所需资源量不大于可用资源量，在步骤 S1609 正常执行运行设置。

如果应用程序具有通过使用硬盘区代替存储区运行应用程序的模式，能显示用于询问用户是否使用该模式的屏幕。当用户选择该模式时，通过使用该模式执行该应用程序。VAS 140 能通过参考该应用程序信息了解该应用程序具有该模式。

图 55 是当运行应用程序时，用于执行运行限制进程的流程图。

当调度程序 144 从控制服务层接收应用程序运行请求消息时，调度程序 144 将具有进程 ID 的消息发送给控制线程 143。控制线程 143 在步骤 S1701 确定请求运行的应用程序是否是除已经预安装的应用程序，诸如复制应用程序外的应用程序，即，请求运行的应用程序是否是新的应用程序。如果在步骤 S1702，应用程序是新的应用程序，控制线程 143 在步骤 S1703，通过参考 HD 200，核对是否有用于应用程序的资源使用信息。

当存在用于该应用程序的资源使用信息文件时，控制线程 143 从资源使用信息文件 1201 获得资源使用信息，并在步骤 S1704 将资源使用信息发送给应用程序运行限制线程 1142。如果在步骤 S1703 中，资源使用信息文件不存在，控制线程 143 请求资源使用信息获得线程 1141 产生资源使用信息文件，以便在步骤 S1705，资源使用信息获得线程 1141 产生用于该新应用程序的资源使用信息文件 1201。在产生该文件后，执行步骤 S1704。

接着，VAS 140 在步骤 S1706 获得可用资源量。

接着，VAS 140 在步骤 S1707，将用于请求运行的新应用程序使用的所需资源量与可用资源量进行比较。在步骤 S1708，应用程序运行限制线程 1142 确定所需资源量是否大于可用资源量。如果所需资源量大于可用资源量，在步骤 S1709，显示警告屏。如果该应用程序具有通过使用硬盘区代替存储区运行应用程序的模式，能显示用于询问用户是否使用该模式的屏幕。当用户选择该模式时，通过使用该模式执行应用程序。

如果所需资源量不大于可用资源量，在步骤 S1710 正常运行该新应用程序。

图 56 表示当产生资源使用信息文件时，用于解释 VAS 140 的操作的流程图。

首先，在步骤 S1801，VAS 140 临时运行 VAS 140 想通过其获得资源信息的应用程序。然后，在步骤 S1802，VAS 140 通过使用进程间通信，从应用程序获得资源信息。然后，在步骤 S1803，VAS 140 通过使用所获得的资源信息产生资源使用信息文件 1201。图 39 是其中记录用于每个应用程序 ID 的文本存储区大小、堆积区大小和堆栈区大小的资源使用信息文件的例子。

如果已经运行该应用程序且将应用程序的资源使用的实际记录存储在过程结构 211 中，VAS 140 能获得使用记录而不从该应用程序获得信息。通过使用实际记录更新资源使用信息文件，能总是获得所更新的资源信息，即，提供学习能力。如果当每次获得记录时改变实际使用记录，记录的平均值或最大值能用作记录在资源使用信息文件 1201 中的资源信息。

在通过使用 RAM210 中的过程结构产生资源使用信息文件 1201 的情况下，当资源使用信息获得线程 1141 从控制线程 143 接收进程 ID、产品 ID 和资源使用信息获得请求消息时，资源使用信息获得线程 1141 引用过程结构 211，并搜索应用程序 ID 的进程 ID 的块的位置。然后，资源使用信息获得线程 1141 从进程 ID 块获得资源使用信息，以及将所获得的信息记录在应用程序 ID 的资源使用信息文件 1201 中。

根据该实施例的复合机，能防止由于缺少资源而使复合机操作不稳定。另外，能安全地执行由用户或第三卖主开发的新应用程序。

在该实施例中，VAS 140 对所有应用程序 130 执行资源使用获得进程以及应用程序运行限制进程。然而，也能将 VAS 140 构造成覆盖部分应用程序。

例如, VAS 140 可仅对新应用程序 117 和 118 执行资源使用信息获得进程和应用程序运行限制进程。

另外, 考虑到改变新应用程序的资源使用量, 即使所需资源量在某种程度上小于可用资源量, VAS 140 也可在新应用程序上执行运行限制。

(第四实施例)

第三实施例的复合机 100 包括用于所有应用程序的一个 VAS。根据第四实施例的复合机, 对每个应用程序运行多个 VAS, 其中每个 VAS 对相应的应用程序执行资源使用信息获得进程和应用程序运行限制进程。

图 57 是第四实施例的复合机的框图。如图 57 所示, 复合机 800 不同于第三实施例的复合机 100 之处在于对每个应用程序操作多个虚拟应用服务 841-848。

VAS841-848 对用于打印机应用程序 111、复制应用程序 112、传真应用程序 113、扫描仪应用程序 114、网络文件应用程序 115 和进程核对应用程序 116 和新应用程序 117 和 118 的相应应用程序, 执行资源使用信息获得进程和应用程序运行限制进程。

图 58 是第四实施例的复合机 800 的 VAS841-848 的结构, 以及 VAS841-848、每个应用程序、控制服务层 150 和通用 OS 121 中的关系的图。

根据第四实施例的复合机 800, 不同于第三实施例的复合机 100, 如图 58 所示, 在 VAS 和每个应用程序间操作 VAS 控制进程 (数据自适应鉴定监视器) 801。VAS 控制进程 801 产生用于每个相应的应用程序的 VAS841-848。VAS 控制进程 801 的操作与控制线程 143 类似。

VAS841-848 的每个进程包括调度程序 145 和进程 1141 和 1142。复合机 800 的每个线程的功能与第三实施例的相应线程的功能相同。根据第四实施例, 并行执行用于每个应用程序的处理, 以便能有效地执行运行限制进程。

根据第三和第四实施例, 文本存储区、堆积存储区和堆栈存储区是实例, 以及其他资源能用于应用程序运行限制。

(第五实施例)

在下文中, 将描述本发明的第五实施例。第五实施例的复合机 100 的功能结构与第一实施例相同。即, 第三实施例的复合机 100 的功能结构如图 1 所示。

在该实施例中, VAS 140 获得应用程序兼容性信息并在硬盘(下述 HD 200)

中产生应用程序兼容性信息，其中应用程序兼容性信息表示对于每个应用程序来说不可兼容应用程序。关于相互不可兼容应用程序，需要互斥控制。

应用程序的下述实例对要求互斥控制。例子是一对相同类型的应用程序，诸如一对验证应用程序以及一对记帐应用程序。另外，一对打印机应用程序和一对显示应用程序的每一对需要互斥控制，因为每对应用程序会干扰相同的设备（打印机或显示部件）。另外，一对不同版本的相同应用程序需要互斥控制。

另外，对下述情形，需要互斥控制。

如第一种情形，对需要大存储区的应用程序需要互斥控制。对使用非常大的 CPU 功率的应用程序，也需要互斥控制。

另外，对生成不同格式的 RAW 文件的应用程序需要互斥控制以便防止由于格式不匹配所带来的问题。

另外，关于验证或记帐应用程序，存在适合于特定验证或记帐应用程序的应用程序。例如，在扫描仪应用程序 A 和复制应用程序 A 适合于验证应用程序 A，以及扫描仪应用程序 B 和复制应用程序 B 适合于验证应用程序 B 的情况下，如果验证应用程序 A 和 B 的验证方法不同，在验证应用程序 A 和扫描仪/复制应用程序 B 之间需要互斥控制。另外，在验证应用程序 B 和扫描仪/复制应用程序 A 之间需要互斥控制。

另外，在日志获得应用程序和复制应用程序间需要互斥控制，其中日志获得应用程序产生所复制的图象的缩略图，以及复制应用程序立即删除在硬盘中的图象数据。

另外，例如，如果存在不能在复合机中同时执行卖主 A 和卖主 B 的应用程序的配置，需要互斥控制。

当 VAS 140 获得应用程序兼容性信息时，VAS 140 还获得每个应用程序的属性信息。例如，为显示屏幕，使用属性信息。

另外，VAS 140 通过使用进程间通信，临时运行应用程序来获得应用程序兼容性信息。VAS 140 可从外部服务器获得应用程序兼容性信息。

当执行运行设置或运行应用程序时，VAS 140 根据应用程序兼容性信息检测不兼容应用程序以便执行运行控制。例如，如果检测到不兼容应用程序，运行一个应用程序以及不运行另一应用程序。

图 59 表示第五实施例的 VAS 140 的结构，并表示 VAS 140、每个应用程

序、控制服务层 150 和通用 OS 121 间的关系。

VAS 140 的进程包括控制线程 143、应用程序兼容性信息获得线程 2141 以及应用程序运行限制线程 2142。还能执行由 VAS 140 的处理而不使用这些线程。

调度程序 144 监视从应用程序或控制服务接收消息。调度程序 144 根据所接收的消息，将进程请求发送给控制线程 143、应用程序兼容性信息获得线程 2141 以及应用程序运行限制线程 2142。在该实施例中，当调度程序 144 接收用于应用程序的应用程序运行请求消息时，调度程序 144 将消息发送给控制线程 143。控制线程 143 将进程请求发送给线程 2141 和线程 2142。

该实施例的应用程序运行限制进程是执行控制进程，其中仅运行一对应用程序中的一个应用程序而不运行另一应用程序。

当应用程序兼容性信息文件 2201 不包括所有安装的应用程序的应用程序兼容性信息时，控制线程 143 能请求线程 2141 以获得所需应用程序兼容性信息。将所获得的应用程序兼容性信息存储在应用程序兼容性信息文件 2201 中。

当应用程序兼容性信息文件 2201 包括所需的应用程序兼容性信息，控制线程 143 根据应用程序兼容性信息，请求应用程序运行限制线程 2142 执行应用程序运行限制。

应用程序兼容性信息获得线程 2141 通过使用进程间通信获得应用程序兼容性信息，并将该信息存储在应用程序兼容性信息文件 2201 中。另外，如果能经网络获得应用程序兼容性信息，应用程序兼容性信息获得线程 2141 通过使用 NCS128，从网络接收所需信息。应用程序兼容性存储记录为用于每个应用程序的记录。

应用程序兼容性信息文件 2201 能由用户存储在复合机 100 中。另外，当从服务器下载应用程序兼容性信息文件 2201 时，VAS 140 能从服务器获得应用程序兼容性文件 2201。

尽管每个线程的上述操作是当运行应用程序时的操作，当执行运行设置时，能执行相似的操作。用于运行设置的应用程序运行限制进程是例如，显示互不兼容应用程序并仅对一个选定的应用程序执行运行设置。

图 60 表示存储在 HDD200 中的应用程序兼容性信息文件 2201 的例子。如图所示，文件包括用于每个应用程序 ID（能称为产品 ID）的与应用程序不兼

容的一个或多个应用程序 ID。另外，HD 200 存储如图 61 所示的文件。如图 61 所示的文件包括用于每个应用程序 ID 的应用程序属性、操作目标、应用程序名（文件名）、版本等等。能使用该信息，用于显示屏幕以及也能使用该信息，用于确定在不兼容的应用程序中应当运行哪个应用程序。

应用程序运行限制进程 2142 引用应用程序兼容性信息文件 2201，确定是否有不兼容的应用程序。如果有不兼容的应用程序，应用程序运行限制线程 2142 进行仅运行不兼容应用程序中的一个应用程序的执行控制。在该进程中，当进行执行控制时，确定将运行哪个应用程序是很必要的。例如，通过为具有用于具有相同名称的应用程序的较新版本的应用程序分配较高优先级，能通过使用如图 61 所示的信息，自动确定在不兼容应用程序中将运行的应用程序。另外，通过在操作面板上显示不兼容应用程序，用户能选择应用程序。

图 62 表示用于不能同时运行的应用程序的用户选择屏的例子。如图 62 所示，屏幕表示当用户选择应用程序时使用的运行请求的应用程序列表 2211、不兼容应用程序列表 2212 以及单选按钮 2212。用户能通过使用操作键选择应用程序。

当应用程序运行限制线程 2142 确定应用程序运行限制是必要的时，应用程序运行限制线程 2142 将运行限制请求消息发送给不能运行的应用程序。另一方面，当应用程序运行限制线程 2142 确定运行限制是不必要的时，线程 2141 将运行请求消息发送给每个应用程序，以便正常运行该应用程序。另外，当运行限制是必要的时，根据互斥控制，将运行请求消息发送给将运行的应用程序。

接着，将描述由 VAS 140 的应用程序运行限制进程。图 63 表示该过程的流程图。

在步骤 S2501，当调度程序 144 从控制服务器 150 接收应用程序运行请求消息时，调度程序 144 将消息发送给具有应用程序的进程 ID 的控制线程 143。在步骤 S2502，控制线程 143 确定是否运行请求运行多个应用程序。当请求运行多个应用程序时，在步骤 S2503，确定是否存在应用程序兼容性信息文件 2201。当存在该文件时，控制线程 143 在步骤 S2504 引用该文件。当不存在该文件时，在步骤 S2505，生成应用程序兼容性信息文件，以及引用所生成的文件。稍后将参考图 64 来描述步骤 S2505 的进程。

在步骤 S2506，应用程序运行限制线程 2142 确定在所请求运行的应用程序中是否存在一对不兼容的应用程序。如果存在这样的对，在步骤 S2507 执行运行限制进程。稍后将参考图 65 来描述运行限制进程。

如果不存在不兼容应用程序对，在步骤 S2508 执行正常运行进程。

图 64 是用于解释生成应用程序兼容性信息文件 2201 的进程的流程图。

首先，在步骤 S2601，控制线程 143 确定是否能经网络获得应用程序兼容性信息。当经该网络不能获得信息时，在步骤 S2602，VAS 140 临时运行请求运行的应用程序。然后，在步骤 S2603，VAS 140 从该应用程序获得应用程序兼容性信息。然后，在步骤 S2604，通过使用应用程序兼容性信息产生用于应用程序的文件 201。用于应用程序的应用程序兼容性信息对应于如图 60 所示的文件记录。

如前所述，代替产生应用程序兼容性信息文件 2201，用户能准备应用程序兼容性信息文件 2201 并将其存储在复合机 100 中。

图 65 是运行限制进程的流程图。该流程图对应于图 63 中的步骤 S2507。

当对一对不兼容应用程序来说，运行限制是必要的时，在步骤 S2701，确定是否运行该对应用程序的一个或对该对应用程序的两个来说均不允许运行。例如，根据初始化设置，能执行该确定。当运行一个应用程序（在步骤 S2701 中的 Yes）时，在步骤 S2702，VAS 140 确定是否有选择标准。该选择标准可是预先确定的优先级，以及可记录在应用程序兼容性文件中。另外，该版本可用作规则的例子，其中当该对应用程序的名称相同时，将选择较新版本的应用程序。

当没有选择标准时，在步骤 S2703，VAS 140 在显示面板上显示包括不兼容应用程序列表的选择屏。图 62 表示该屏的例子。所请求运行的程序列表 2211 表示具有该版本的三个应用程序名。不兼容应用程序列表 2212 表示具有其他信息（版本、属性、目标）的两个不兼容应用程序（验证 AP 和验证 AP），以及在左侧提供用于选择的单选按钮。在步骤 S2704，用户参考在操作面板上显示的信息以使用户能通过核对单选按钮，轻易选择将运行的应用程序。

如上所述，在步骤 S2704，用户选择应用程序，进行执行控制，其中在步骤 S2705，限制运行未选择的应用程序。

在步骤 S2702，如果有运行应用程序的选择标准，运行该规则选择的应

用程序，以及在步骤 S2706，限制运行其他应用程序。

如果在步骤 S2701 限制运行所有不兼容应用程序，或在步骤 S2704，用户未选择任何应用程序，在步骤 S2707 中不选择所有不兼容应用程序。

尽管上述例子涉及实际运行应用程序，当执行运行设置，也能执行互斥控制。在运行设置的情况下，执行运行设置代替实际运行应用程序。

如上所述，根据该实施例的复合机 100，由于执行互斥控制以防止同时运行不兼容应用程序，复合机 100 稳定地操作。

另外，根据该实施例的复合机，当经网络获得应用程序兼容性信息时，能容易更新应用程序兼容性信息文件以包括最新信息。因此，能支持新应用程序。

(第六实施例)

第五实施例的复合机 100 包括用于所有应用程序的一个 VAS。根据第六实施例的复合机，为每个应用程序运行多个 VAS941-948，其中每个 VAS 为响应应用程序执行应用程序兼容性信息获得进程以及应用程序运行限制进程。

图 66 是表示第六实施例的复合机 100 的 VAS941-948 的结构，以及 VAS941-948、每个应用程序、控制服务层 150 和通用 OS 121 的关系的图。

根据第六实施例的复合机 100，不同于第五实施例的复合机，如图 66 所示，VAS 控制进程（数据自适应鉴定监视器）901 在 VAS 和每个应用程序间操作。VAS 控制进程 901 操作用于每个相应的应用程序的 VAS941-948。VAS 控制进程 901 的操作控制线程 143 类似。

VAS941-948 的每个进程包括调度程序 145 和线程 2141 和 2142。该实施例的复合机 100 的每个线程的操作与第五实施例的相应的线程相同。根据第六实施例，能并行执行用于每个应用程序的处理，以便能有效地执行运行限制进程。

(第七实施例)

在该实施例中，复合机 100 能同时运行不可缺少关系或推荐关系的应用程序。不可缺少关系的应用程序表示对执行另一应用程序来说，执行一个应用程序是不可缺少的。推荐关系的应用程序表示为执行另一应用程序，推荐执行一个应用程序。

该实施例的复合机 100 的结构与第五实施例的复合机 100 类似。VAS 140 的功能在第五实施例和该实施例间是不同的。

第七实施例的 VAS 140 引用不可缺少应用程序信息文件 2202, 以及识别能单独执行的应用程序, 以及要求运行另一应用程序的应用程序。可由 VAS 140 产生不可缺少的应用程序信息文件 2202, 或可由用户将其存储在复合机 100 中。另外, 复合机 100 能经网络从服务器获得不可缺少应用程序信息文件 2202。

例如, 在当通过将另一应用程序(插件软件)增加到应用程序以提供新能力的情况下, 不可缺少应用程序信息文件 2202 中的不可缺少应用程序是有关插件软件的信息。然而, 不可缺少应用程序信息不限于此。不可缺少的应用程序信息可是有关应用程序对的任何信息, 该应用程序对能同时执行以提供能力。

VAS 140 能经网络获得不可缺少应用程序信息, 以及还能通过临时运行该应用程序, 从应用程序获得信息。

VAS 140 获得对应于请求运行的应用程序的不可缺少应用程序信息。如果有用于应用程序的不可缺少的应用程序, VAS 140 控制复合机 100 以便运行该应用程序和不可缺少的应用程序。例如, 如果未请求运行不可缺少的应用程序, 不执行对应于该不可缺少的应用程序的应用程序的运行。另外, 例如, 当将不可缺少的应用程序安装在复合机 100 中时, 自动与该应用程序一起运行该不可缺少应用程序。另外, 例如, 可输出催促用户运行该不可缺少应用程序的通知。例如, 通知是在操作面板上显示的信息、语音指南、警告声等等。

图 67 表示该实施例的 VAS 140 的结构, 并表示 VAS 140、每个应用程序、控制服务层 150 和通用 OS 121 间的关系。

VAS 140 的进程包括调度程序 144、控制线程 143、不可缺少的应用程序信息获得线程 2145、以及应用程序运行控制线程 2146。调度程序 144 的功能与第五实施例相同。

当不可缺少的应用程序信息文件 2202 不包括用于应用程序的不可缺少的应用程序信息时, 控制线程 143 能请求线程 2145 以便获得所需不可缺少的应用程序信息。将所获得的信息存储在不可缺少应用程序信息文件 2202 中。

当不可缺少应用程序文件 2202 包括所需不可缺少应用程序信息时, 根据从文件 2202 读取的不可缺少应用程序信息, 控制线程 143 请求应用程序运行控制线程 2146 执行应用程序运行限制。

象第五实施例，不可缺少应用程序信息获得线程 2145 通过临时运行应用程序或经网络获得不可缺少应用程序信息，并生成不可缺少应用程序信息文件 2202。另外，能由用户准备不可缺少应用程序信息文件 2202 并将其存储在复合机 100 中。另外，复合机 100 可从服务器获得不可缺少应用程序信息文件 2202。

图 68 是存储在 HD 200 中的不可缺少应用程序信息文件 2202 的例子。如图 68 所示，不可缺少应用程序信息文件 2202 记录用于每个应用程序 ID 的存在或不存在不可缺少应用程序、应用程序类型以及不可缺少应用程序的应用程序 ID。

当应用程序运行控制线程 2146 从控制服务层 150 接收应用程序运行请求时，应用程序运行控制线程 2146 引用不可缺少应用程序信息文件 2202，并确定有关请求运行的应用程序的信息是否存储在文件 2202 中。如果记录了该信息，确定是否有对应于该应用程序的不可缺少的应用程序。如果不可缺少的应用程序不存在，正常运行该应用程序。如果不可缺少的应用程序存在，VAS 140 执行用于该应用程序和该不可缺少的应用程序的运行控制。在图 68 中，浏览程序 (ID: 103) 以及插件程序 (ID: 105) 是相互不可缺少的。

当有关请求运行的应用程序的信息未记录在不可缺少应用程序信息文件 2202 中时，应用程序运行控制线程 142 请求不可缺少应用程序信息获得线程 2145 以便获得有关该应用程序的信息。

图 69 是用于提示用户运行用于所请求运行的应用程序 (浏览程序) 的不可缺少应用程序 (插件程序) 的屏幕的例子。

接着，将描述由 VAS 140 执行的应用程序运行控制进程。图 70 表示该过程的流程图。

当在步骤 S2801，调度程序 144 从控制接收层 150 接收应用程序运行请求消息时，调度程序 144 将消息发送给具有应用程序的进程 ID 的控制线程 143。在步骤 S2802，控制线程 143 核对有关请求运行的应用程序的信息是否记录在不可缺少的应用程序信息文件 2202 中。当记录该信息时，在步骤 S2803 引用该文件 2202。

如果在步骤 S2802，信息未记录在文件 2202 中，在步骤 S2804，准备文件 2202，以及在步骤 S2803，引用该文件 2202。稍后将参考图 71 描述步骤 S2804 的进程。

在步骤 S2805, 应用程序运行控制线程 2146 通过引用不可缺少应用程序消息文件 2202, 确定是否有对应于请求运行的应用程序的不可缺少的应用程序。如果存在不可缺少的应用程序, 在步骤 S2806, 执行运行控制进程。如果不存在不可缺少的应用程序, 在步骤 S2807 中, 正常运行请求运行的应用程序。稍后, 将参考图 72, 详细地描述用于该应用程序和不可缺少的应用程序的运行控制进程。

图 71 是用于解释生成不可缺少的应用程序信息文件 2202 的进程的流程图。

首先, 在步骤 S2901, 控制线程 143 确定是否能通过网络获得不可缺少应用程序信息。当不能经网络获得该信息时, 在步骤 S2902, VAS 140 临时运行请求运行的应用程序。然后, 在步骤 S2903, VAS 140 获得所需信息。然后, 在步骤 S2904, 通过使用所获得的信息, 产生用于该应用程序的文件 2202。

图 72 是表示用于请求运行的应用程序和相应的不可缺少应用程序对的运行控制进程的流程图。

如图 72 所示, 首先, 在步骤 S3001, 应用程序运行控制线程 2146 确定是否请求运行不可缺少应用程序。如果是, 在步骤 S3008, 正常运行应用程序和不可缺少应用程序。

在步骤 S3002, 确定是否应当限制运行请求运行的应用程序直到请求运行不可缺少的应用程序为止。通过将表示应当运行该应用程序还是直到运行该不可缺少的应用程序才运行该应用程序的信息存储在文件 2202 中, 并通过引用该文件 2202, VAS 140 能执行步骤 S3002 的确定。

如果应当限制运行请求运行的应用程序, 在步骤 S3003, 输出用于提示运行不可缺少的应用程序的通知。作为通知的例子, 在操作面板上显示如图 69 所示的错误消息。在输出通知后, 进程返回到步骤 S3001, 以及 VAS 140 确定是否请求运行该应用程序和该不可缺少的应用程序。在步骤 S3003 后, 如果用户运行该应用程序和不可缺少的应用程序, 在步骤 S3008 中正常运行该应用程序 (在步骤 S3001 中为是)。

在步骤 S3002, 如果确定不限制运行请求运行的应用程序, 在步骤 S3004, 确定是否已经安装相应的不可缺少的应用程序。如果已经安装不可缺少的应用程序, 在步骤 S3007 和 S3008 自动正常地运行不可缺少的应用程序。

如果在步骤 S3004, 未安装不可缺少的应用程序, 在步骤 S3005, 确定是

否可安装不可缺少的应用程序。如果可安装，安装和运行不可缺少的应用程序（步骤 S3006 至 S3008）。

如果在步骤 S3005，不可缺少的应用程序是不可安装的，由于不能运行这两个应用程序，因此，进程结束。此时，VAS 140 能显示表示不可安装不可缺少的应用程序的屏幕。

尽管当实际运行该应用程序时，执行上述进程，当执行用于该应用程序的运行设置时，能执行类似的进程。在这种情况下，代替实际运行该应用程序，执行用于该应用程序的运行设置。

如上所述，根据第三实施例的复合机 100，由于能同时执行相互不可缺少的应用程序，复合机 100 能稳定地操作。

用与第六实施例相同的方式，能运行用于每个应用程序的多个 VAS941-948，其中每个 VAS 执行不可缺少的应用程序信息获得进程以及用于相应的应用程序的应用程序运行控制进程。

图 73 是表示复合机 100 的 VAS941-948 的结构，VAS941-948、每个应用程序、控制服务层 150 和通用 OS 121 中的关系的图。

（第八实施例）

在下文中，将描述第八实施例。第八实施例的复合机 100 的整个结构与第一实施例相同。

在第八实施例中，复合机 100 存储能表示用于应用程序的不兼容或非推荐关系和不可缺少或推荐关系的数据库。VAS 140 通过引用该数据库执行运行控制。在本说明中，不兼容或非推荐关系可简单地称为“不兼容关系”，以及不可缺少或推荐关系可简单地称为“不可缺少关系”。

在图 74 中示出了数据库文件的例子。如图 74 所示，文件表示用于每个应用程序的不可缺少或不兼容应用程序。在“不兼容或不可缺少应用程序”字段中，括号内的数字表示不兼容关系或不可缺少关系级。在正侧中的绝对值越大，不可缺少关系的等级越高。例如，“-1”表示即使可同时执行两个应用程序，由于存在会降低系统速度的可能性，因此不推荐执行这两个应用程序。另外，例如“-3”表示不能同时执行这些应用程序。另外，例如，“+1”表示尽管可单独执行每个应用程序，如果同时执行两个应用程序的话，能提供更多的功能。另外，例如，“+3”表示应当同时执行这些应用程序。关于不兼容关系，除上述等级外，能包括运行的优先级。VAS 140 通过引用如图 74

所示的文件执行运行控制。

除图 74 所示的信息外，文件可包括用于每对应用程序的不可缺少或不兼容关系的原因，以及其他信息。通过包括这些信息，能将这些原因显示在操作面板上。

能由用户将上述文件存储在复合机 100 中。另外，复合机 100 可从服务器获得文件。在当将文件存储在服务器中的情况下，如果找出有关应用程序不可缺少或不兼容的问题，通过使用正确的格式，能将问题信息发送给服务器，以便能自动更新文件。另外，当开发应用程序以及该应用程序对另一应用程序不可缺少或与另一应用程序不兼容时，能将这些信息发送给服务器，以便能自动更新该文件。

文件不必存储在复合机 100 中，以及 VAS 140 能引用存在于复合机 100 外的文件。

接着，将参考图 75 所示的流程图，描述通过使用图 74 中所示的文件的复合机 100 的运行控制的过程。下述过程用于当执行运行设置时的情形。

首先，在步骤 S3101 中，VAS 140 在操作面板上显示用于运行设置的屏幕。用于运行设置屏幕的例子如图 76 所示。如图 76 所示的屏幕表示已经执行用于“simply copy”的运行设置。

接着，用户选择用户想在其上执行运行设置的应用程序（在下文中称为“选定应用程序”）。VAS 140 获得选定应用程序的产品 ID，以及在步骤 S3102，根据产品 ID，搜索如图 74 所示的文件。

然后，在步骤 S3103，VAS 140 获得产品 IDs 以及对应于关键产品 ID 的相关信息（诸如，不兼容或不可缺少关系的原因）。例如，在图 74 的例子中，如果选定应用程序的产品 ID 为 101，VAS 140 获得 102 (-3) 和 103 (+1)。

然后，VAS 140 根据所获得的信息，执行运行控制。

如果对应于从文件获得的产品 ID 的应用程序相对于选定应用程序是不可兼容的应用程序，即，如果等级为负（在步骤 S3104 为是），在步骤 S3105，VAS 140 核对是否已经执行用于不兼容应用程序的运行设置。如果还未执行用于该不兼容应用程序的运行设置，在步骤 S3106，执行用于选定应用程序的运行设置。

如果已经执行用于不兼容应用程序的运行设置（在步骤 S3105 为是）。在步骤 S3107 核对等级。然后，VAS 140 显示警告屏（S3108，如果等级为低），

或显示选择在不兼容应用程序和选定应用程序间执行运行设置的应用程序的选择屏 (S3109)。警告屏表示询问用户是否能允许用于两个应用程序的运行设置的信息。当显示警告屏时, 在步骤 S3110, 停止选定的应用程序和不兼容应用程序的一个的运行设置或执行用于两个应用程序的运行设置。当显示选择屏时, 在步骤 S3111, 在选择屏中选定的应用程序上执行运行设置。在这种情况下, 如果选择所选定的应用程序, 释放用于不兼容应用程序的运行设置。

当对选定的应用程序来说, 与所获得的产品 ID 对应的应用程序是不可缺少的时 (在步骤 S3104 中为 N), 在步骤 S3112, VAS 140 核对是否已经在不可缺少的应用程序上执行运行设置。如果已经在不可缺少的应用程序上执行了运行设置, 在步骤 S3113 中执行所选定的应用程序的运行设置。如果还未在不可缺少的应用程序上执行运行设置, 执行下一进程 (在步骤 S3112 中为 N)。

在步骤 S3114 中核对不可缺少关系的等级。如果等级为高, 以便应当同时执行两个应用程序时, 在步骤 S3115, VAS 显示提示用户在不可缺少的应用程序上执行运行设置的屏幕。然后, 在步骤 S3116, 根据用户确定, 在不可缺少应用程序上执行运行设置。在这种情况下, 执行用于选定应用程序和不可缺少应用程序的运行设置。

例如, 在图 76 所示的屏幕中, 假定 “simple scanner” 是用于 “edit print tool” 的不可缺少应用程序, 当选择用于执行运行设置的 “edit print tool” 时, 如图 77 所示, 显示如图 78 所示的屏幕, 因为还未在 “simple scanner” 上执行运行设置。当用户按压 OK 按钮时, 在 “edit print tool” 和 “simple scanner” 上执行运行设置。

如果在步骤 S3114 中, 不可缺少关系的等级为低, 即, 如果该等级表示推荐关系 (例如, 能由另一应用程序提供扩展), 在步骤 S3117, VAS 140 显示用于通知用户所推荐的应用程序能将扩展增加到所选定的应用程序的屏幕。然后, 在步骤 S3118, VAS 140 在选定的应用程序和相应的推荐应用程序上执行运行设置。

例如, 在图 77 所示的例子中, 如果 “user stamp add module” 和 “user comment add module” 能将扩展增加到 “edit print tool”, 显示如图 79 所示的屏幕。如果按压 OK 按钮, 为这三个应用程序执行运行设置。

为执行用于不可缺少关系的应用程序的运行设置，能执行用于该应用程序的运行设置，而不询问用户。

当未将应当与选定的应用程序一起运行的不可缺少的应用程序安装在复合机 100 时，在显示图 78 的屏幕前显示图 80 的屏幕，以便能下载不可缺少的应用程序。如图 81 所示，可显示表示应用程序的引用的屏幕，而不显示用于下载的屏幕。

在上述例子中，在用于执行运行设置所选择的应用程序与另一应用程序不兼容的情况下，已经在该另一应用程序上执行过运行设置，VAS 140 能引用在所选定应用程序和另一应用程序间的优先级，并自动执行用于具有较高优先级的应用程序的运行设置。例如，如果所选定应用程序的优先级高于另一应用程序，VAS 140 能释放用于另一应用程序的运行设置并执行用于所选定应用程序的运行设置。

尽管上述例子是用于当执行运行设置时的情况，可将上述方法应用于当真正运行应用程序时的情形。

如上所述，根据本发明，提供能安装应用程序的成象装置，以及该成象装置包括：应用程序信息获得部件，用于获得用于确定是否可在成象装置中安装应用程序的应用程序信息。

根据本发明，能确定是否可安装应用程序。

在成象装置中，应用程序信息获得部件通过使用应用程序信息获得部件和应用程序间的进程间通信获得包括在应用程序的执行文件中的应用程序信息。

应用程序信息获得部件可从与该应用程序分开的文件获得应用程序信息。例如，可经网络获得应用程序信息。

在成象装置中，应用程序使成象装置执行：根据来自应用程序信息获得部件的请求，确定是否临时运行该应用程序或正常运行该应用程序的步骤；如果临时运行该应用程序，将应用程序信息提供给应用程序信息获得部件的步骤。因此，能选择正常运行或临时运行。

成象装置可进一步包括用于存储由应用程序信息获得部件获得的应用程序信息的存储器。存储器至少是硬盘、非易失存储器和存储卡的一种，以及应用程序信息获得部件通过读取存储在该存储器中的信息来获得应用程序信息。

另外，应用程序信息包括专用于该应用程序的产品 ID。该应用程序可进一步包括卖主名、应用程序名、版本信息以及资源信息中的至少一个。

在成象装置中，存储器存储可安装应用程序的产品 IDs，以及成象装置通过核对是否存储目标应用程序的产品 ID 来确定是否可安装目标应用程序。因此，很容易确定是否可安装应用程序。

另外，根据本发明，提供包括用于显示或输入操作信息的操作面板的成象装置，以及成象装置包括：

信息获得部件，用于获得有关在成象装置中使用的应用程序的应用程序信息；以及

显示部件，用于在操作面板上显示应用程序信息以及有关应用程序信息的信息。

根据本发明，由于在操作面板上显示应用程序信息，用户能确定是否可安装应用程序。

成象装置可进一步包括确定部件，用于根据由成象部件获得的应用程序信息，确定在成象装置中是否可安装应用程序。因此，成象装置可自动确定是否可安装应用程序。显示部件根据确定部件的确定结果，可在操作面板上显示一个或多个可安装应用程序，其中，可从一个或多个可安装应用程序选择可安装应用程序。

另外，显示部件根据确定部件的确定结果，可显示不能安装应用程序的原因。根据本发明，用户能了解不可安装应用程序的原因。例如，如果由于缺少资源资源不能安装应用程序，用户能知道该原因。

另外，当所选定的用于安装的应用程序已经安装在成象装置中时，显示部件能显示警告。

在成象装置中，在操作面板上显示的应用程序信息或与应用程序信息有关的信息可以是用于确定是否可安装应用程序所需的信息。根据该结构，用户能确定是否可安装应用程序。

成象装置可进一步包括用于使远程装置显示如在操作面板上显示的屏幕相同的屏幕的部件。因此，用户能从远程装置操作成象装置。

另外，根据本发明的，提供能安装多个应用程序的成象装置，其中成象装置包括：

信息获得部件，用于获得在成象装置中执行应用程序的所需资源信息和

可用资源信息;

运行限制部件, 用于将所需资源信息与可用资源信息进行比较, 并根据所需资源信息和可用资源信息间的比较结果, 在应用程序上执行运行限制进程。

根据本发明, 成象装置能防止在资源缺乏的情况下, 错误运行应用程序。因此, 能稳定地操作成象装置。

在成象装置中, 图象获得部件能获得由应用程序使用的资源的使用记录, 作为所需资源信息。因此, 能获得准确的信息。信息获得部件能从保存由成象装置的进程所使用的使用记录的系统信息中获得使用记录。

在成象装置中, 所需资源信息表示所需资源量, 以及可用资源信息表示可用资源量, 以及其中运行限制部件能在操作面板上显示所需资源量与可用资源量之比。根据该结构, 用户能获得所需资源量与可用资源量之比。

另外, 当所需资源量大于可用资源量时, 运行限制部件能在显示面板上显示警告。因此, 用户能知道缺少用于应用程序的资源。

在成象装置中, 如果应用程序能通过使用不同于对应于所需资源量的第一资源的第二资源来操作, 运行限制部件使应用程序使用第二资源代替第一资源。因此, 即使当缺乏资源时, 能通过使用另一资源执行该应用程序。

另外, 根据本发明, 提供其中安装多个应用程序的成象装置, 其中成象装置包括:

用于引用数据库的部件, 数据库包括用于每个应用程序的有关应用程序的组的特性的信息; 以及

运行控制部件, 用于根据有关特性的信息, 在应用程序上执行运行控制进程。

根据本发明, 根据有关应用程序的组的特性的信息, 能运行适当的应用程序。

在成象装置中, 有关特性的信息表示多个应用程序中的不兼容关系或不推荐的关系, 以及运行控制部件能在操作面板上显示用于从多个应用程序中选择至少一个应用程序的屏幕。因此, 用户能在互不兼容的应用程序中选择所需应用程序。

运行控制部件能运行多个应用程序中的至少一个应用程序, 或在多个应用程序的至少一个应用程序上执行运行设置。根据该结构, 成象装置能防止

同时运行互不兼容的应用程序。

另外，当不推荐应用程序的组合时，运行控制部件能在操作面板上显示警告屏。由于显示警告屏，用户能确定运行一个应用程序还是两个应用程序。

在成象装置中，有关特性的信息能表示不兼容关系或不推荐关系的等级，其中运行控制部件能根据该等级执行运行控制进程。

在成象装置中，有关特性的信息可表示多个应用程序中的不可缺少关系或推荐关系，其中运行控制部件能在操作面板上显示表示用于另一应用程序的不可缺少或推荐应用程序的屏幕。根据该结构，用户能选择不可缺少或推荐应用程序。

另外，运行控制部件能运行用于另一应用程序的不可缺少或推荐应用程序，或在用于另一应用程序的不可缺少或推荐应用程序上执行运行设置。当未在成象装置中安装不可缺少或推荐应用程序时，运行控制部件能经网络，从服务器下载不可缺少或推荐应用程序。同时，运行控制部件能显示用于获得不可缺少或推荐应用程序的信息。

在成象装置中，有关特性的信息能进一步表示不可缺少关系或推荐关系的等级，其中运行控制部件能根据等级执行运行控制过程。

本发明不限于特别公开的实施例，以及在不脱离本发明的范围的情况下，能做出各种改变和改进。

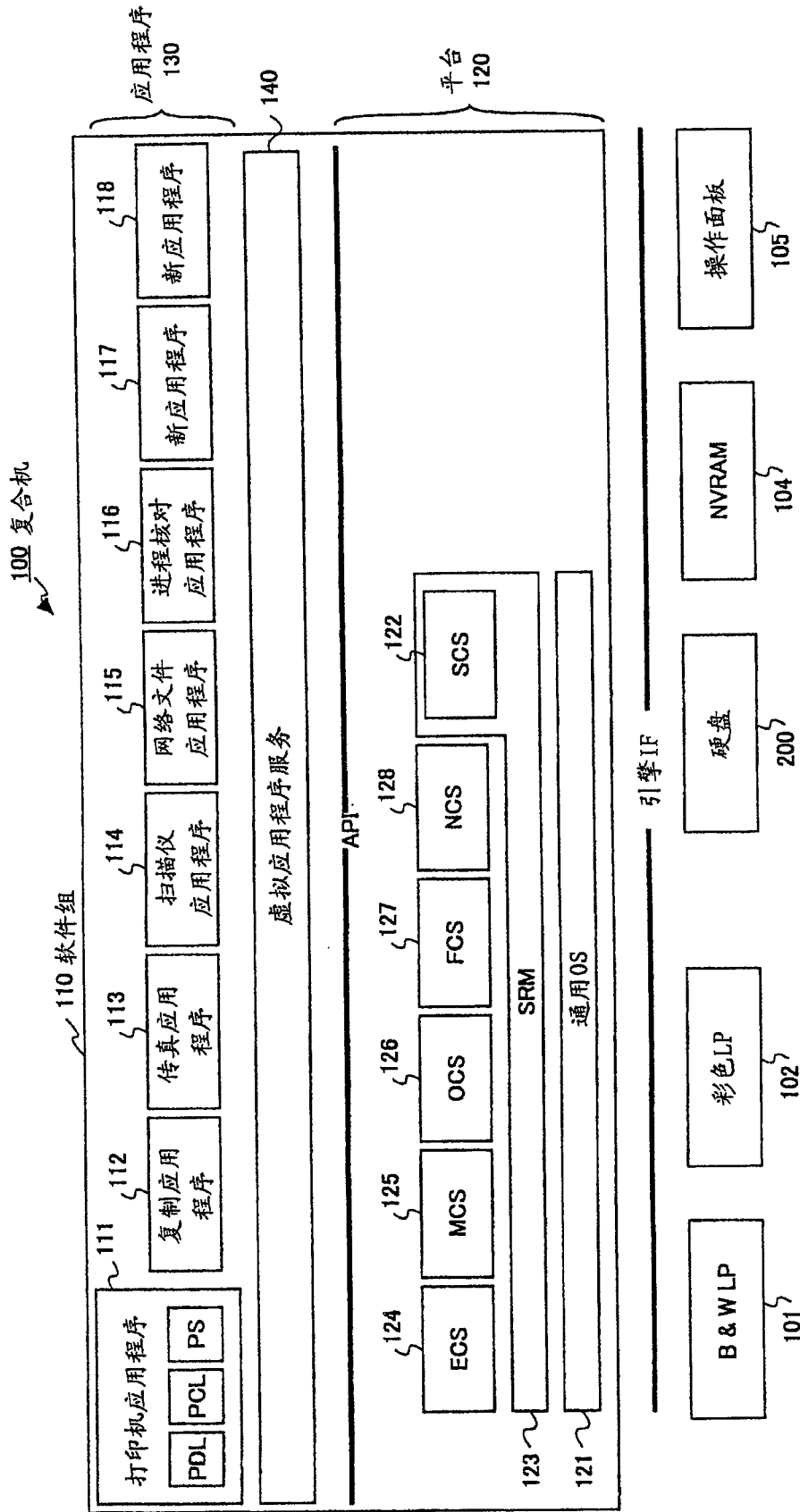


图 1

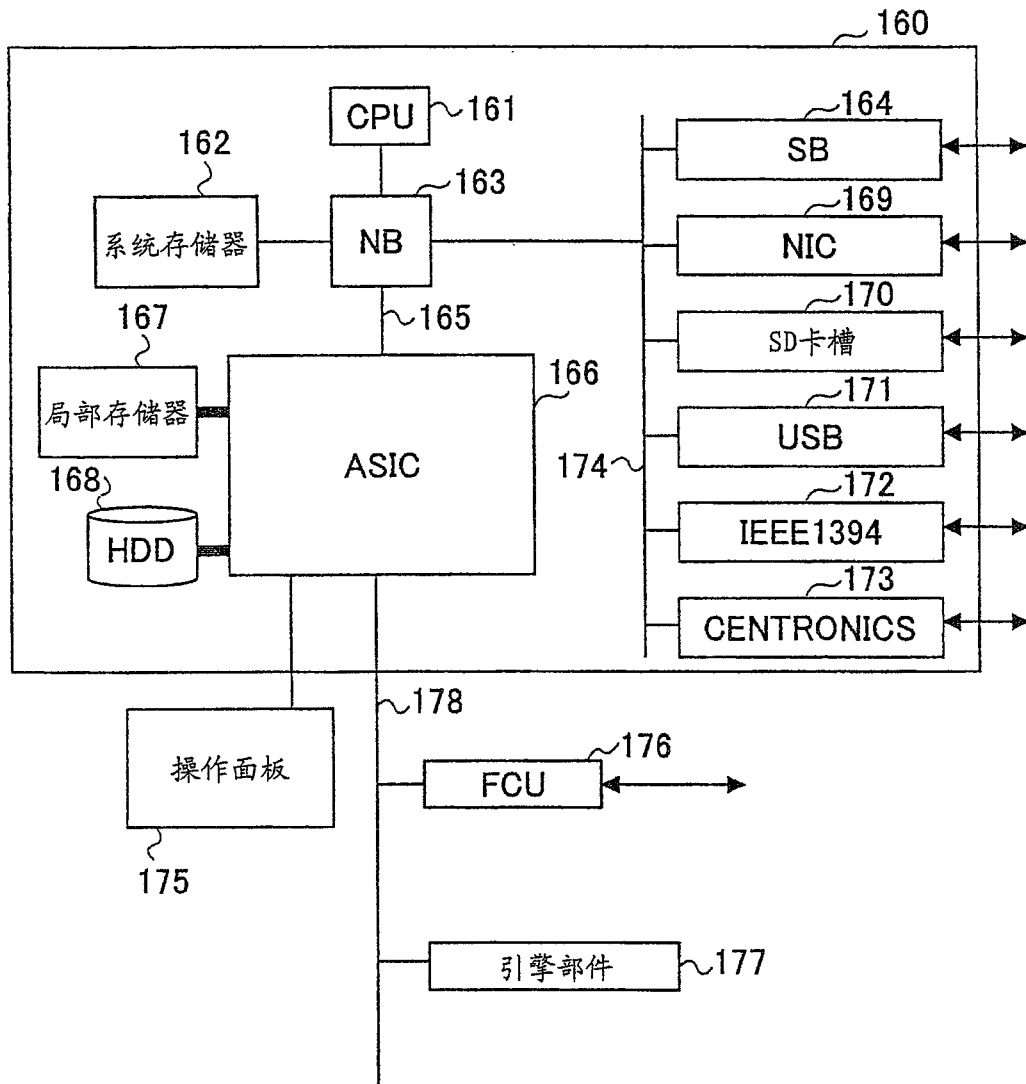


图 2

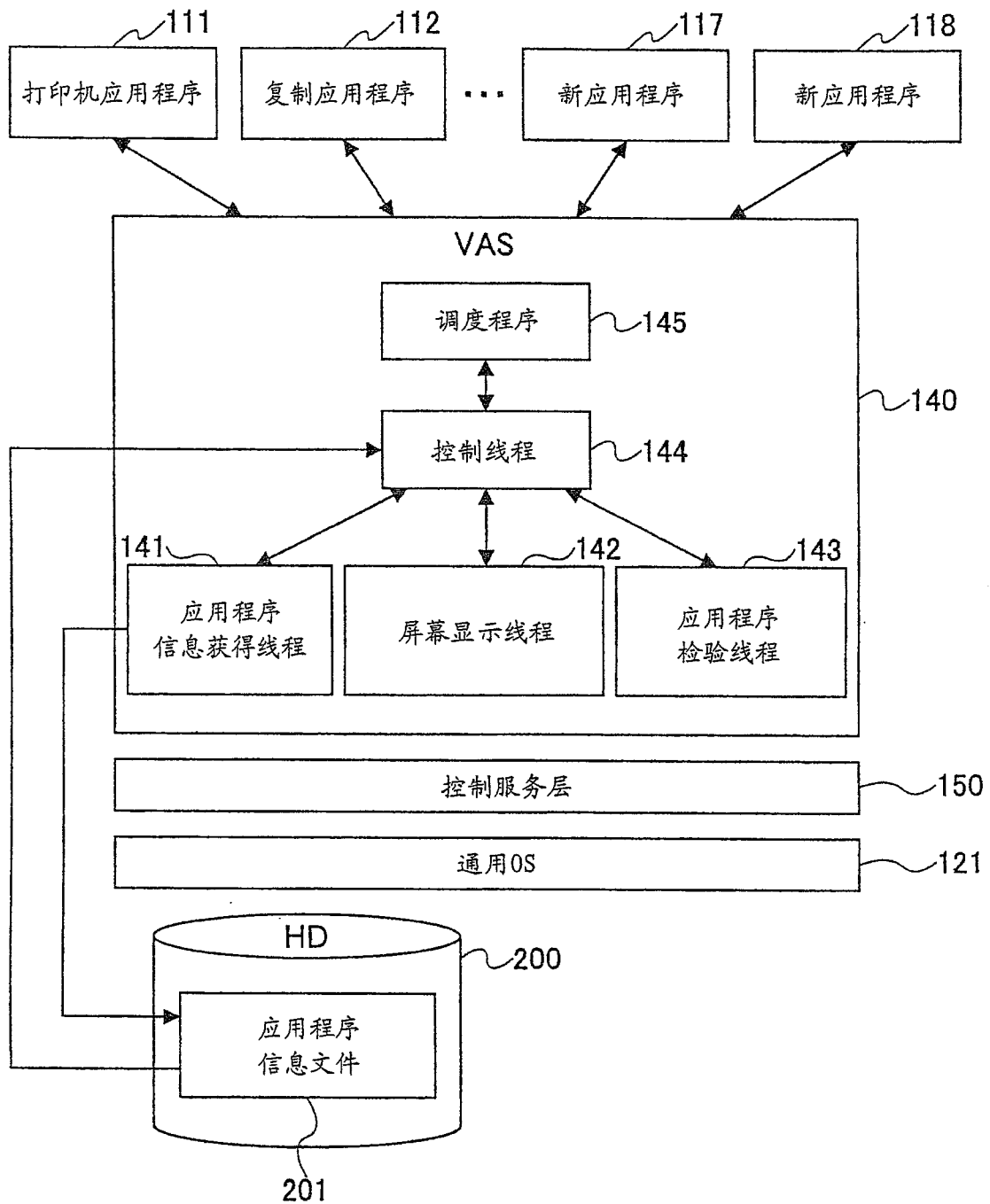


图 3

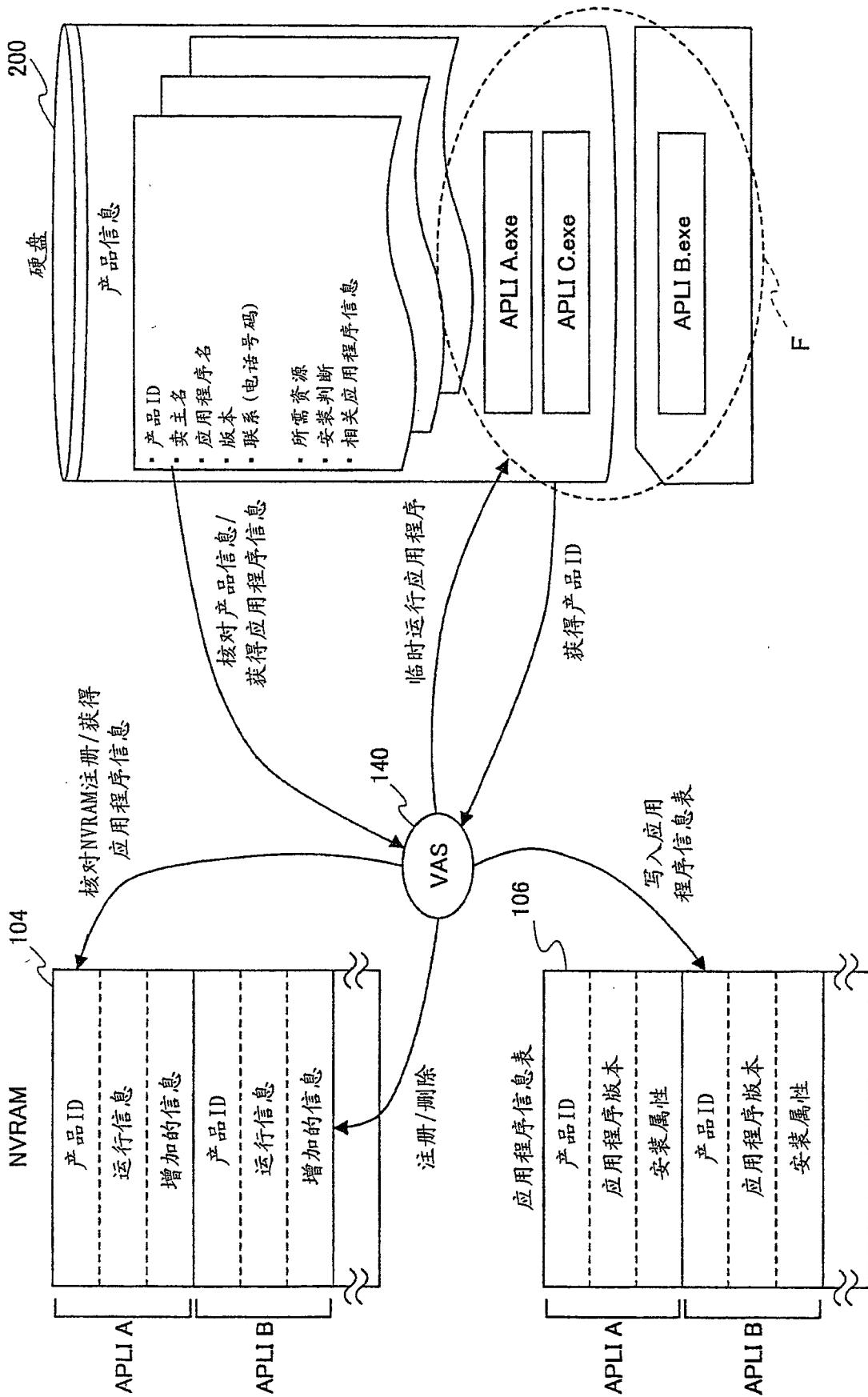


图 4

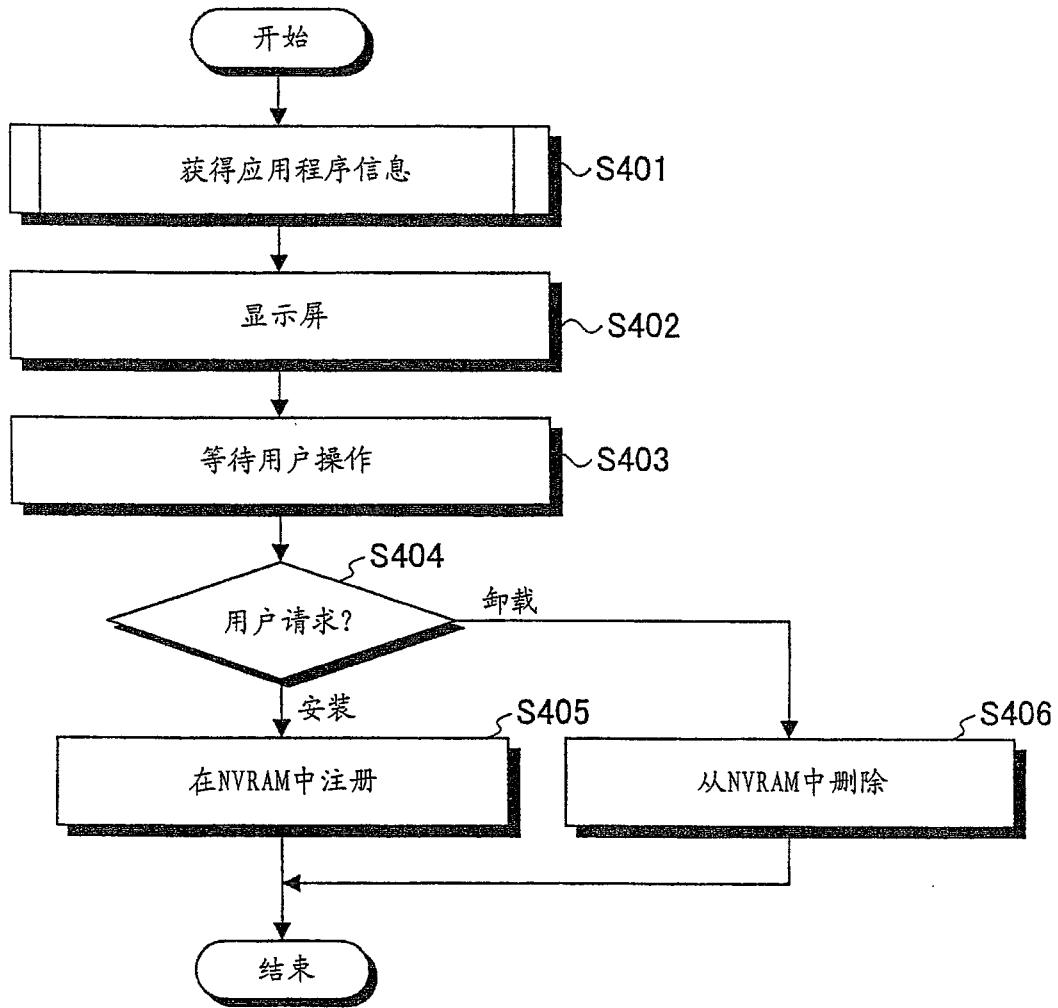


图 5

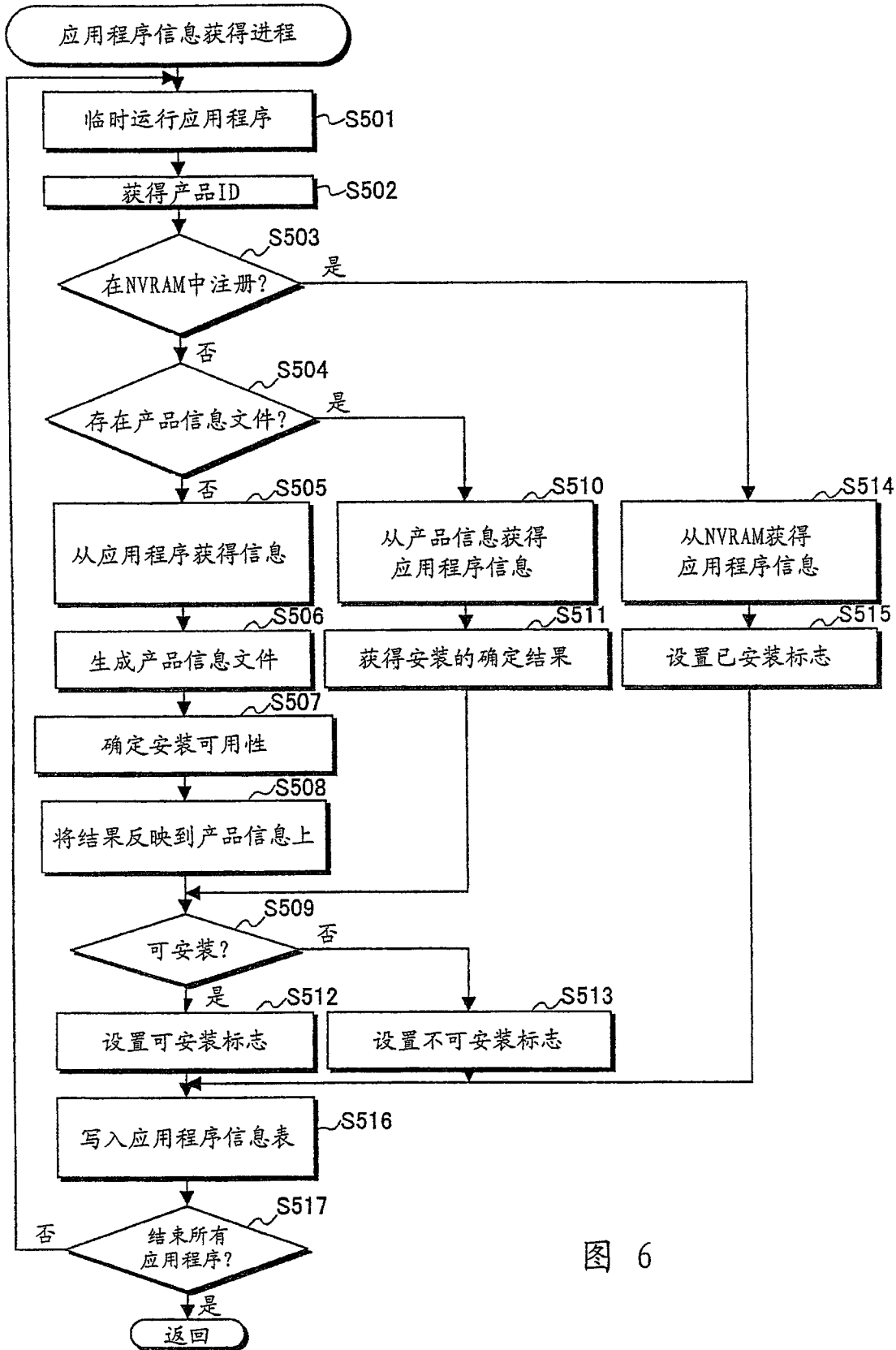


图 6

```
int main(int argc, char argv)
{
    if(argc==2) {
        if(strcmp(argv[1], "-v")==0)
            // 执行初始运行
            if(ConnectVAS()==OK) { // 准备与VAS通信
                // 提供应用程序信息
                SetAplInfo();
                // 通信结束
                Close
            }
        exit(1)
    }
    // 正常运行处理
}
```

图 7

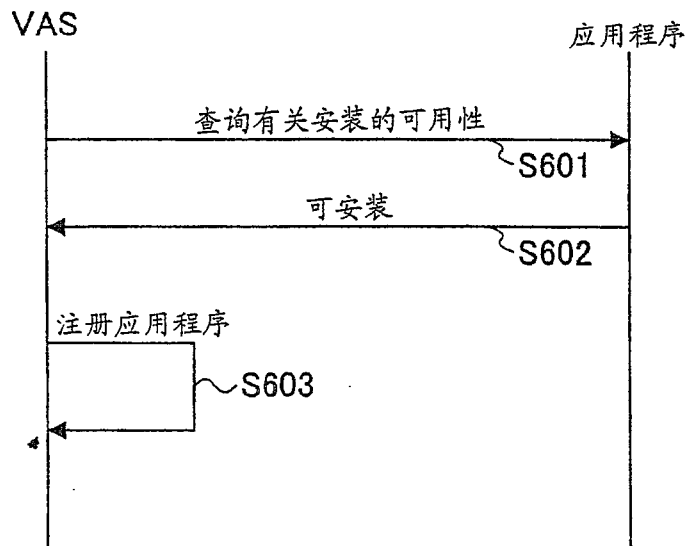


图 8

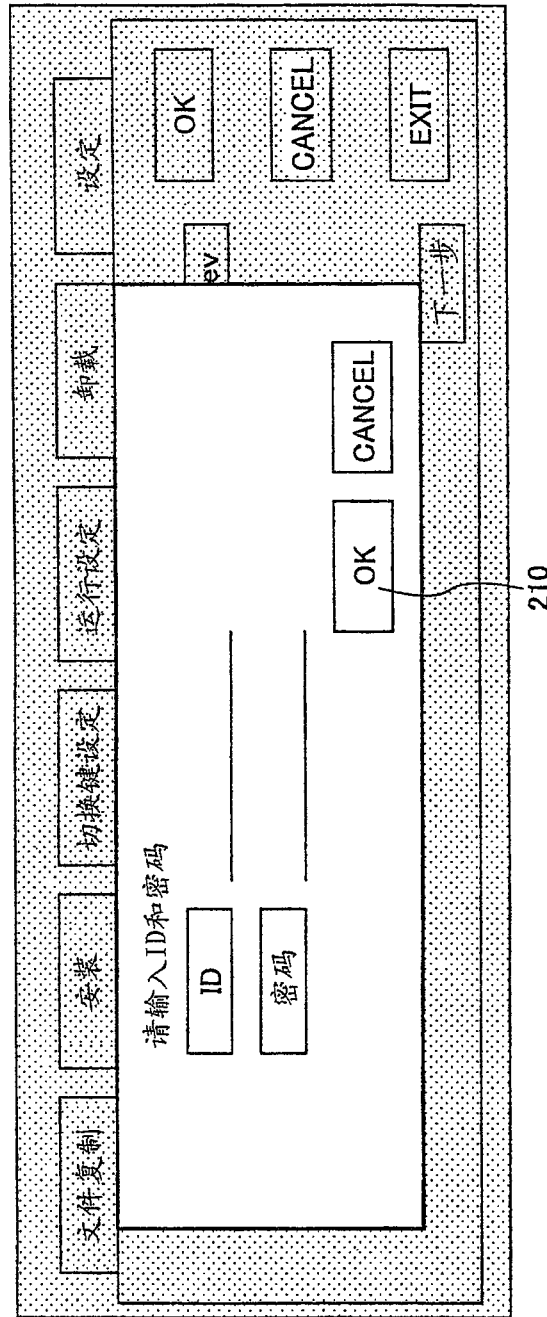


图 9

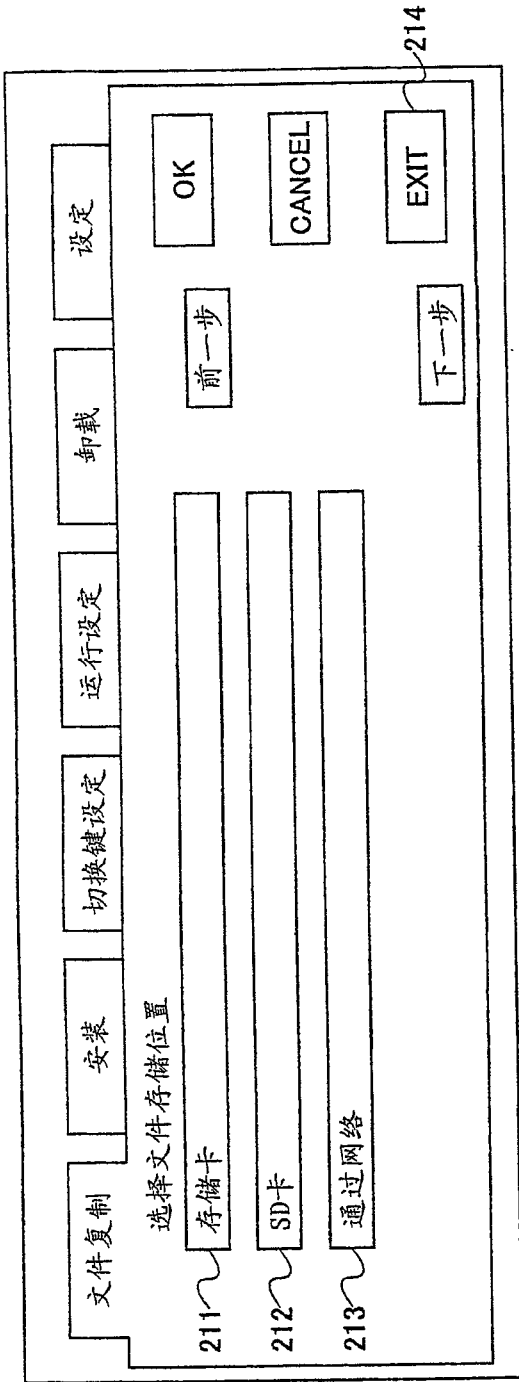


图 10A

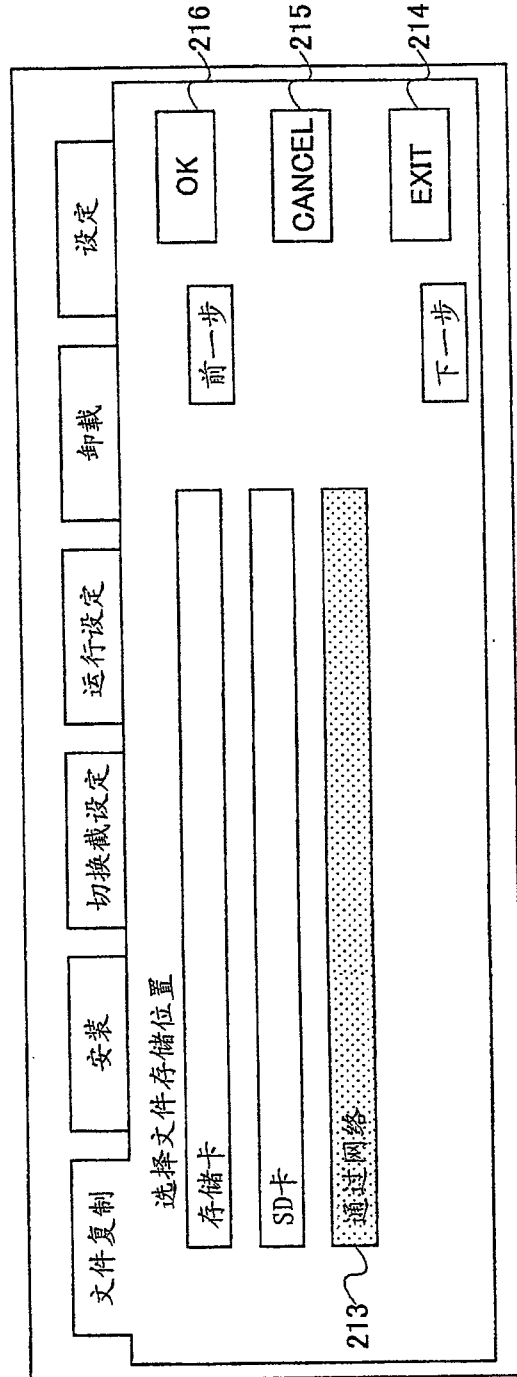


图 10B

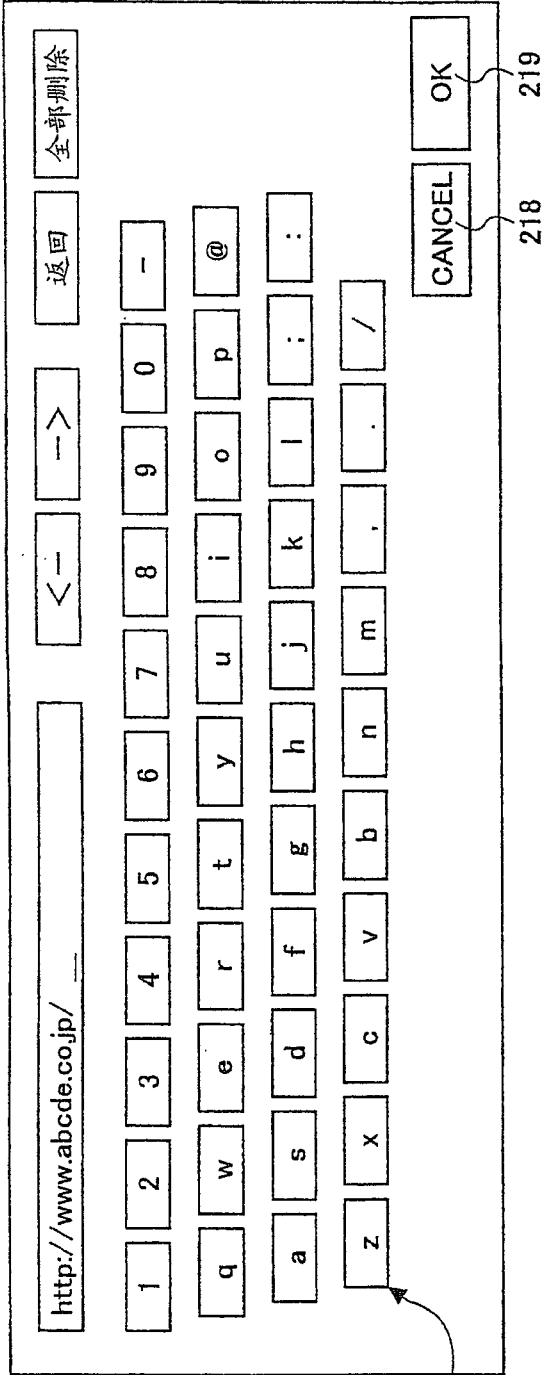


图 11A

217

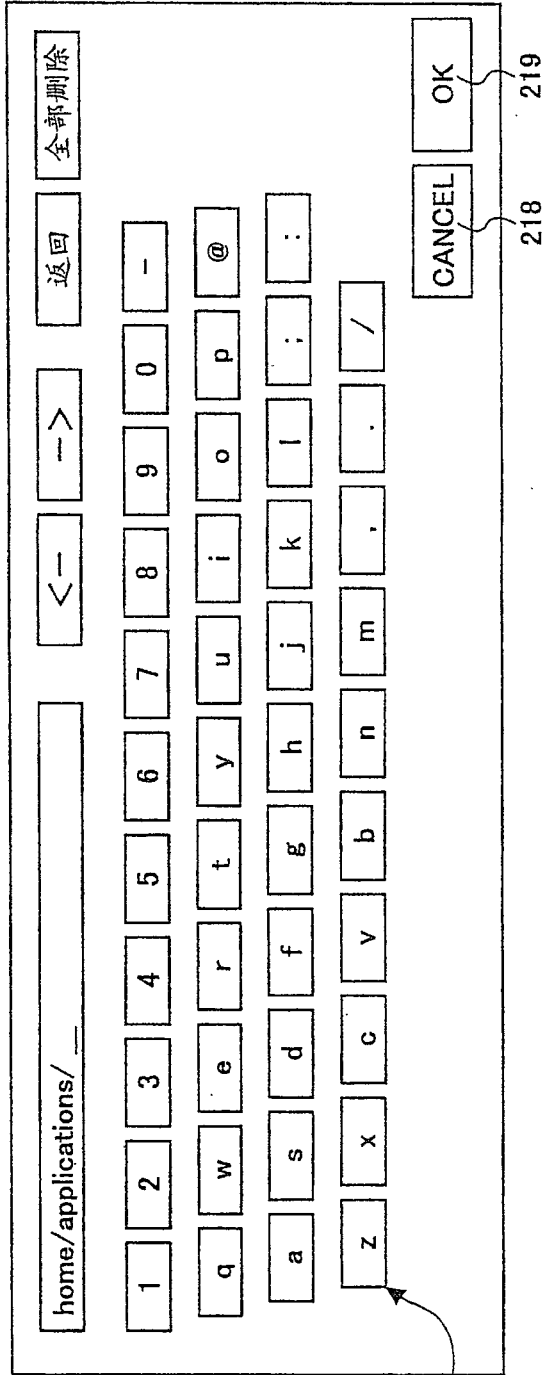


图 11B

217

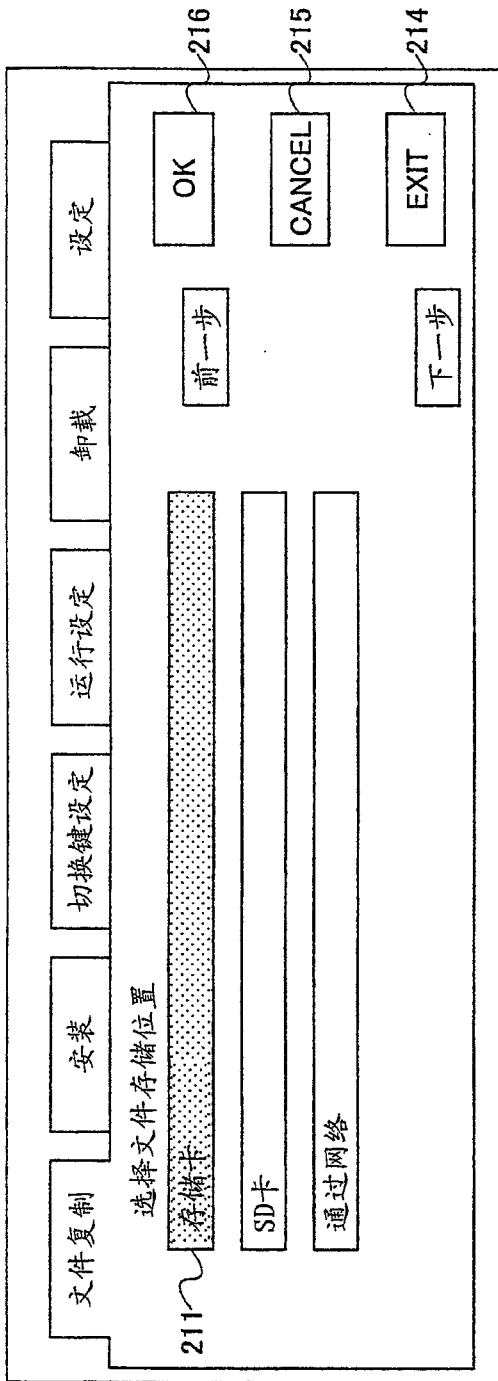


图 12A

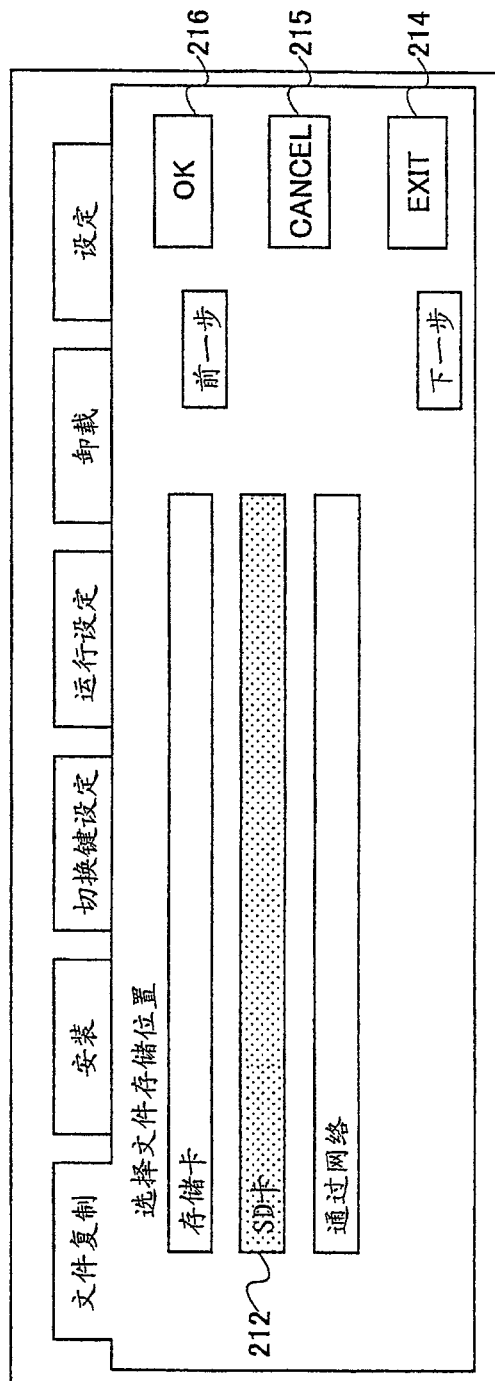


图 12B

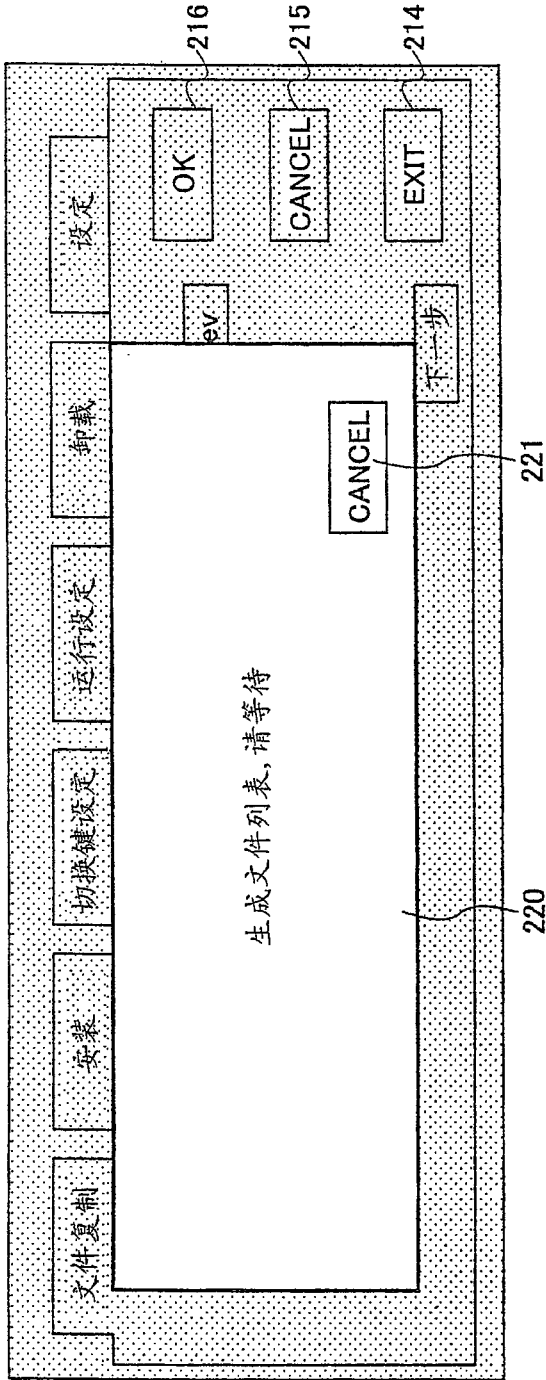


图 13A

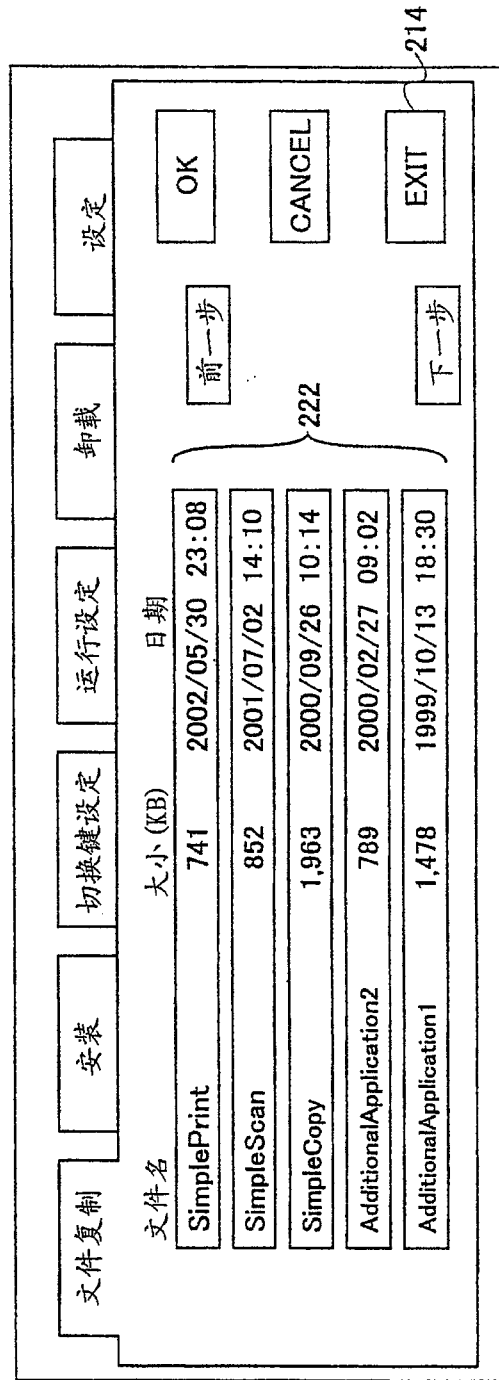


图 13B

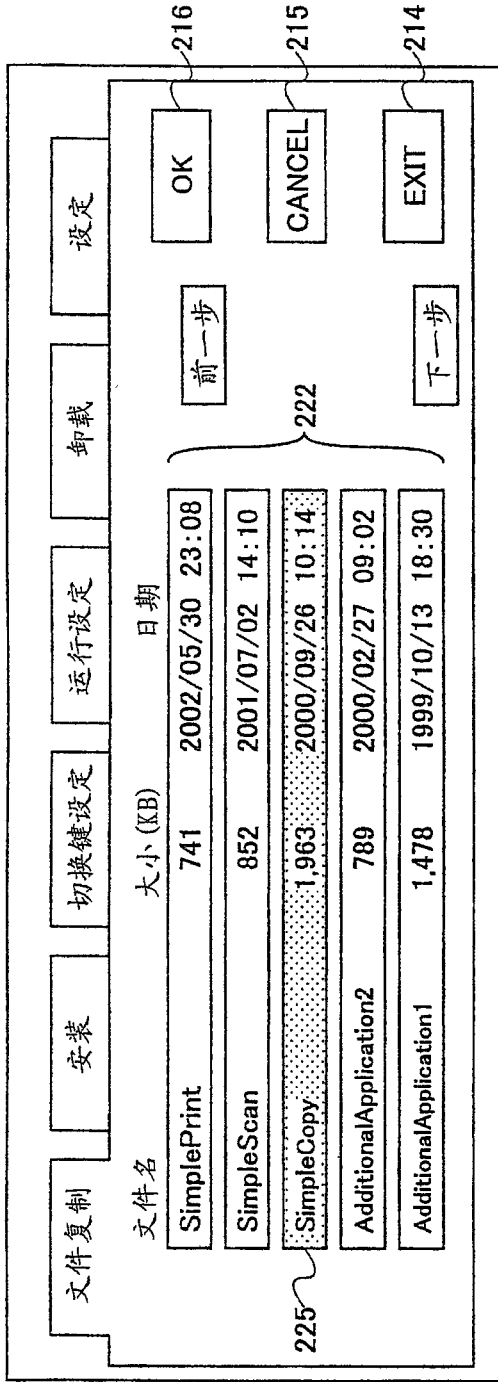


图 14A

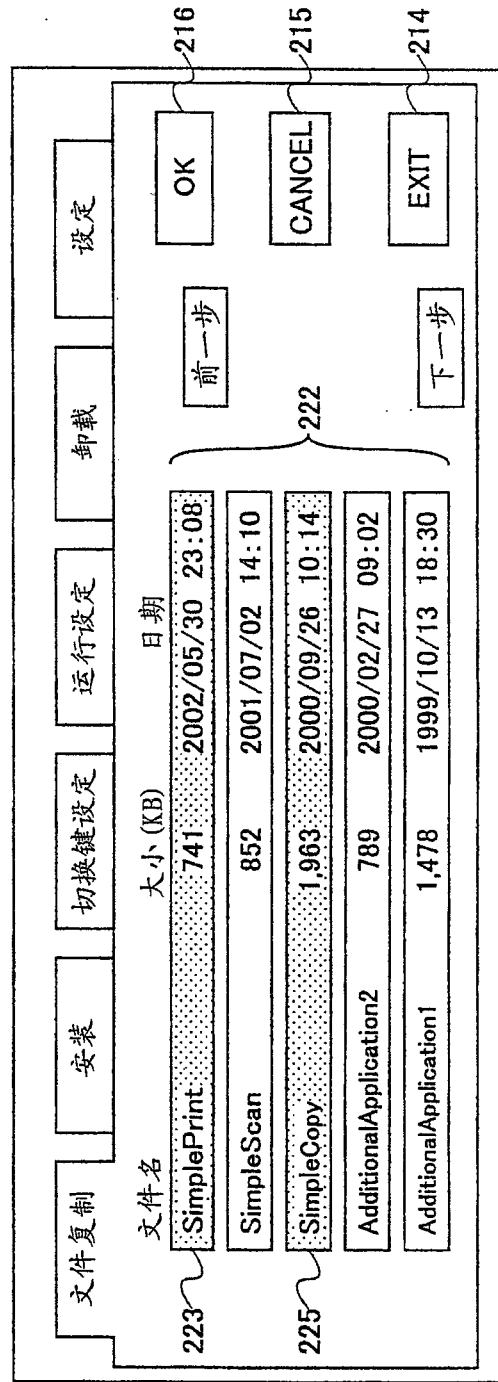


图 14B

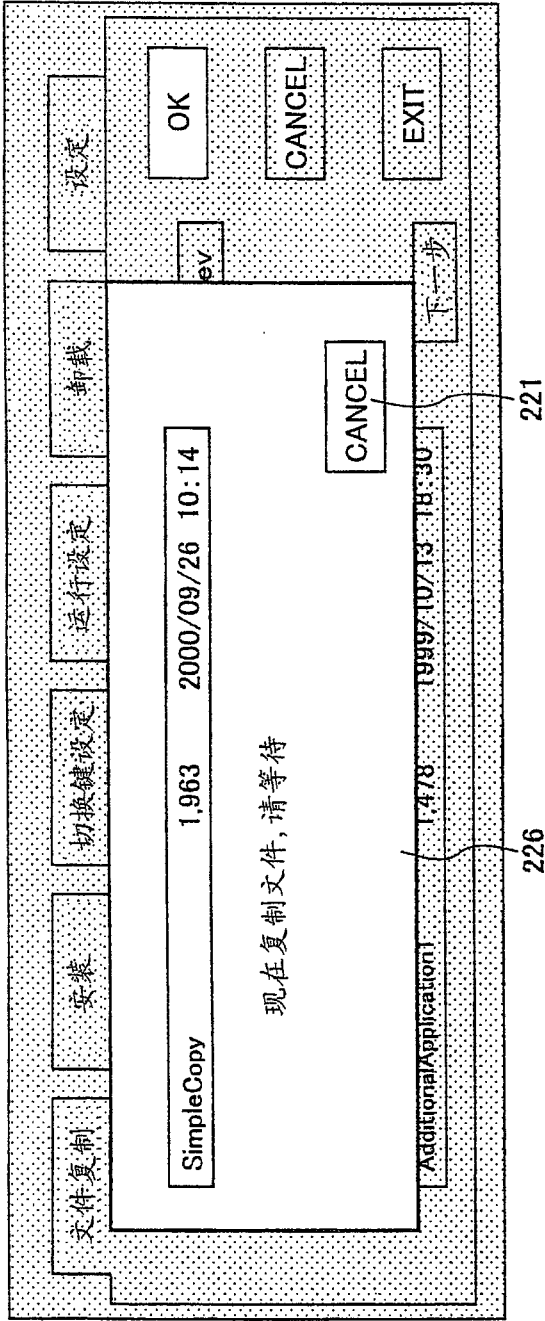


图 15A

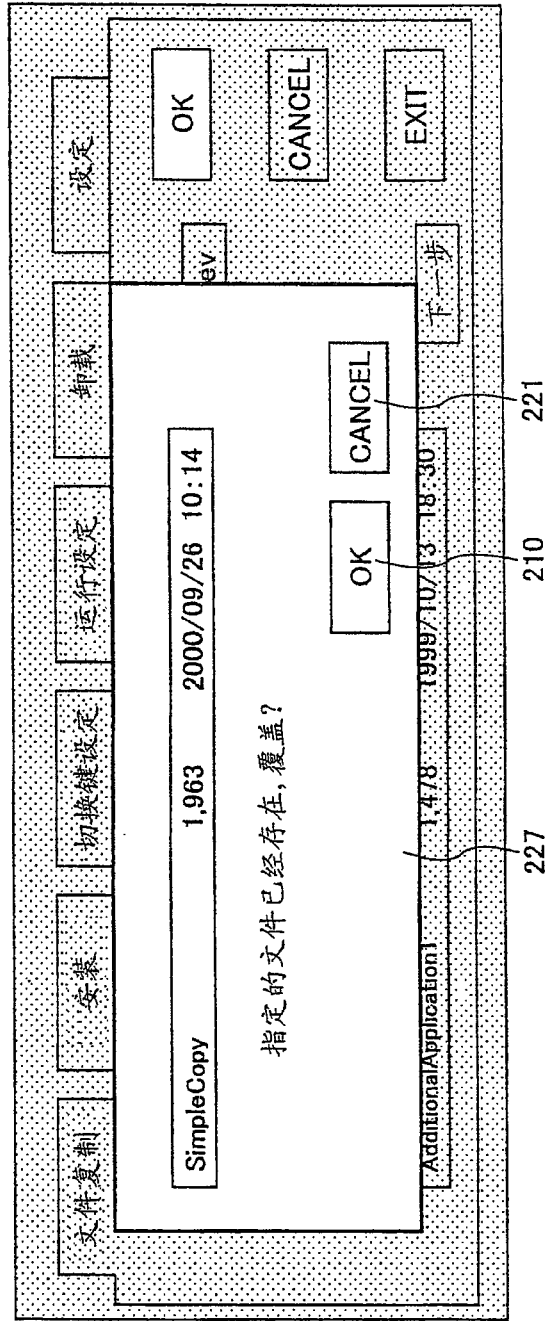


图 15B

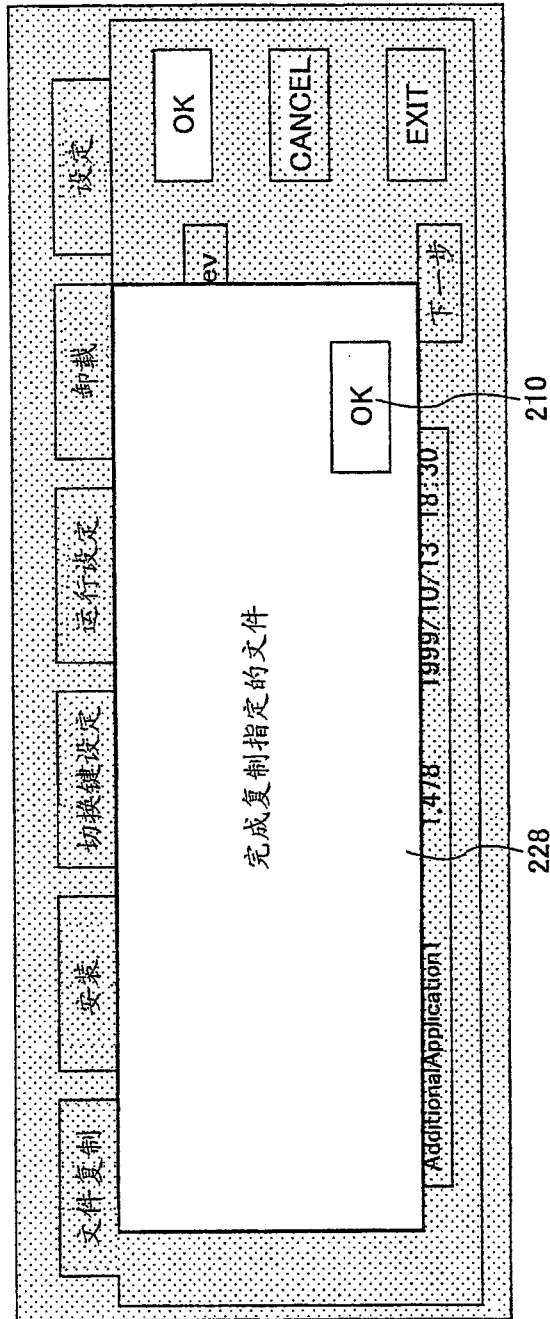


图 16

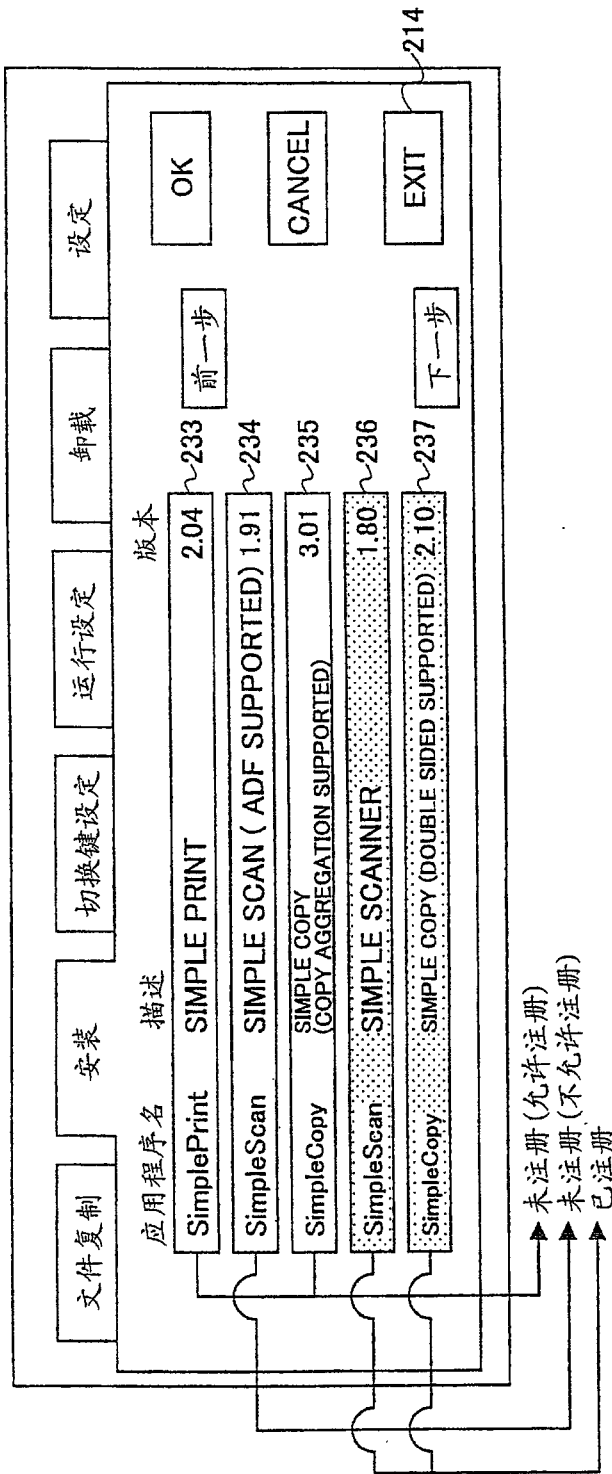


图 17A

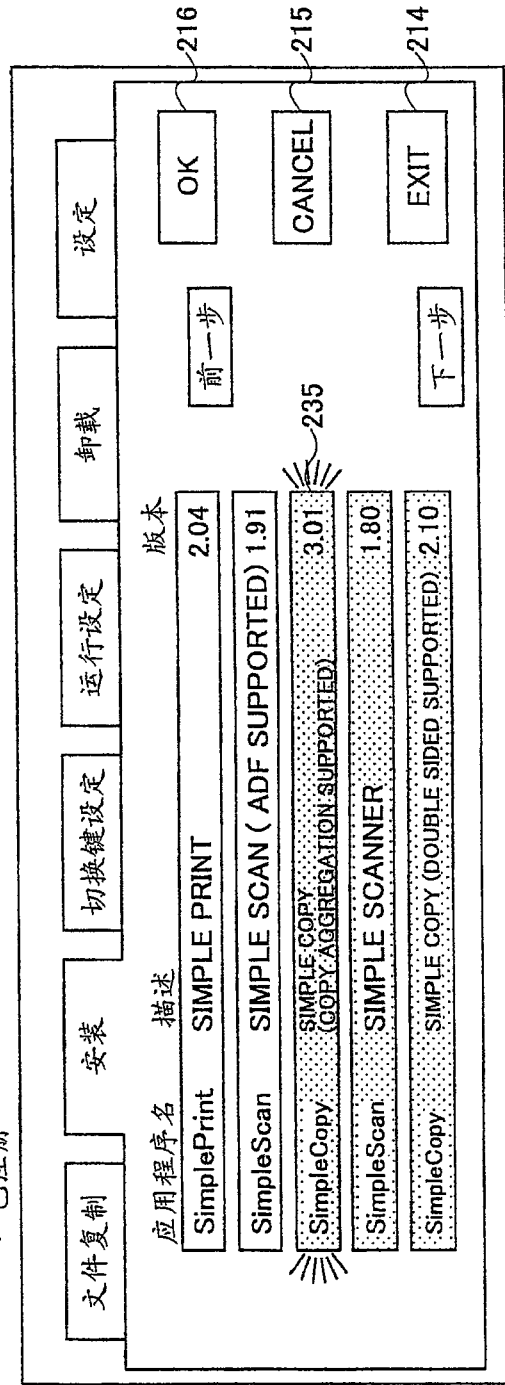


图 17B

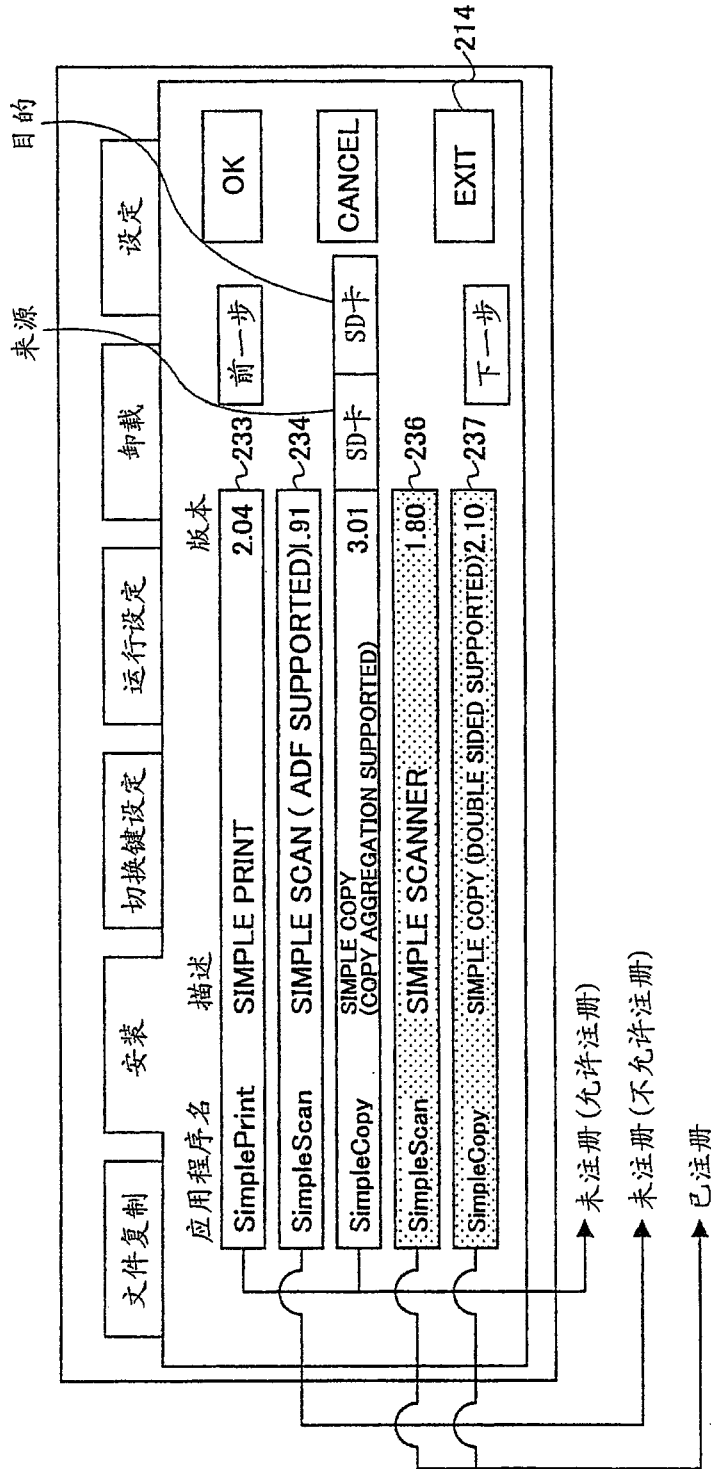


图 18

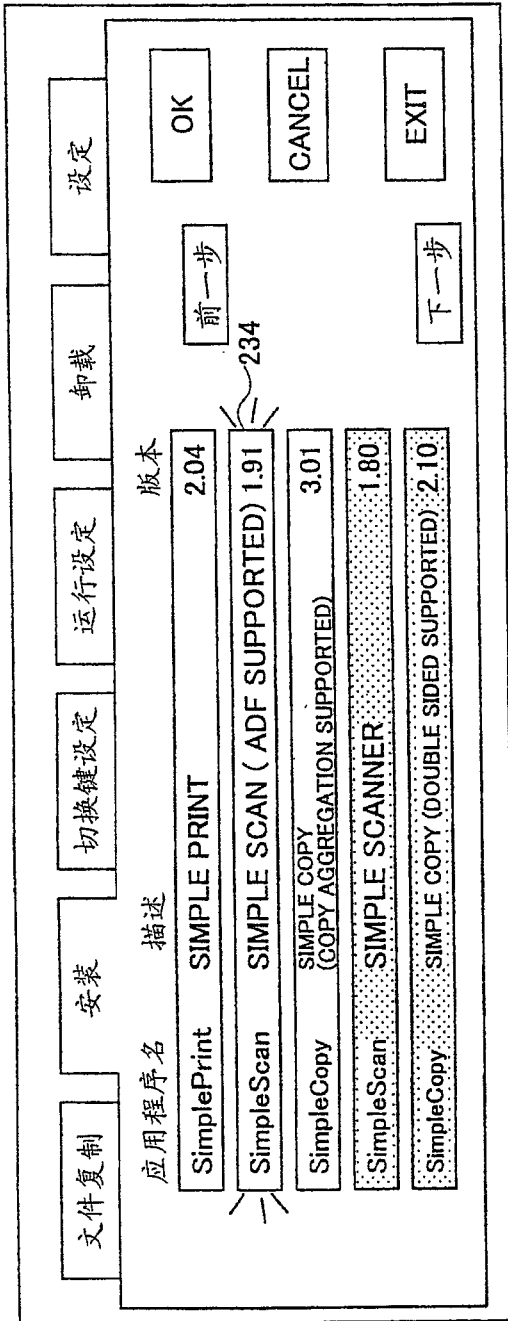


图 19A

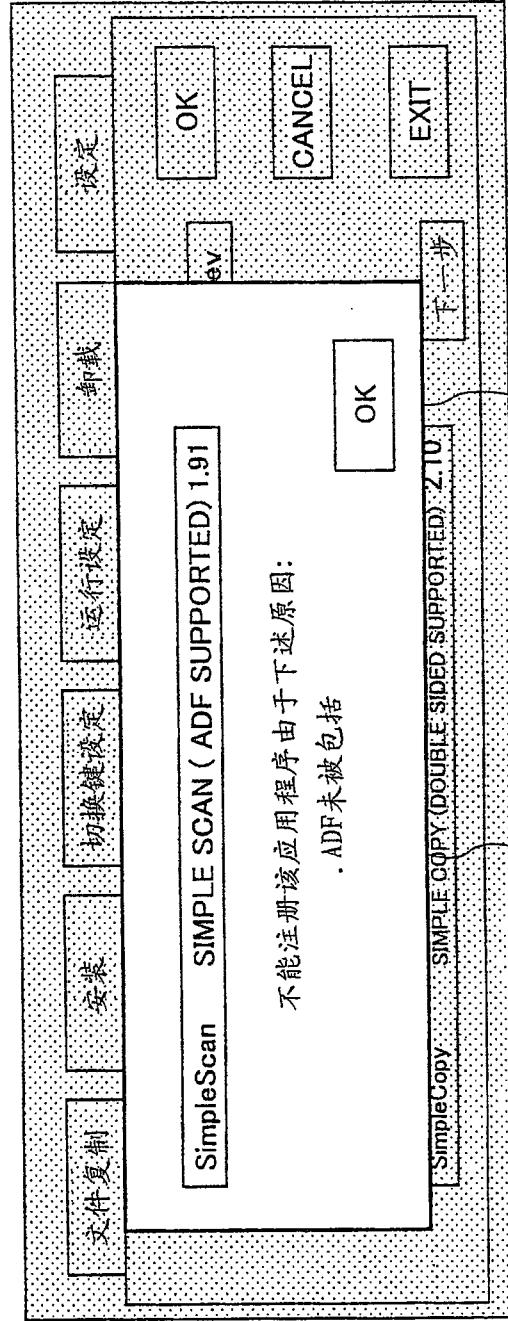


图 19B

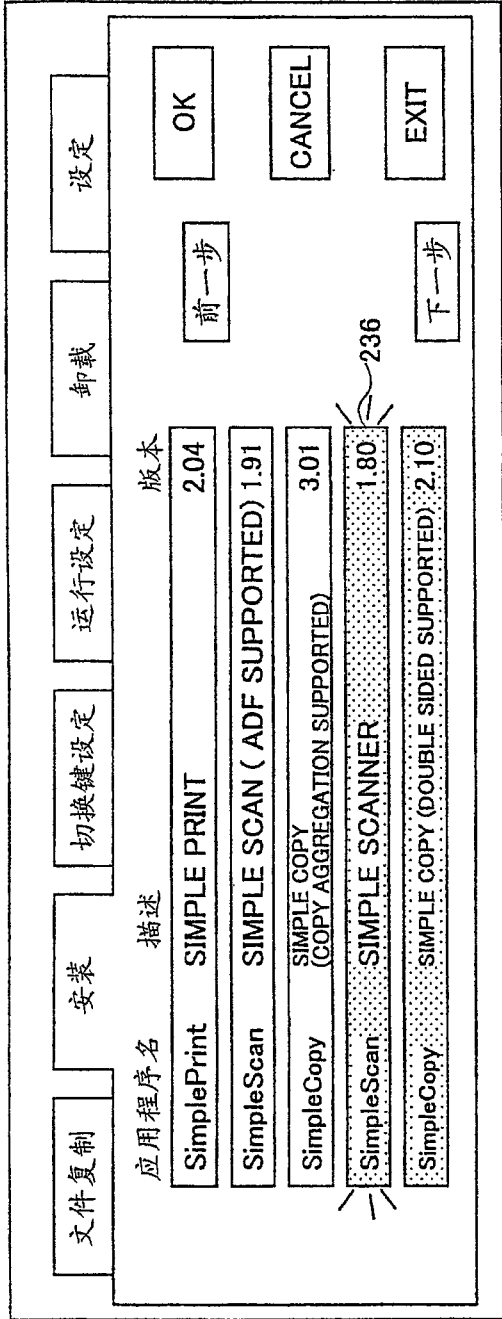


图 20A

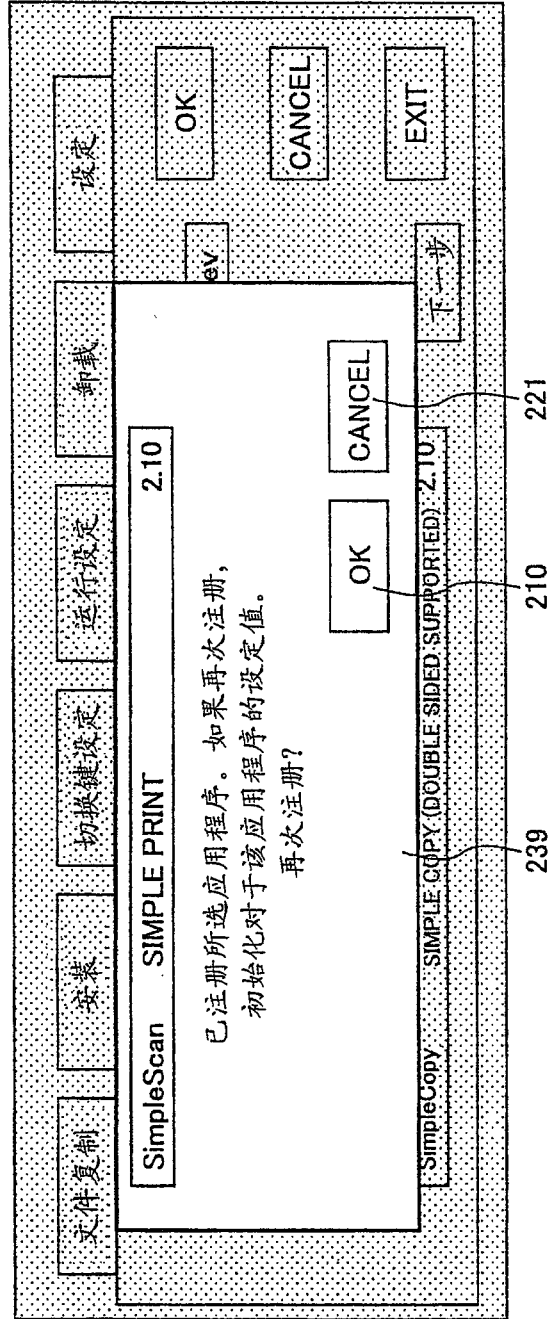


图 20B

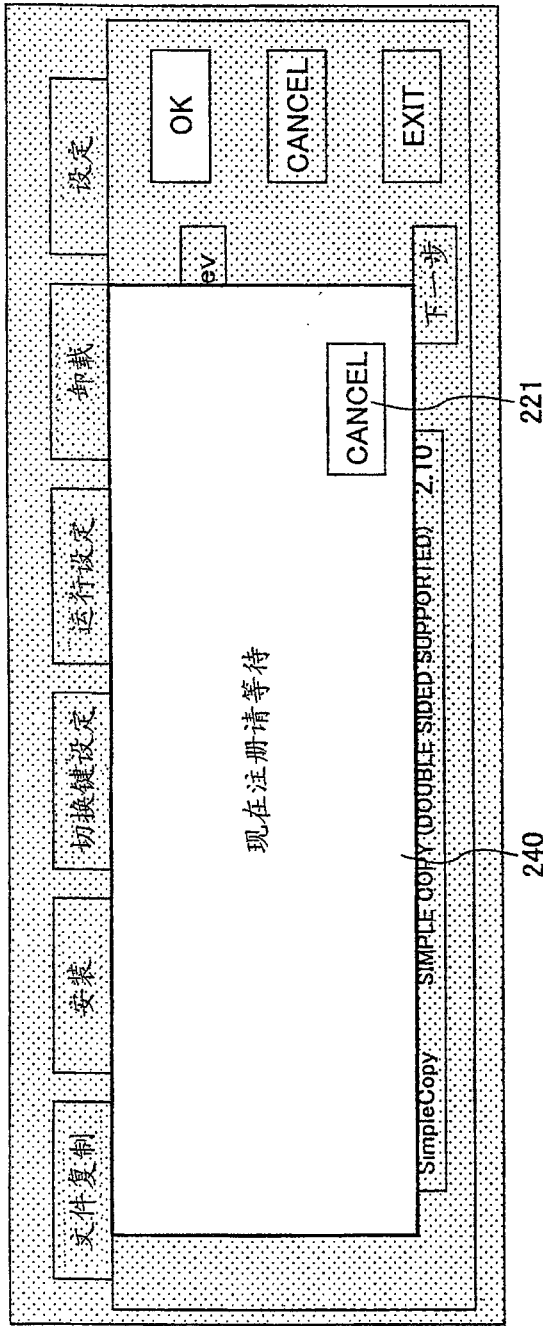


图 21A

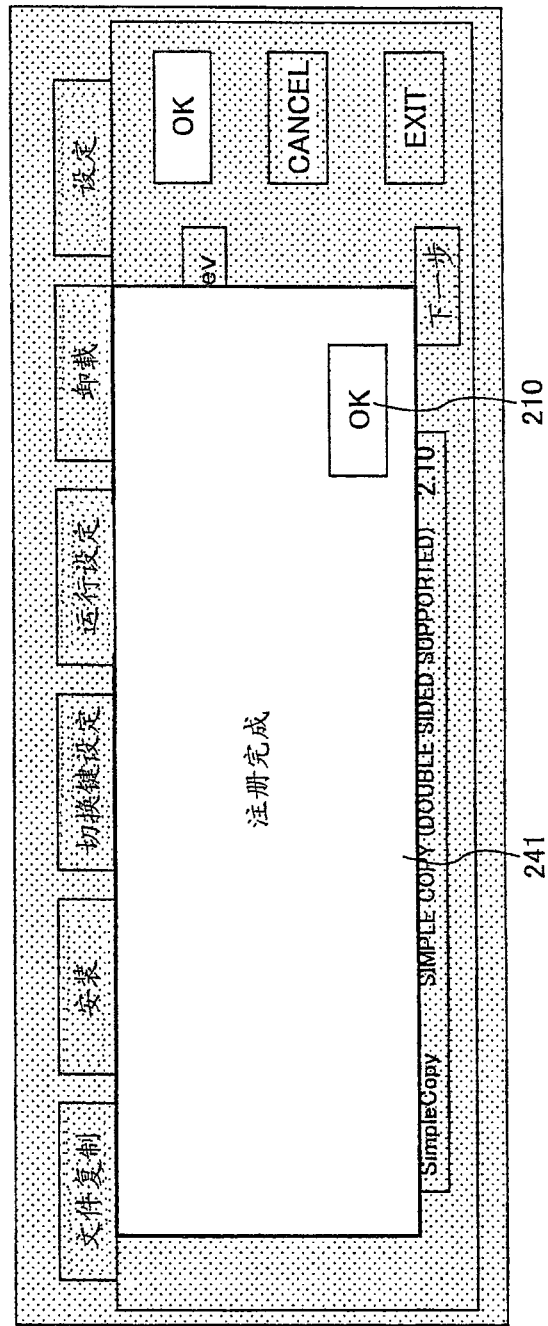


图 21B

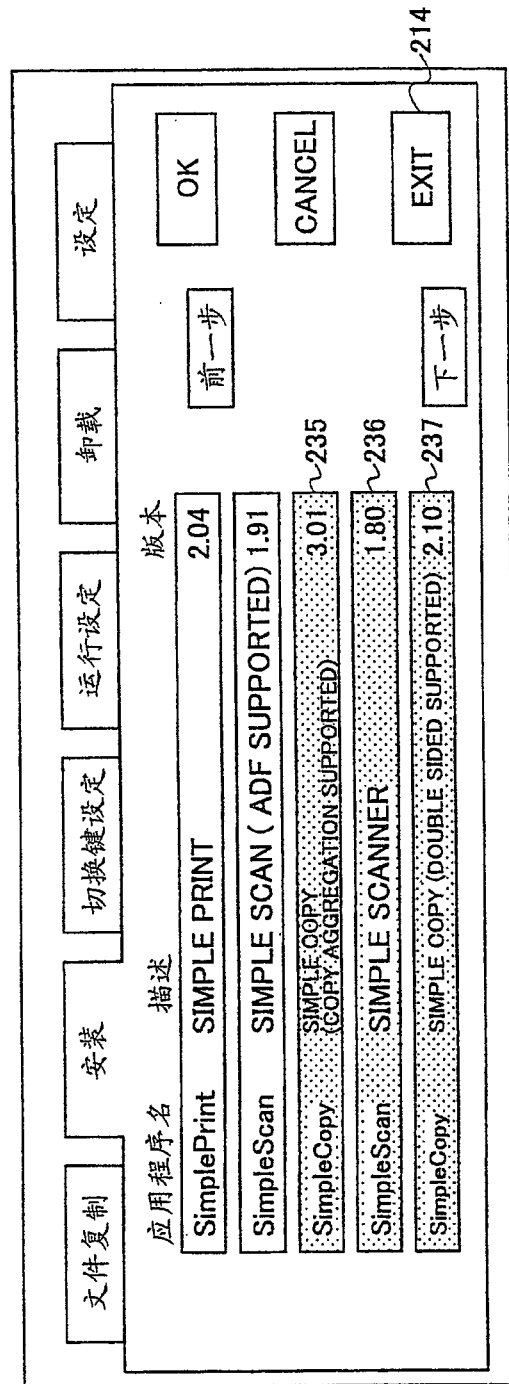


图 22

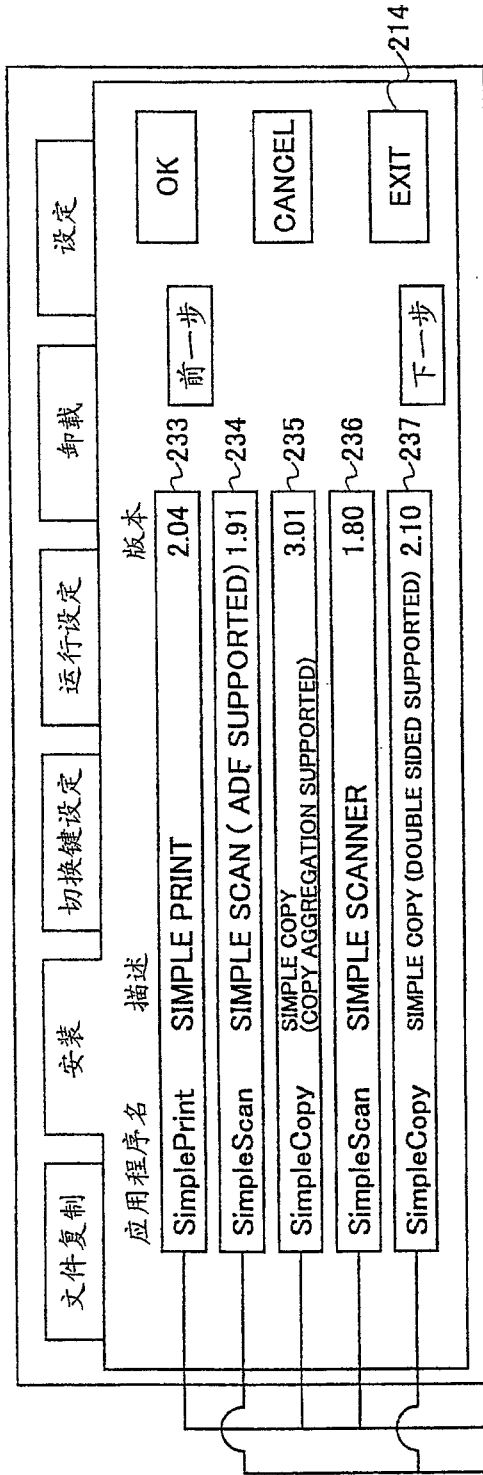


图 23A

未注册 (允许注册)

已注册或未注册 (不允许注册)

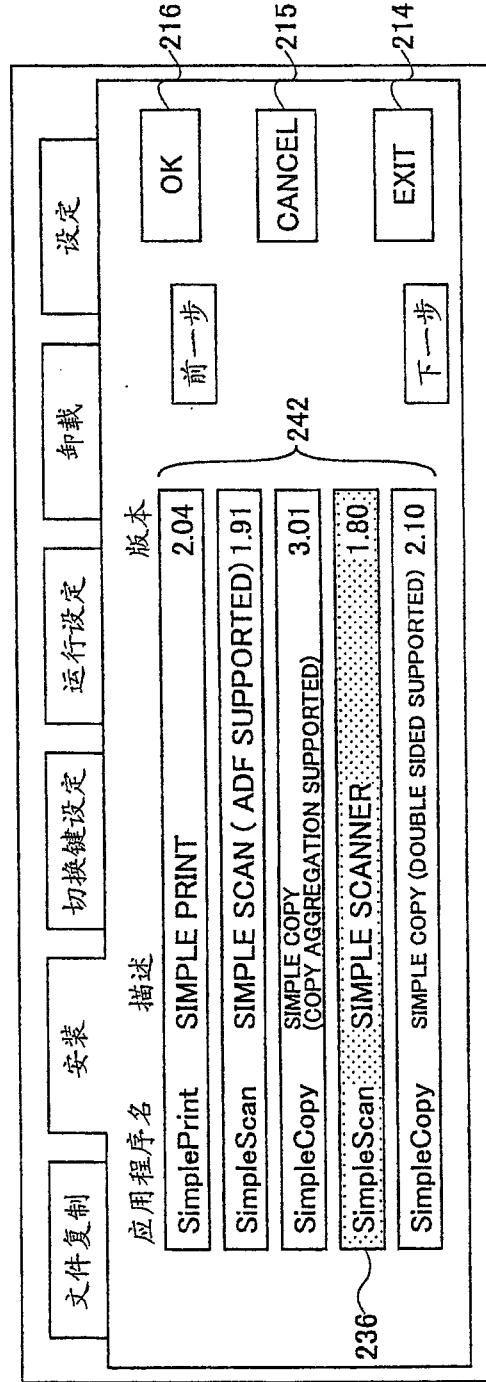


图 23B

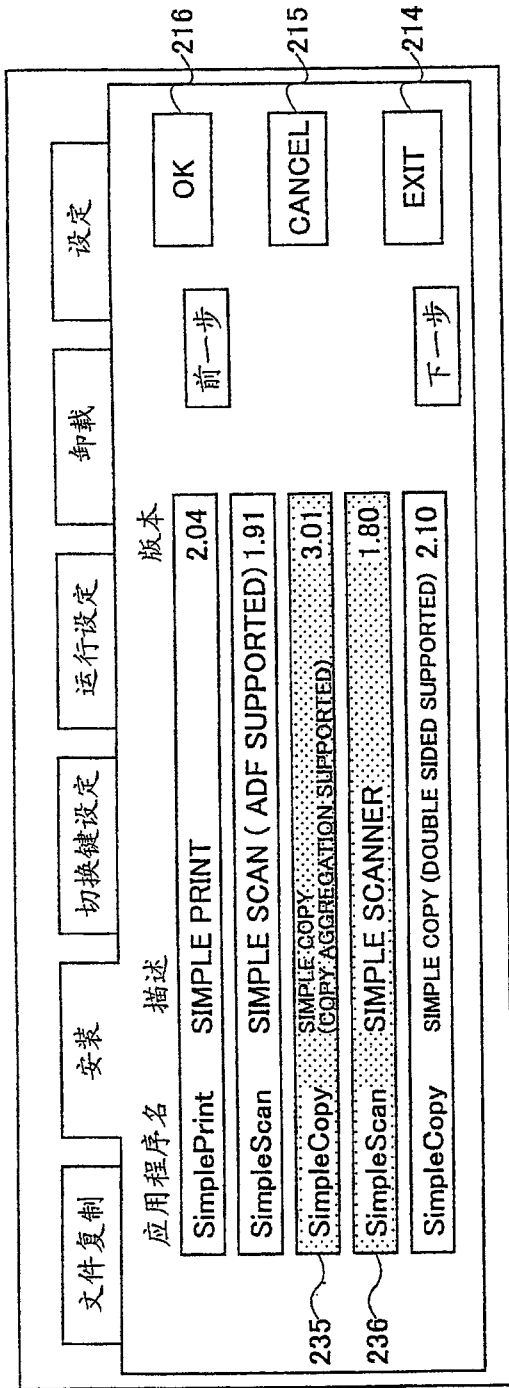


图 24A

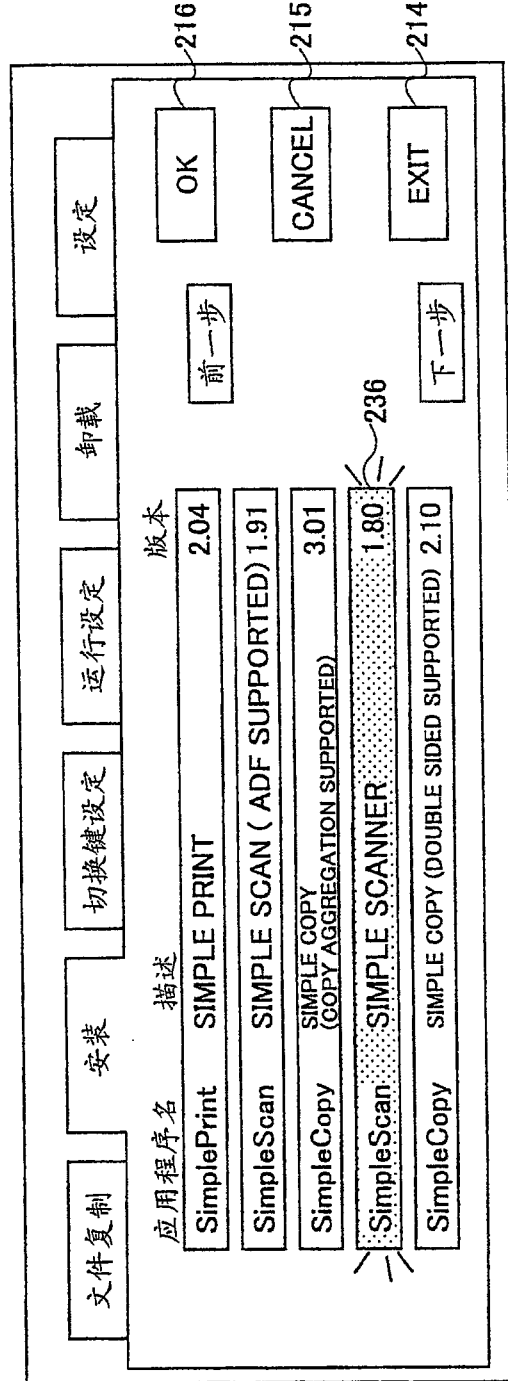


图 24B

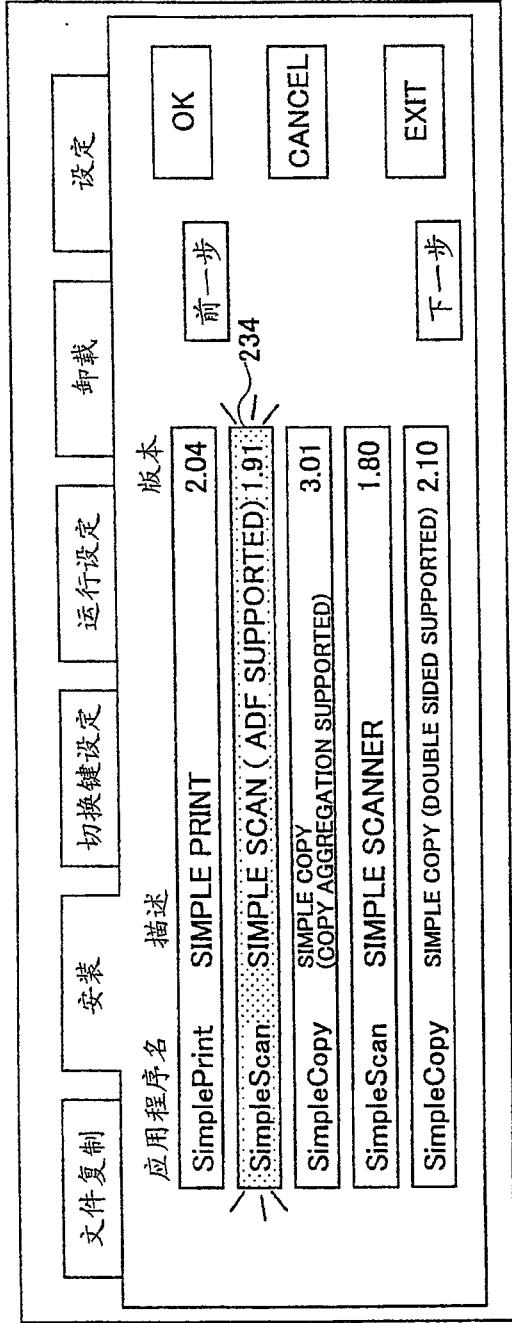


图 25A

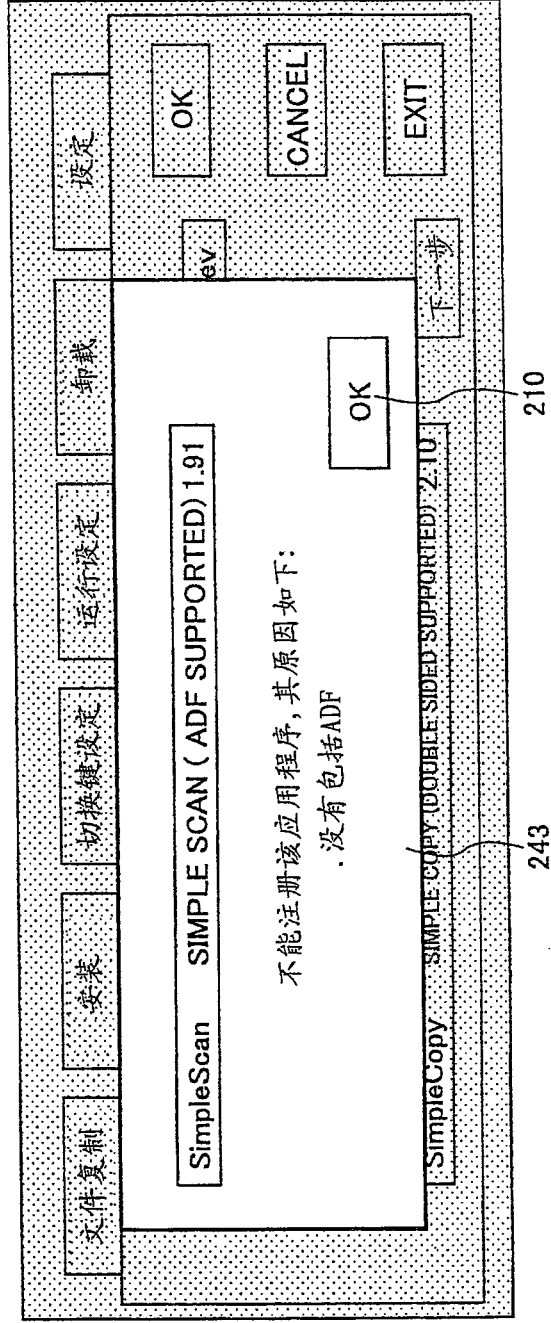


图 25B

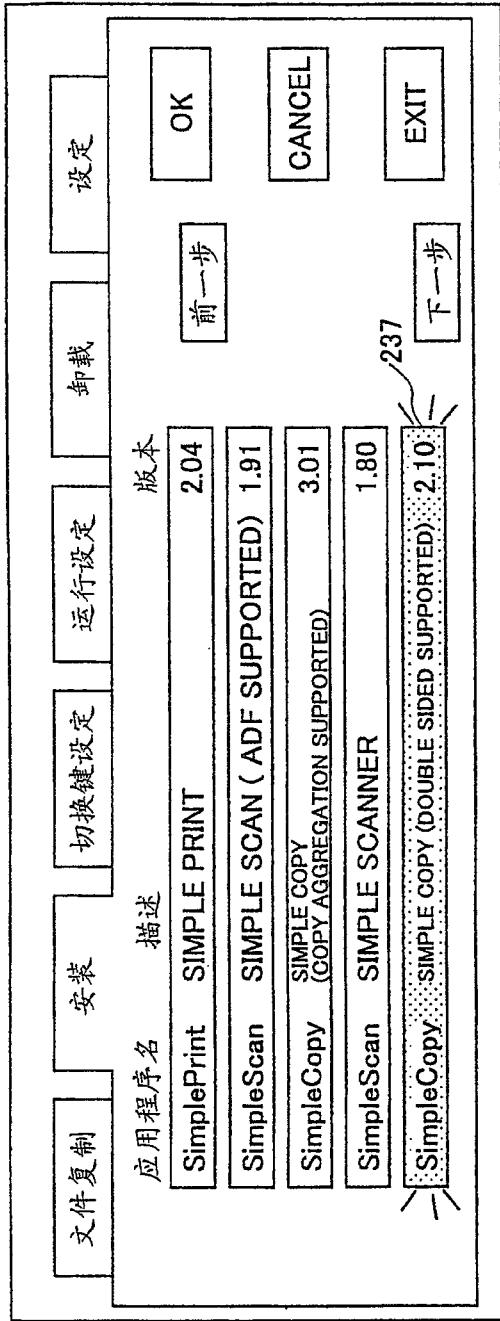


图 26A

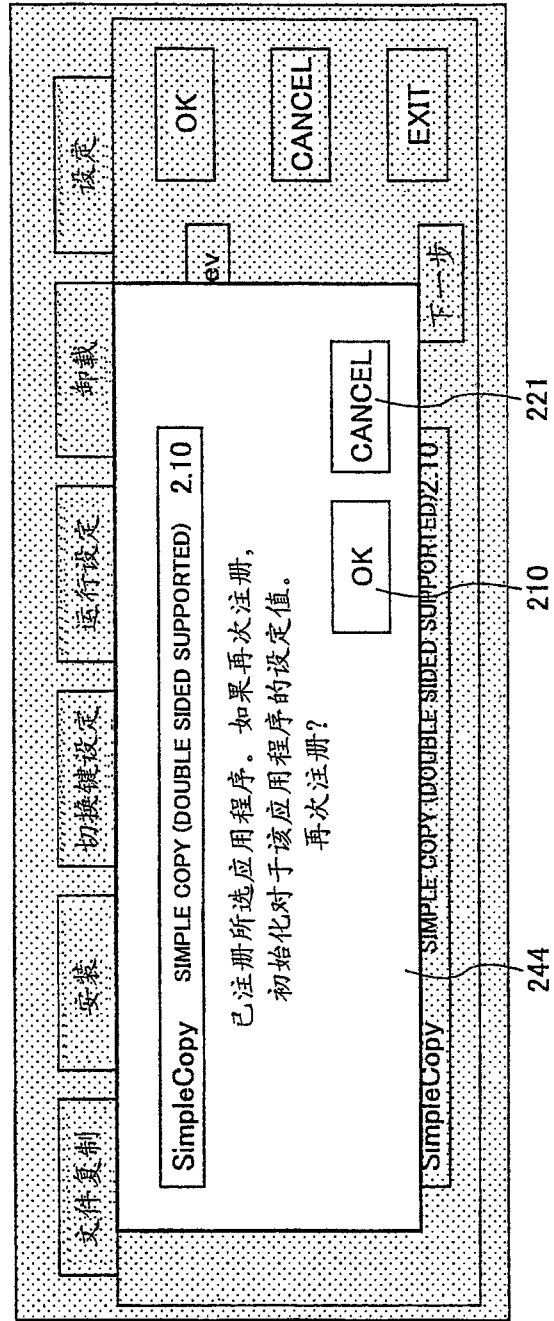


图 26B

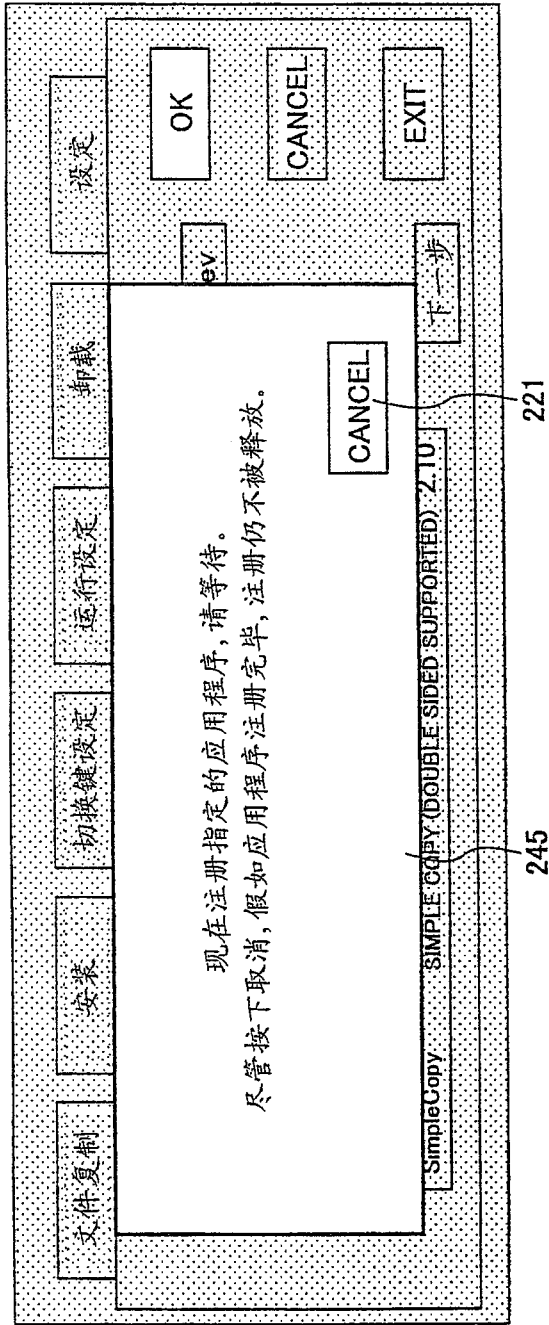


图 27A

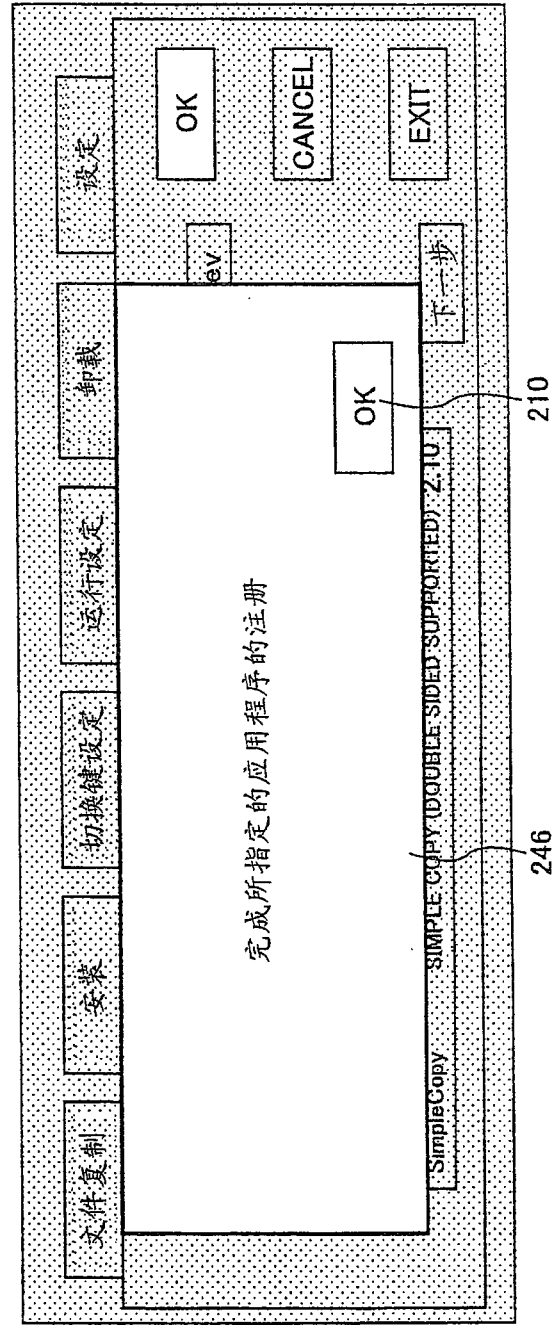


图 27B

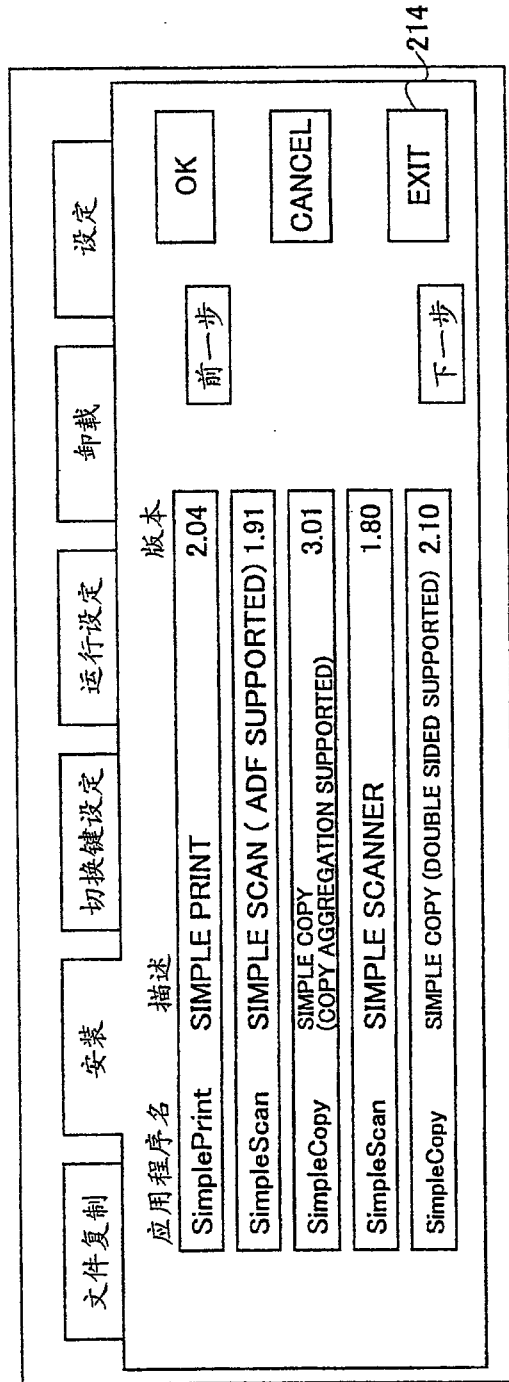


图 28

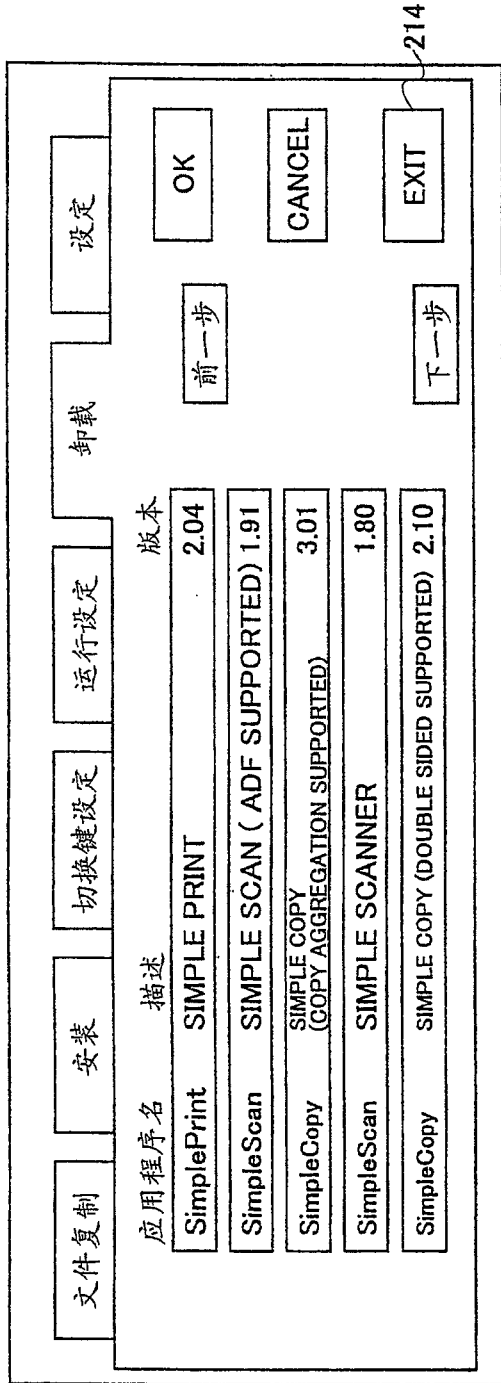


图 29A

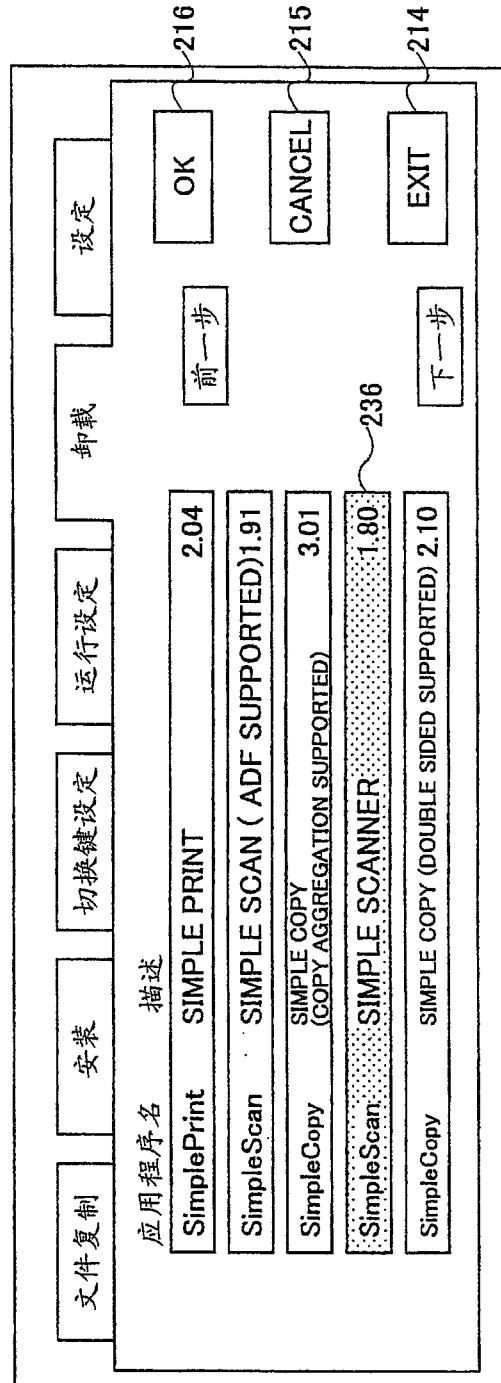


图 29B

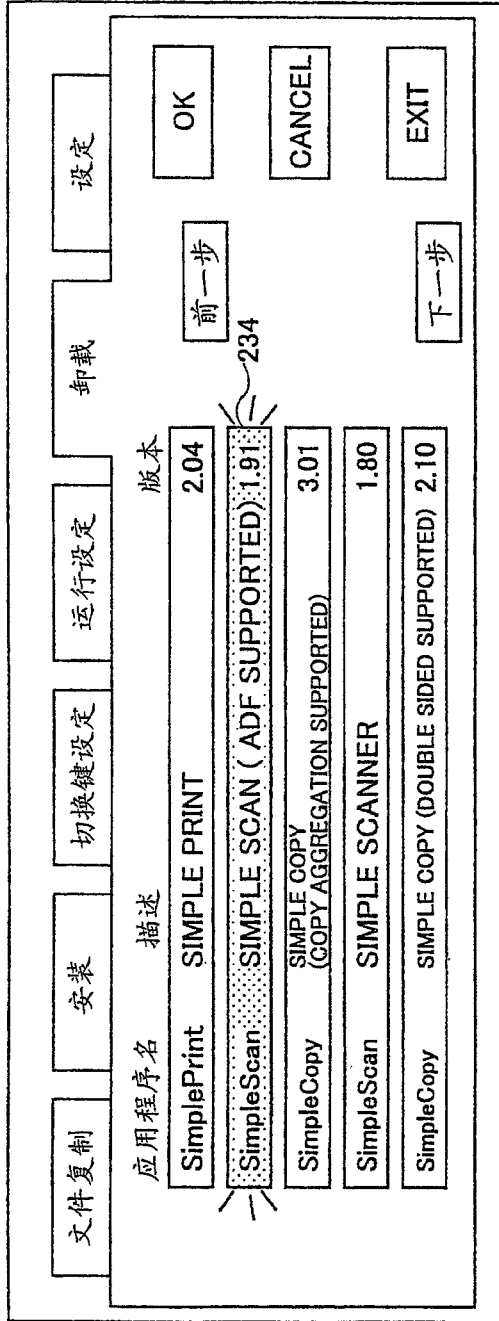


图 30A

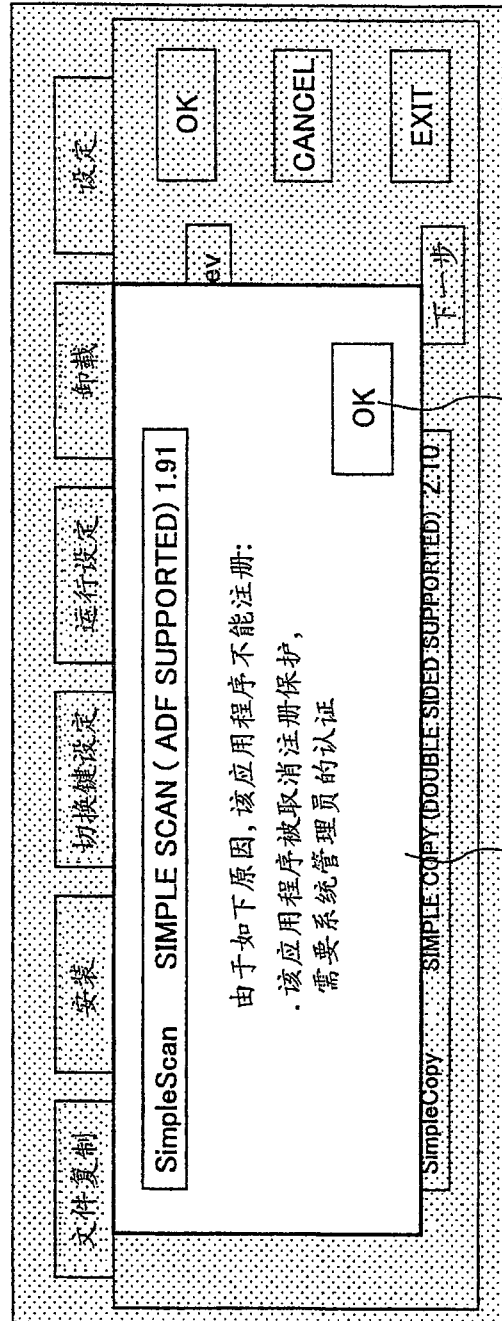


图 30B

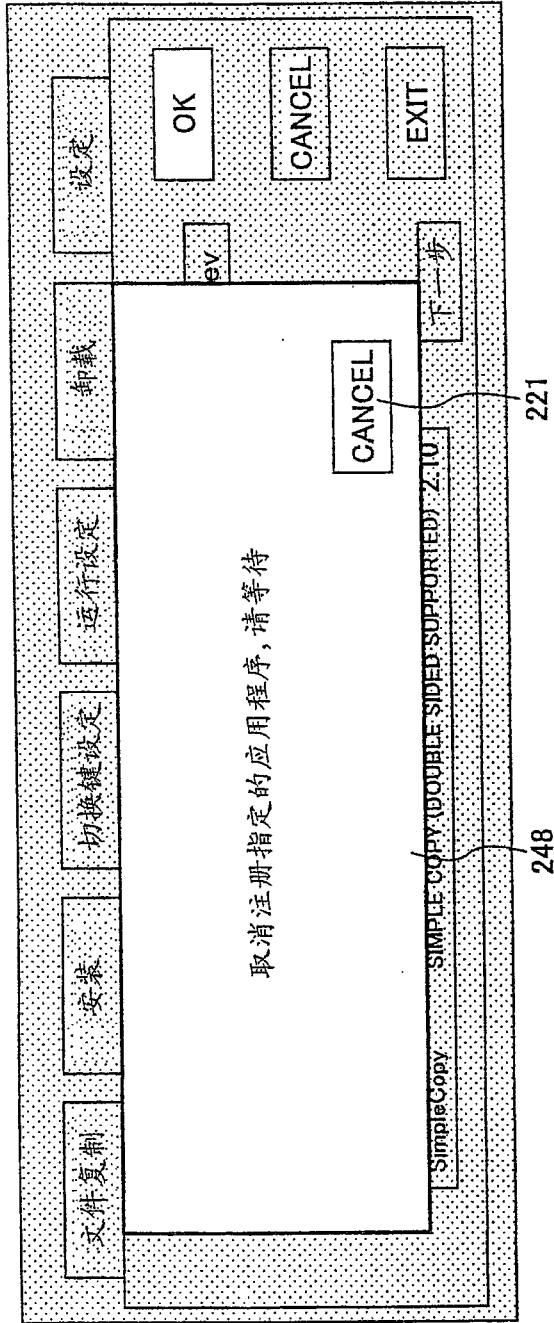


图 31A

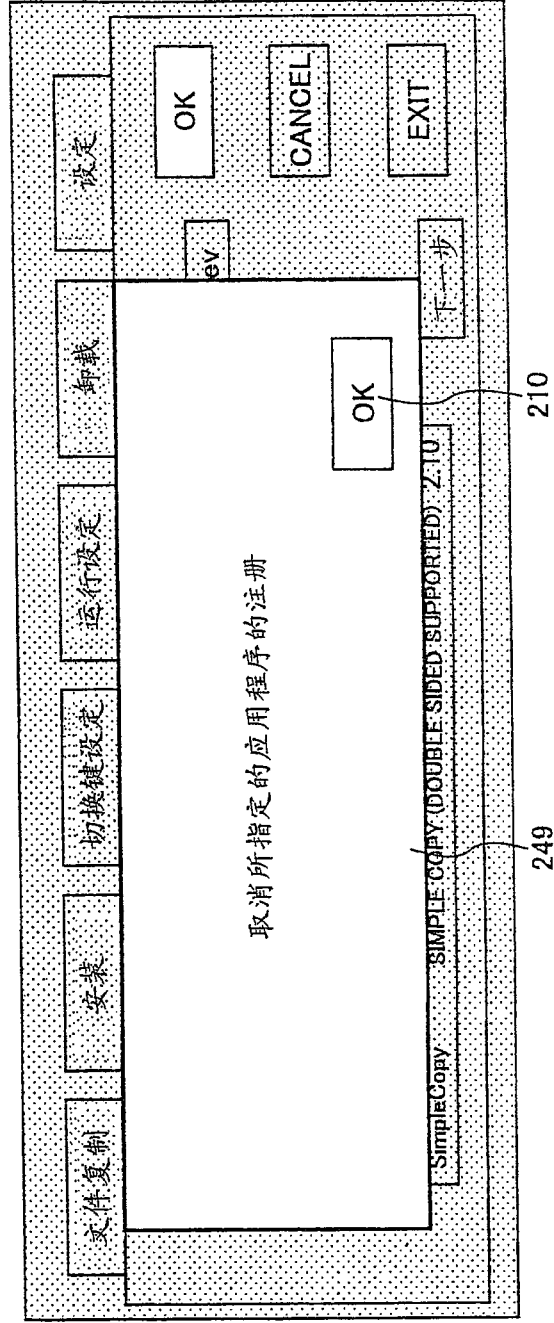


图 31B

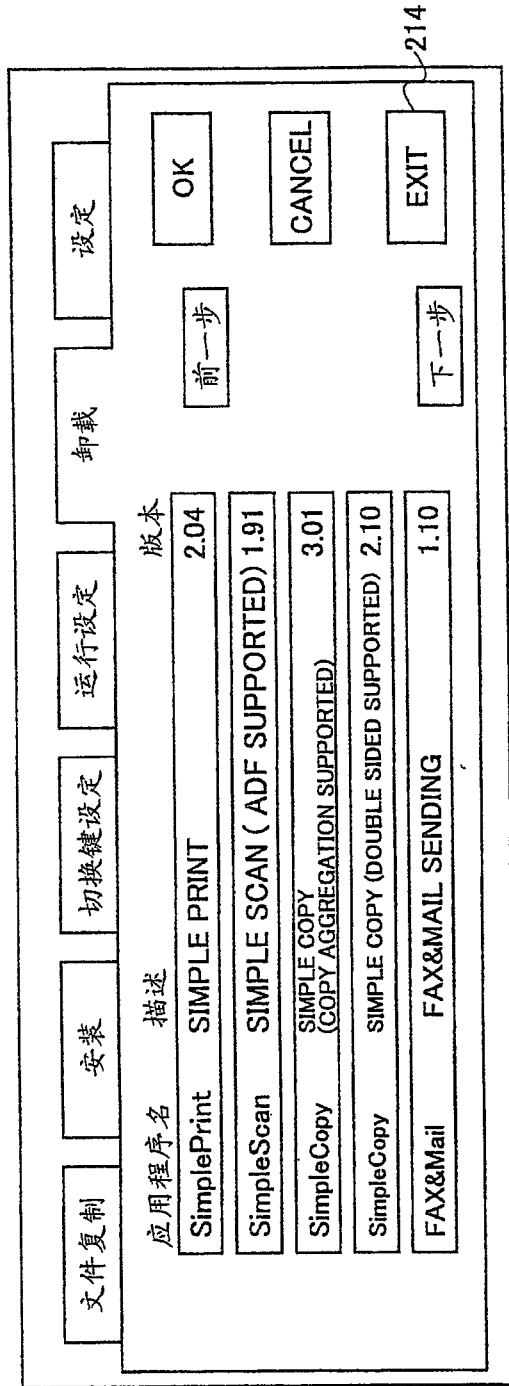


图 32

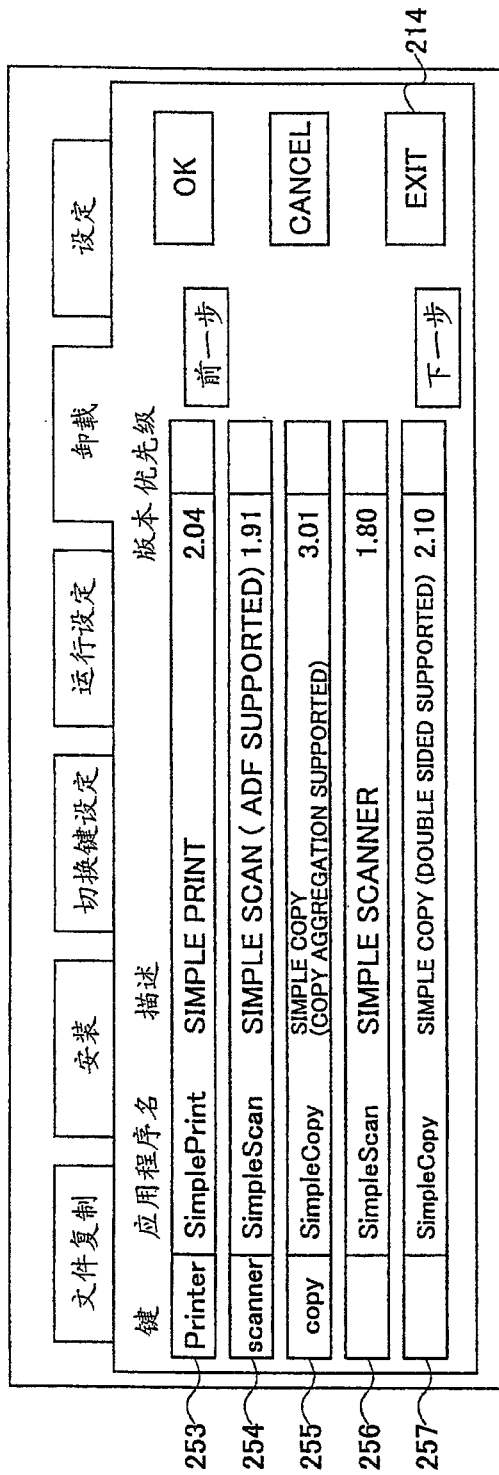


图 33A

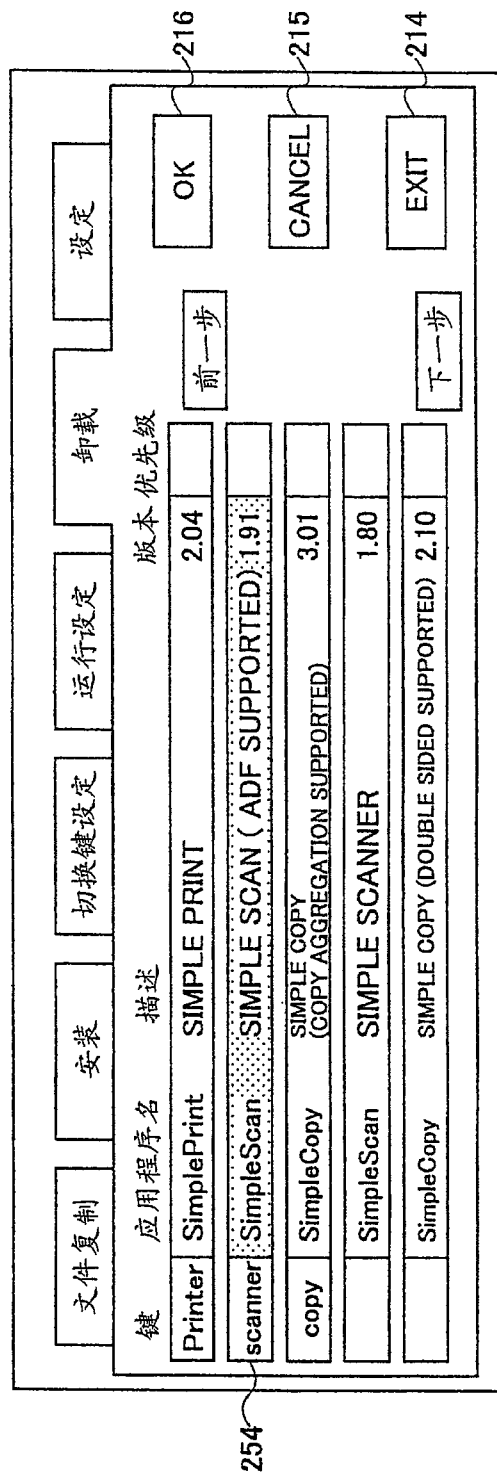


图 33B

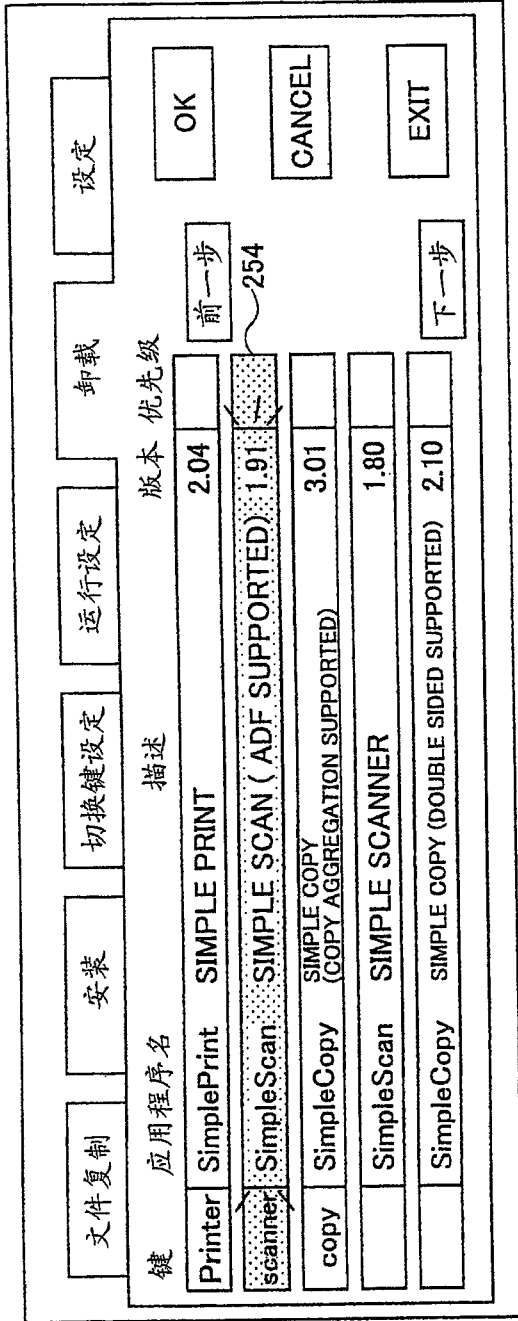


图 34A

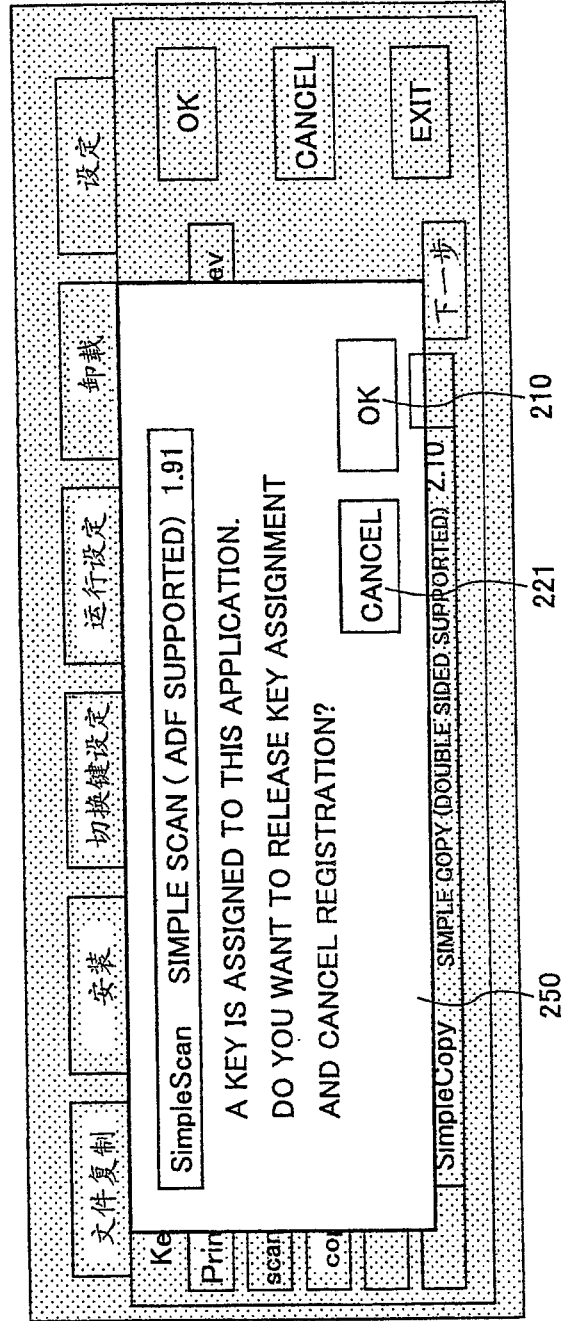


图 34B

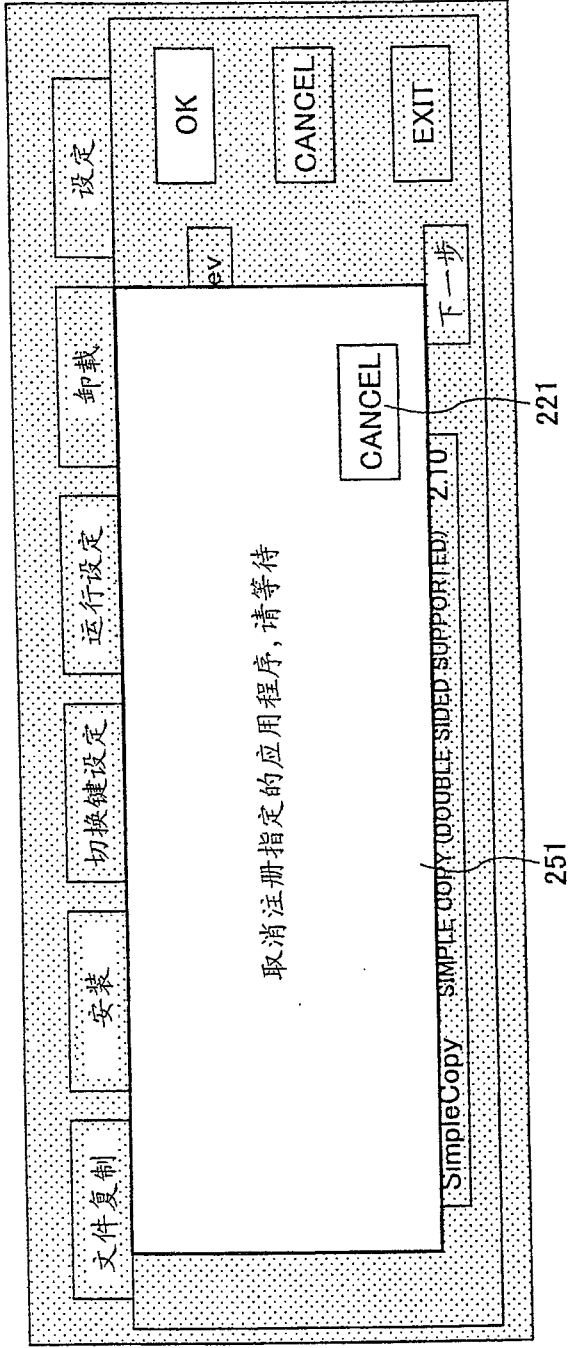


图 35A

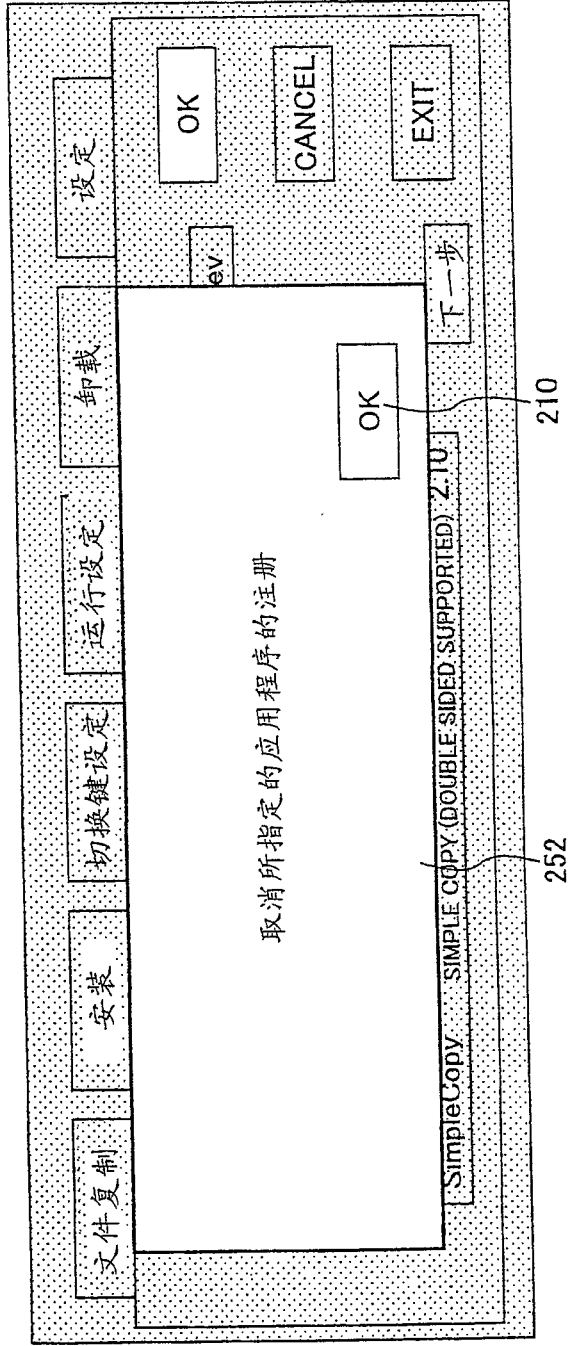


图 35B

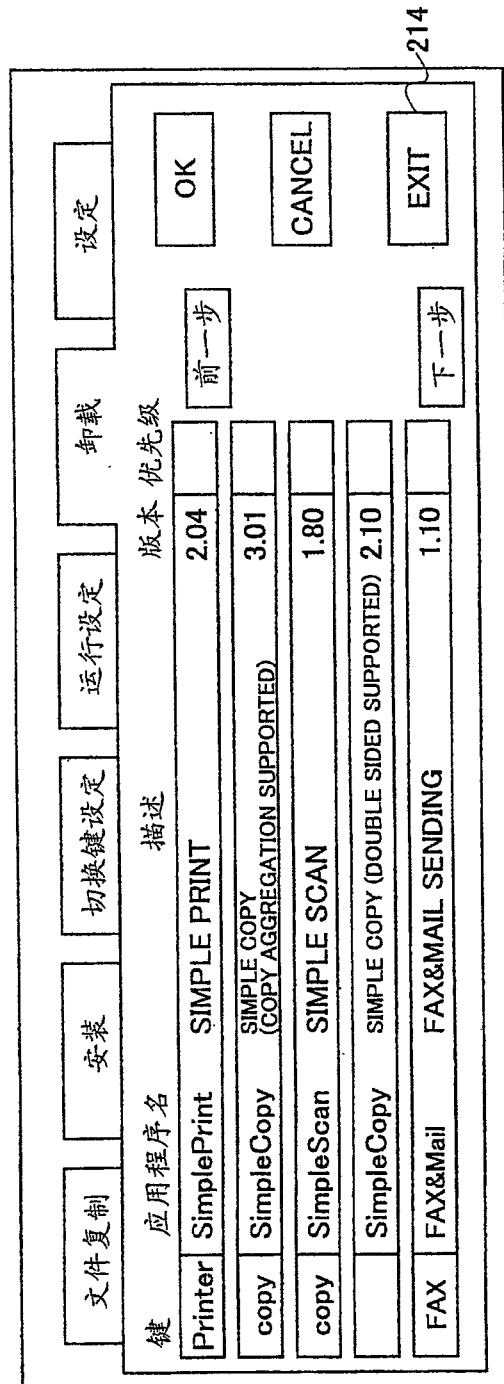


图 36

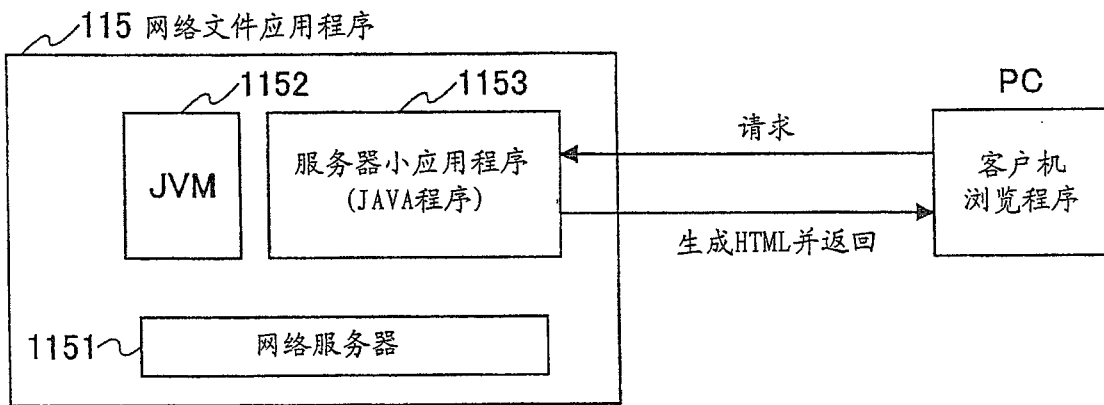


图 37

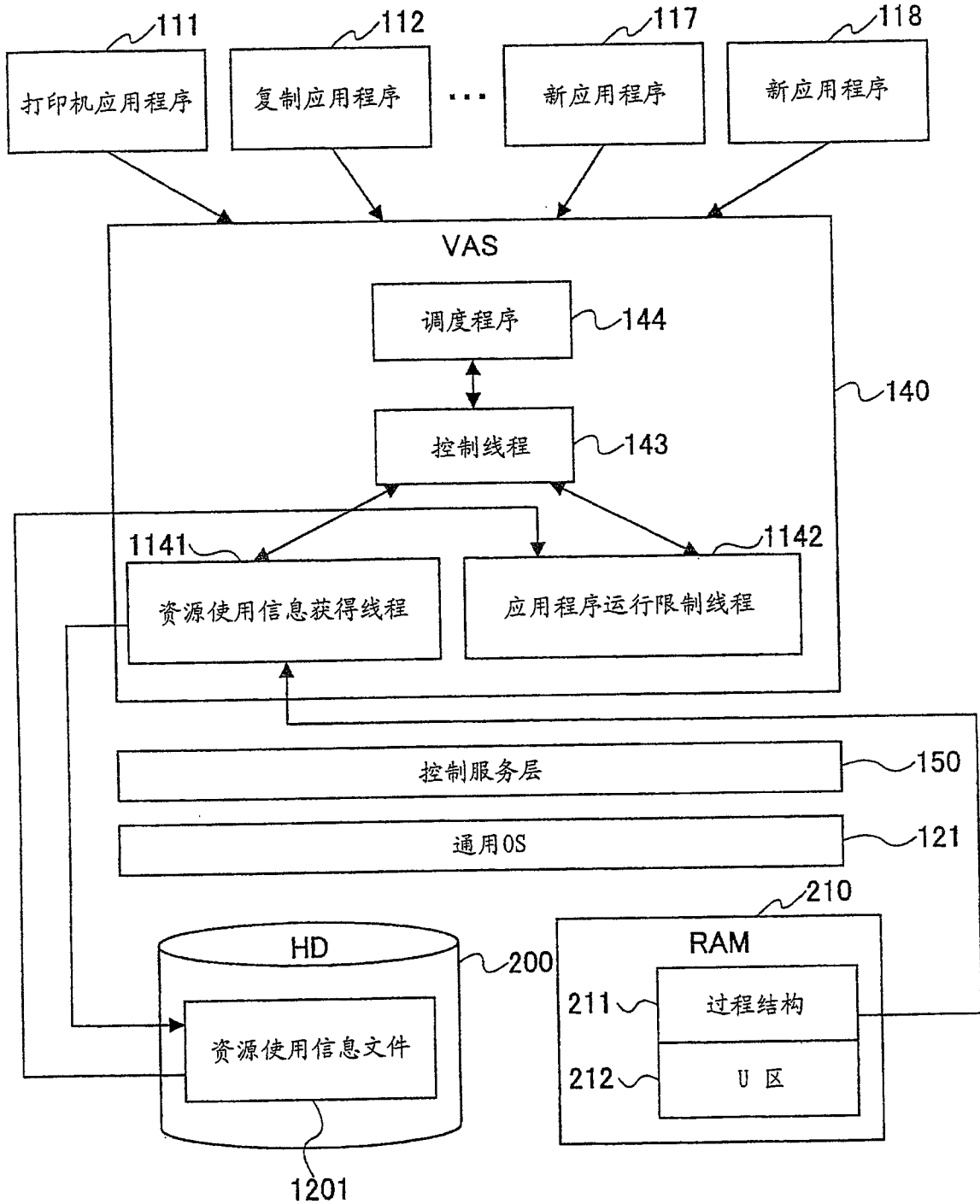


图 38

资源使用信息文件

201

应用文件ID	文本存储区大小	堆积区大小	堆栈区大小
101	3MB	1MB	10KB
102	2MB	500KB	5KB
103	4MB	200KB	3KB
.	.	.	.
.	.	.	.
.	.	.	.

图 39

过程结构

211

成员名	描述进程ID
p_pid	
p_txtsz	文本存储区大小
p_heapsz	堆积区大小
p_stacksz	堆栈区大小
.	.
.	.
.	.

图 40

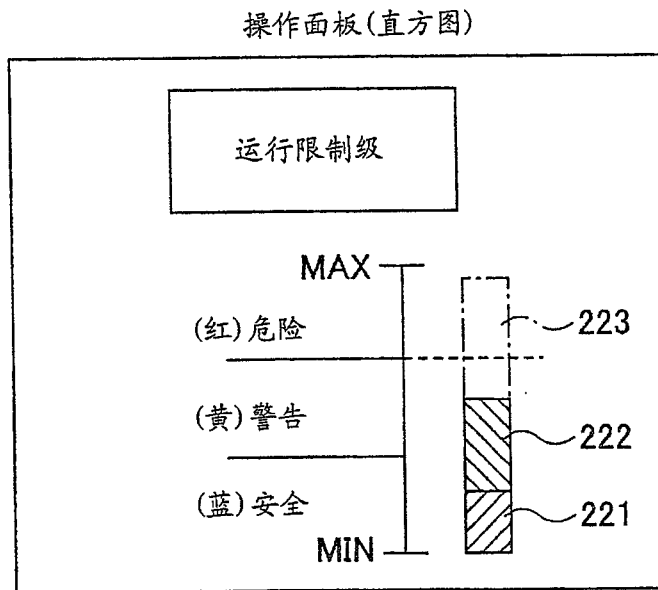
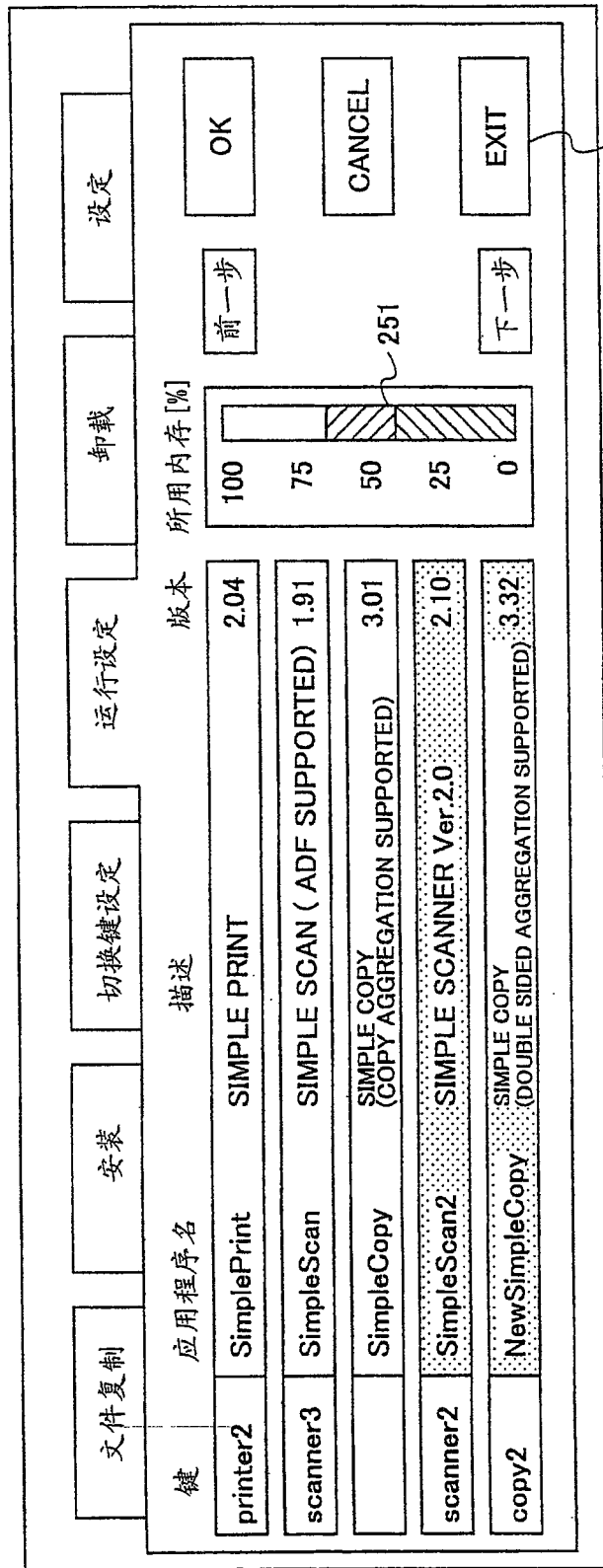


图 41



到初始设置屏

图 42

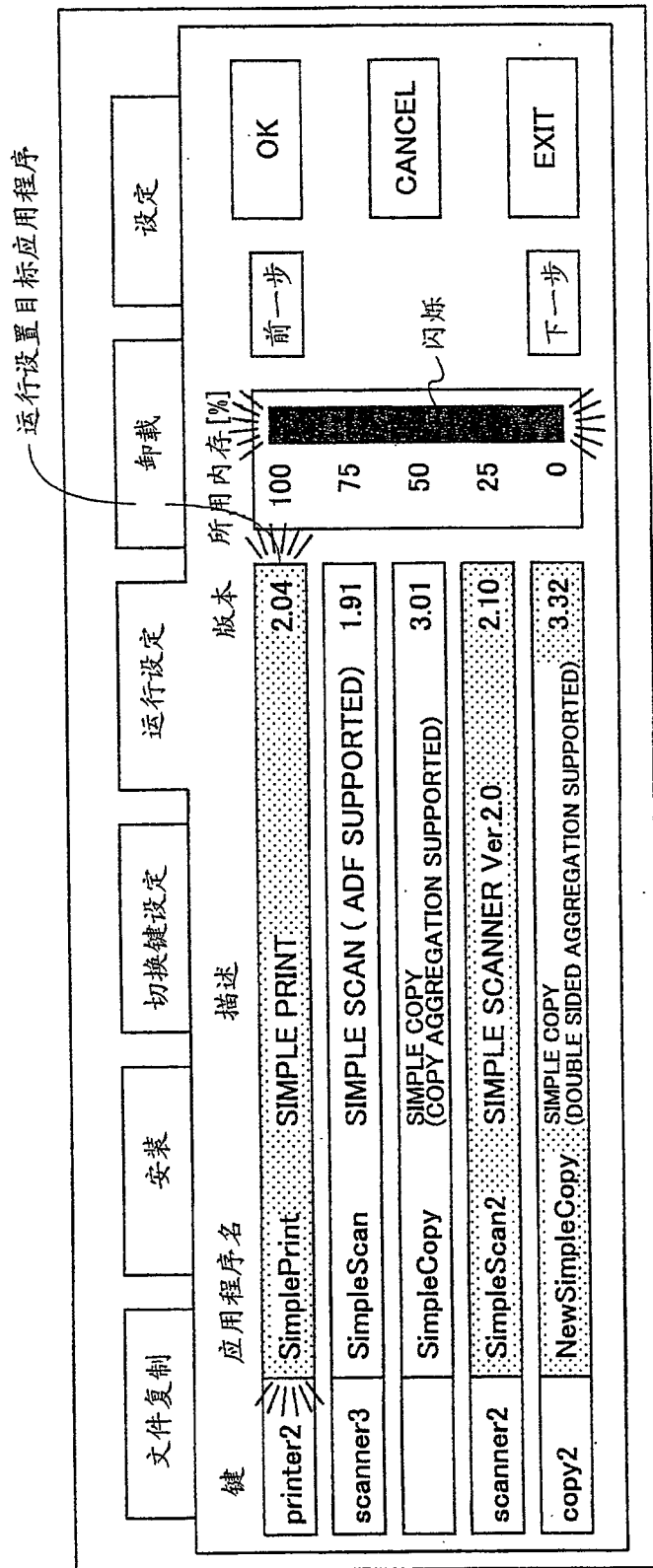


图 43

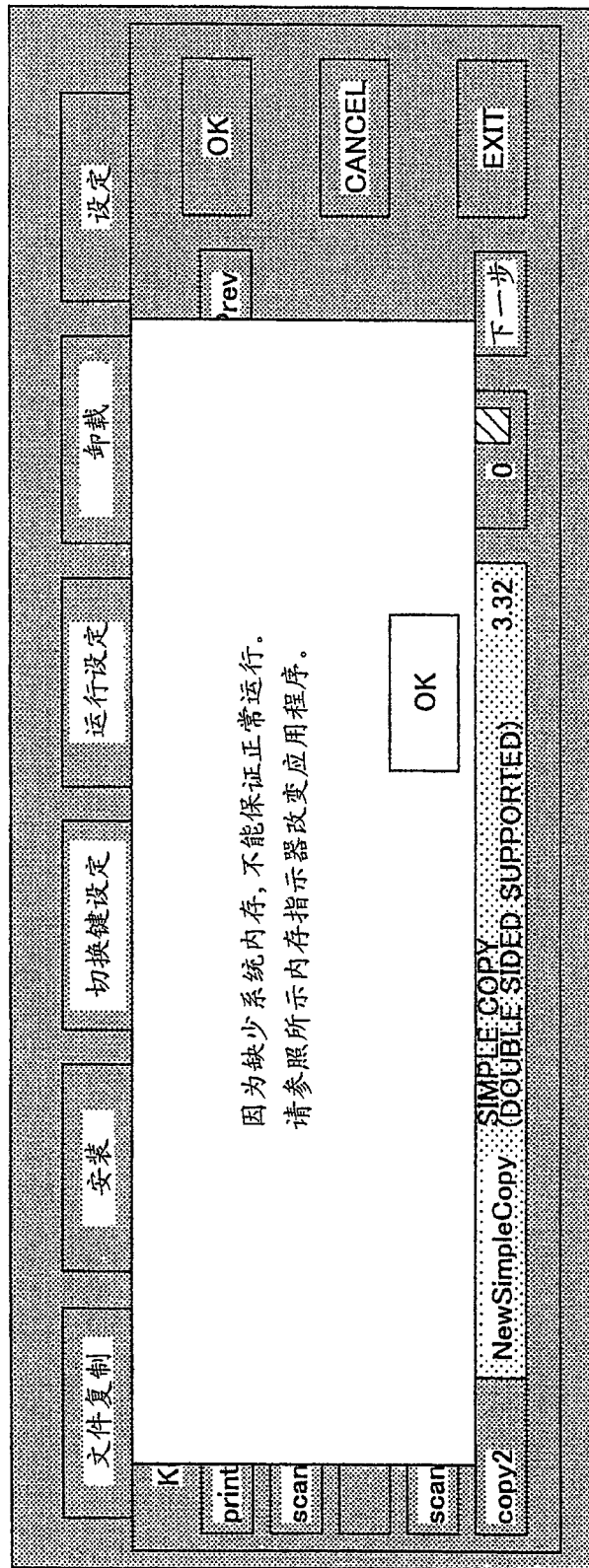


图 44

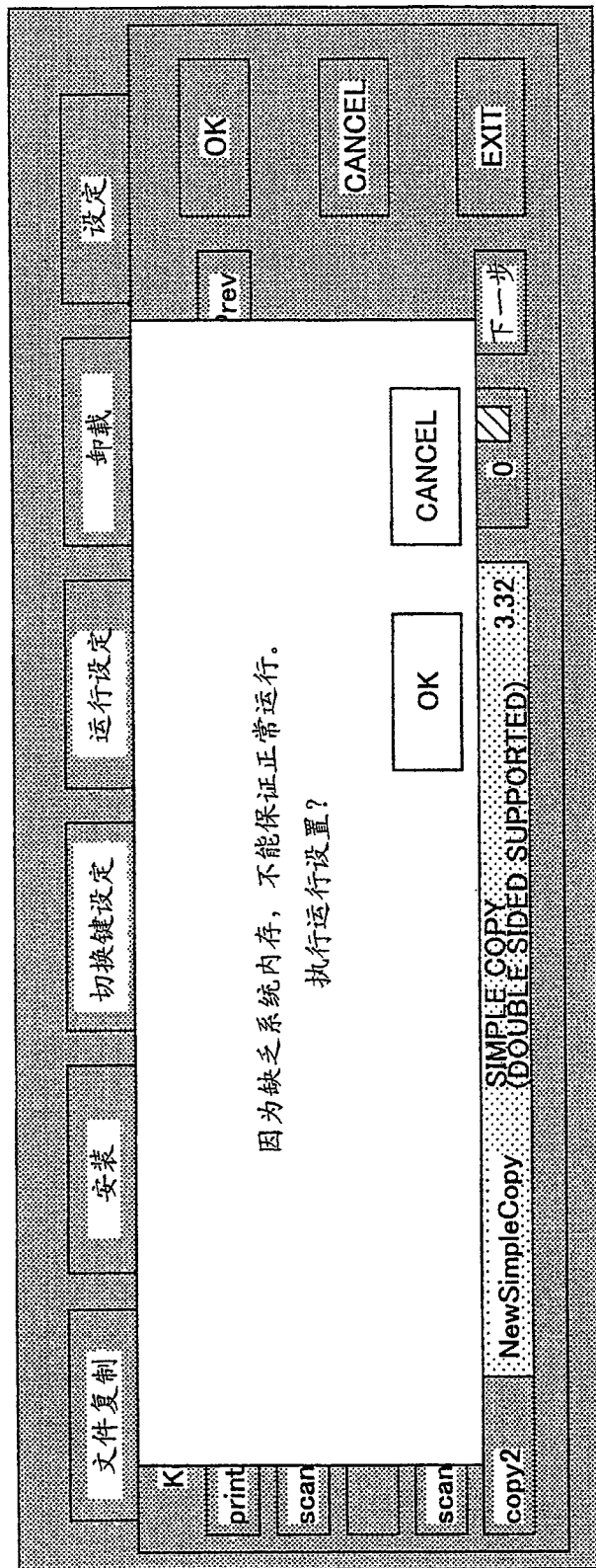


图 45

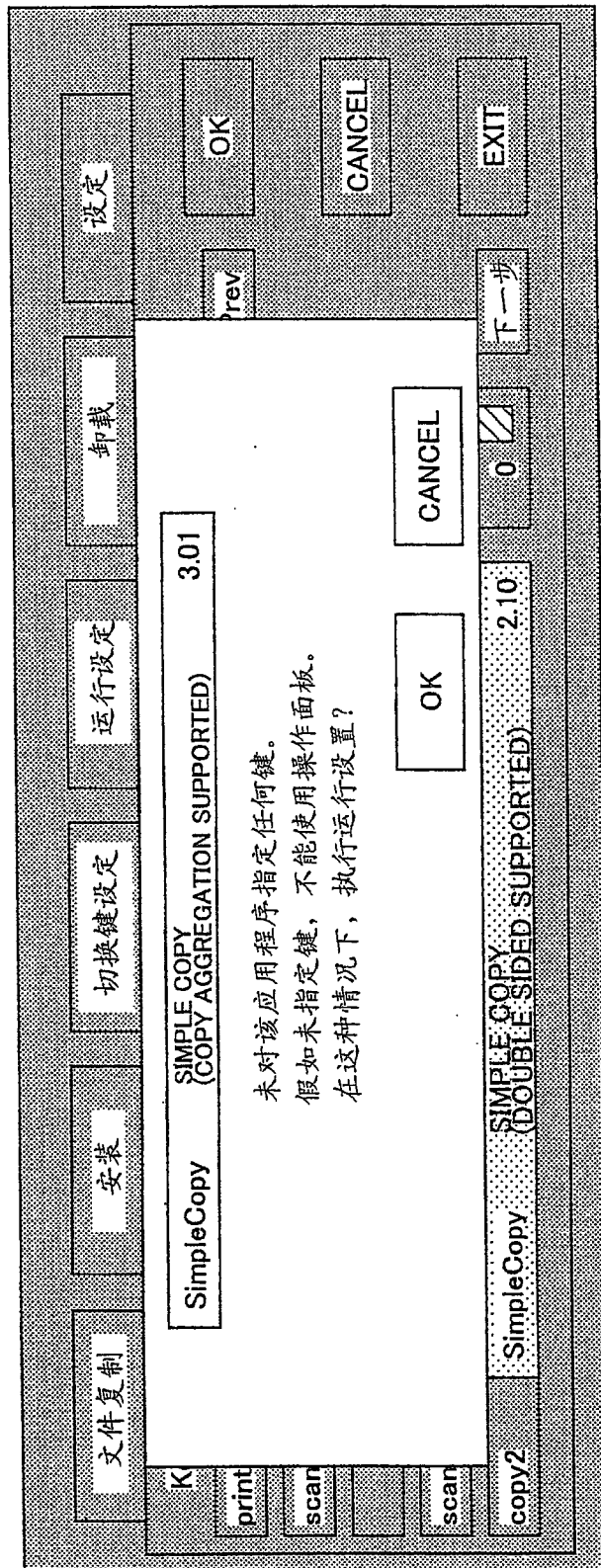


图 46

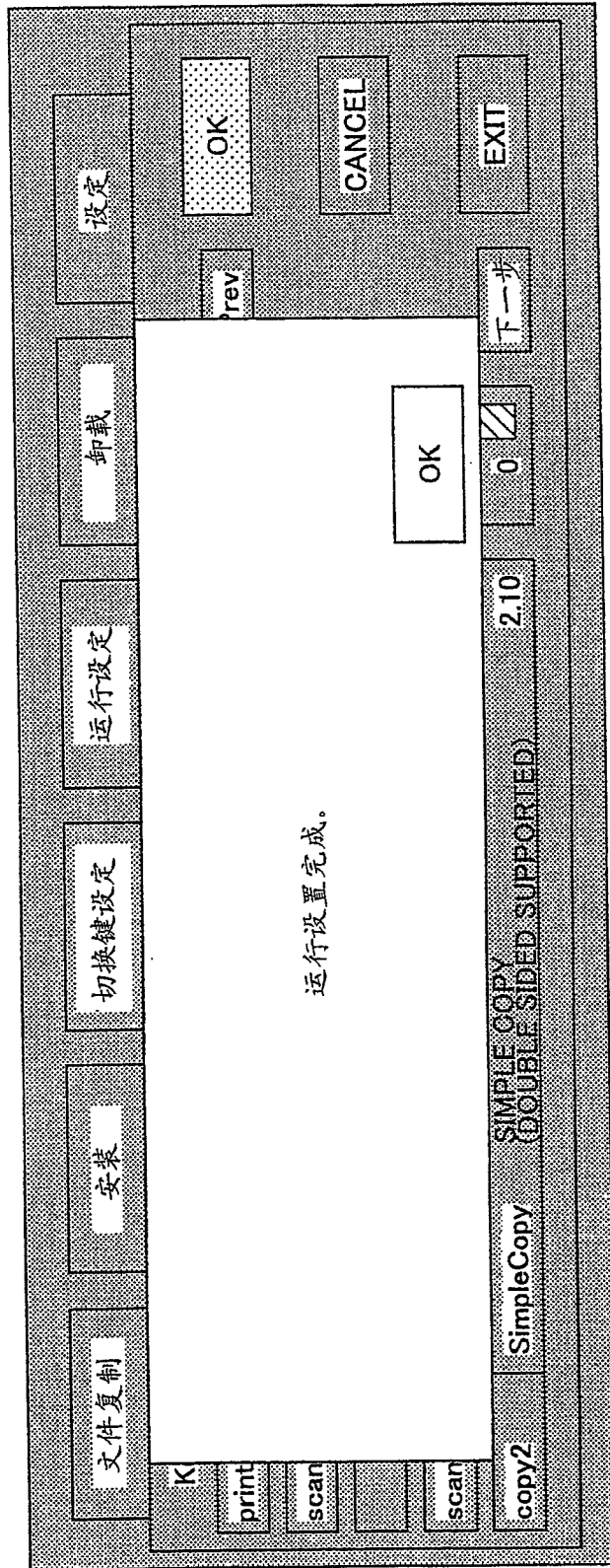


图 47

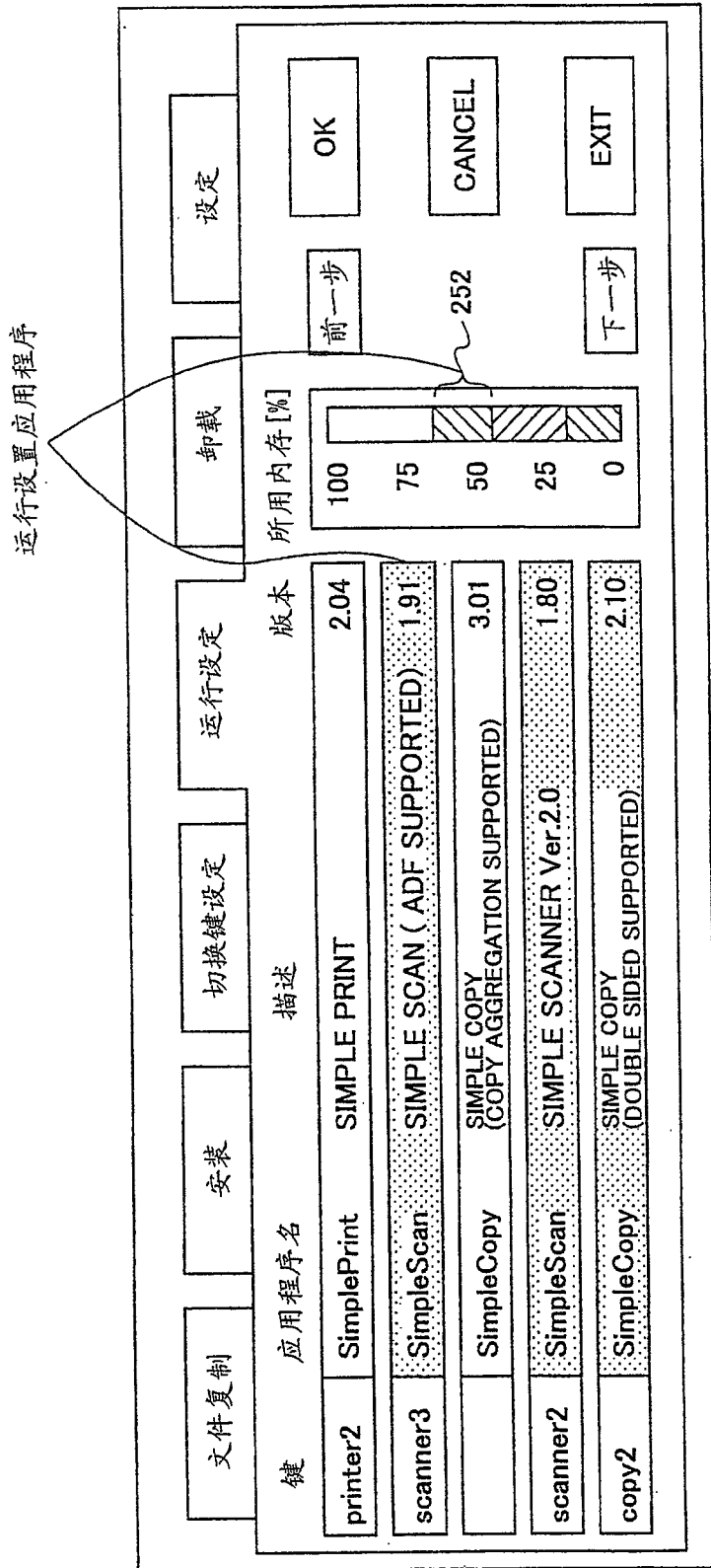


图 48

运行设置应用程序

文件复制 安装 切换键设定 运行设定 卸载 设定

键	应用程序名	描述	版本
printer2	SimplePrint	SIMPLE PRINT	2.04
scanner3	SimpleScan	SIMPLE SCAN (ADF SUPPORTED)	1.91
	SimpleCopy	SIMPLE COPY (COPY AGGREGATION SUPPORTED)	3.01
scanner2	SimpleScan	SIMPLE SCANNER Ver.2.0	1.80
copy2	SimpleCopy	SIMPLE COPY (DOUBLE SIDED SUPPORTED)	2.10

所用内存 [%]

100 75 50 25 0

前一步 下一步

OK CANCEL EXIT

图 49

运行设置释放目标应用程序

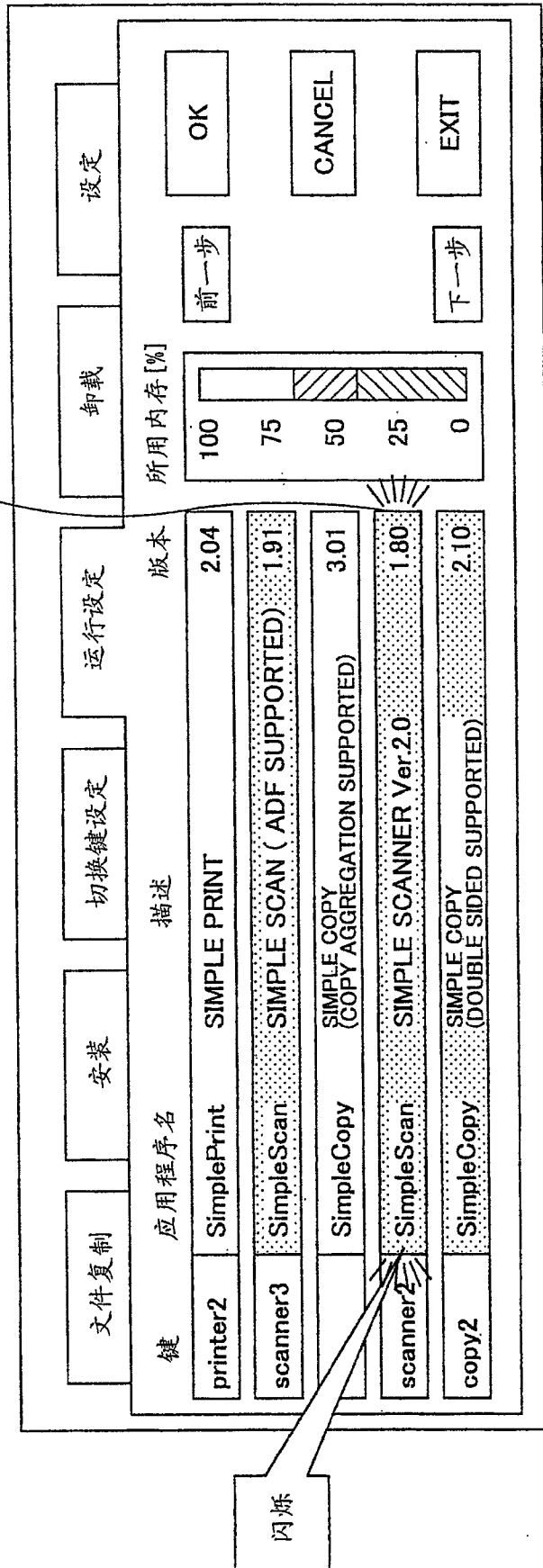


图 50

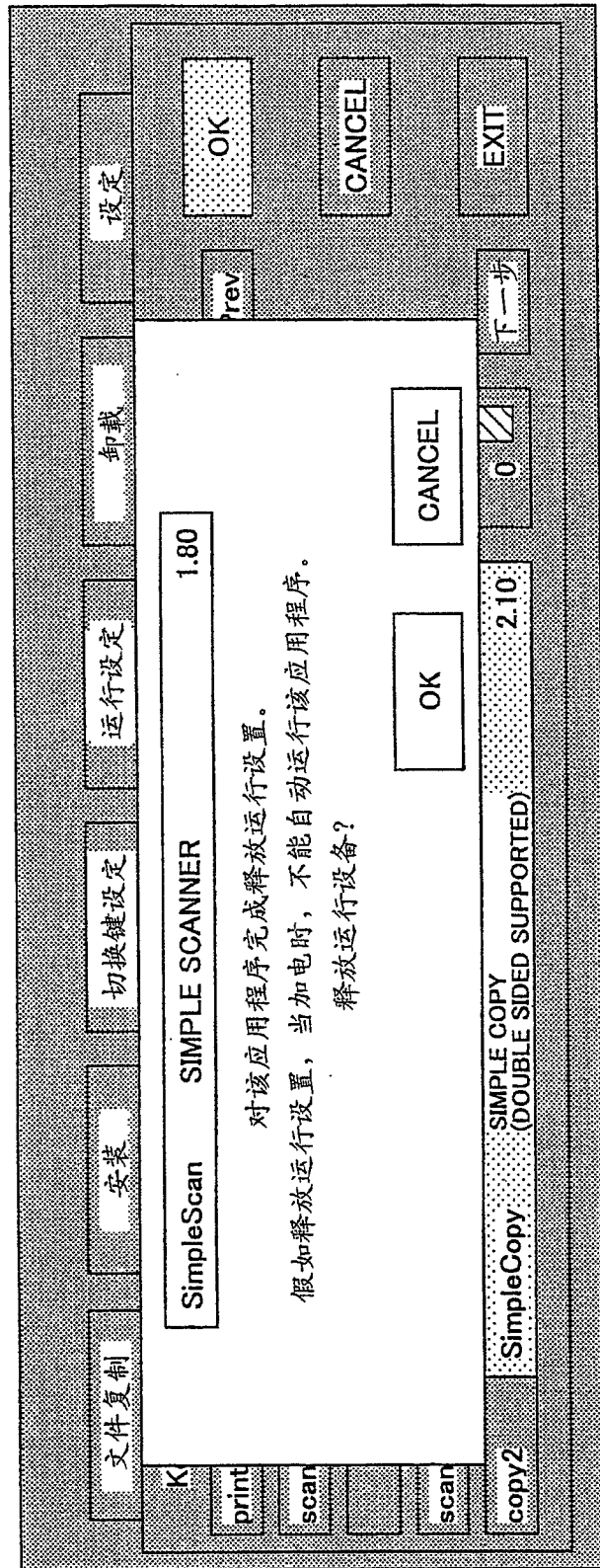


图 51

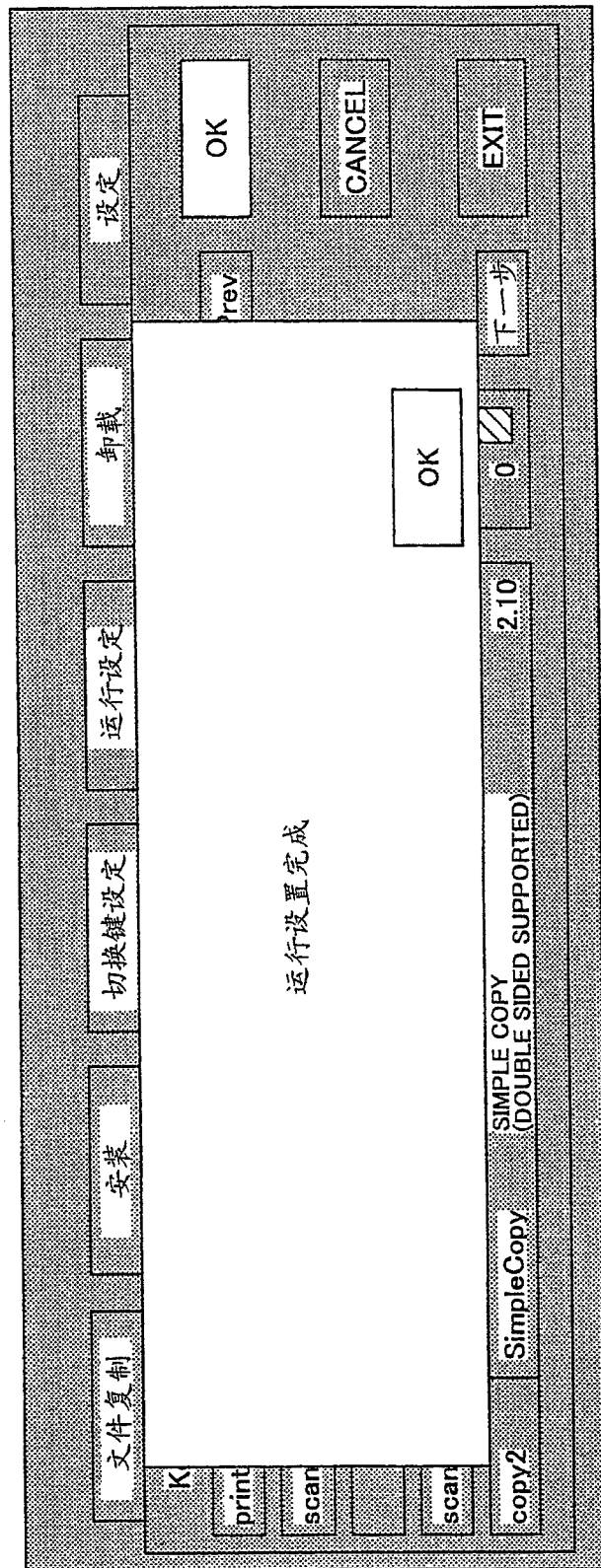


图 52

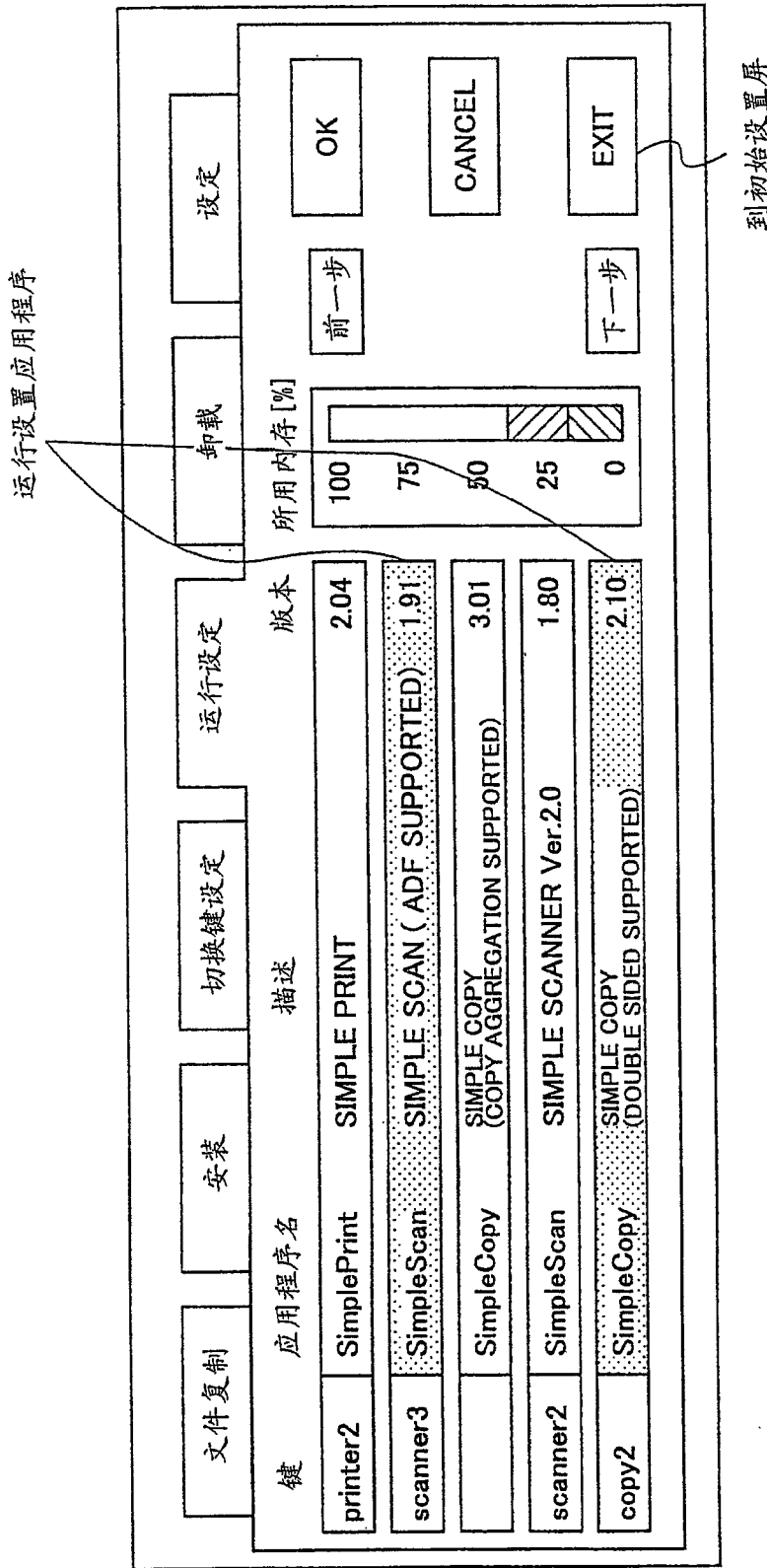


图 53

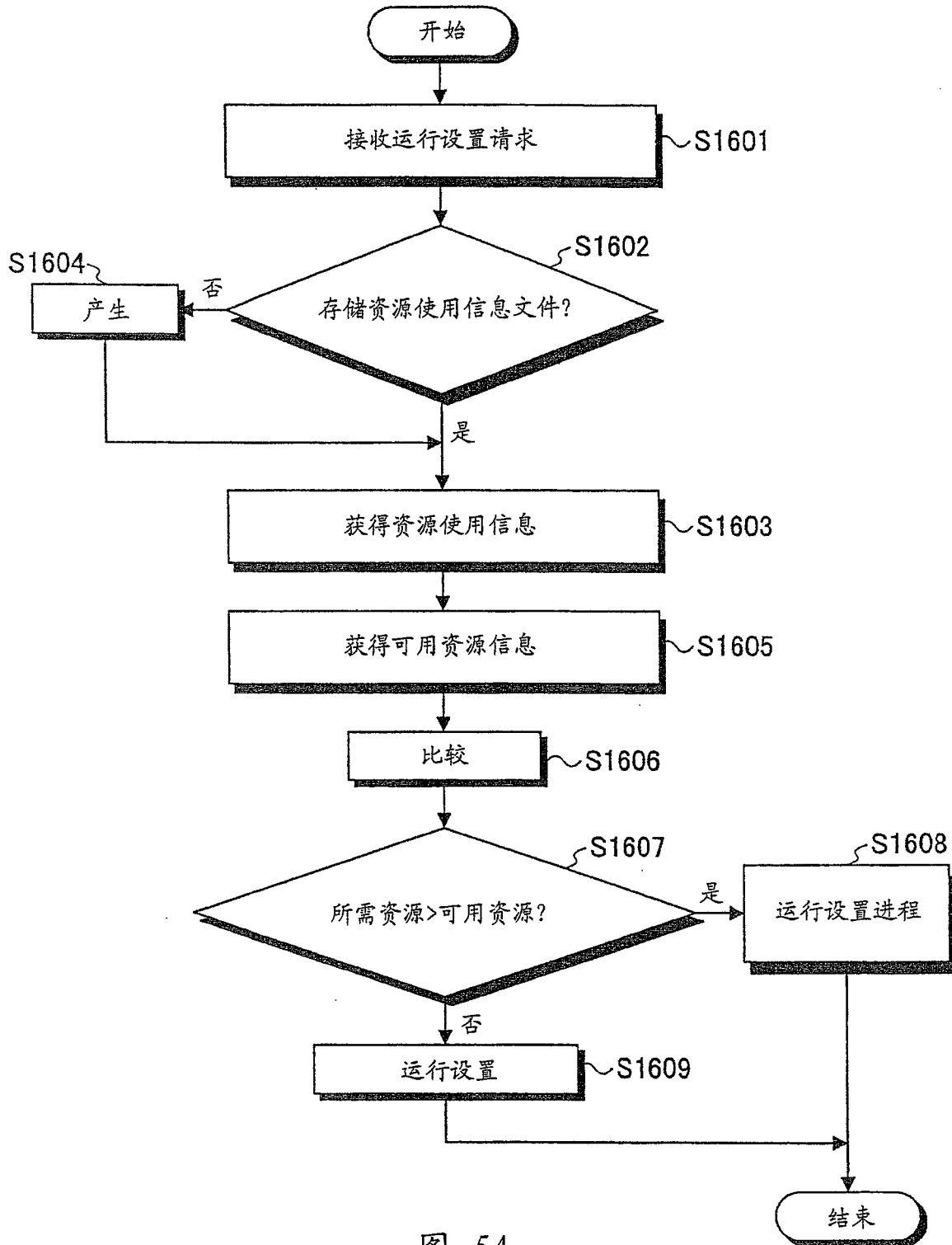
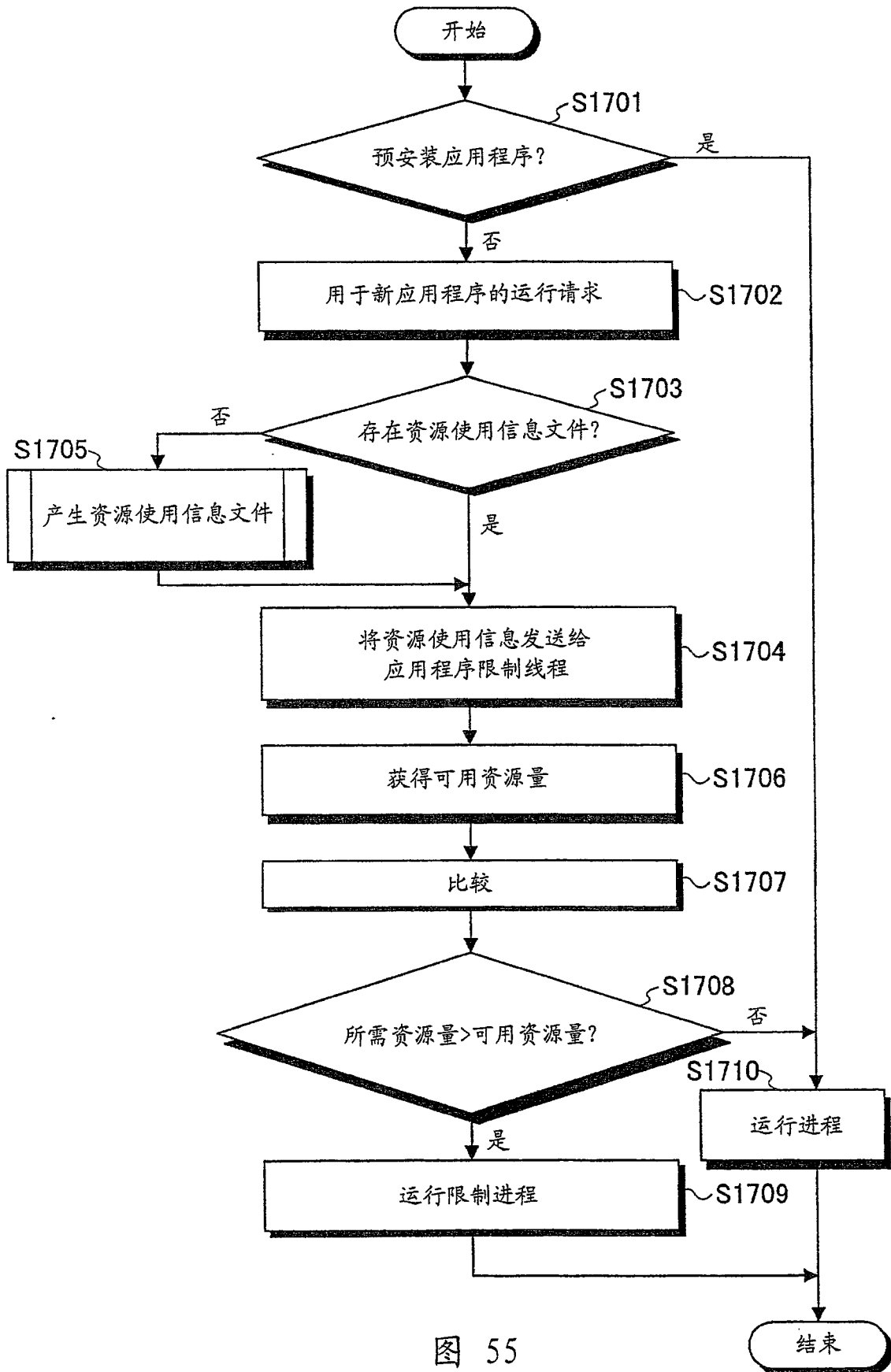


图 54



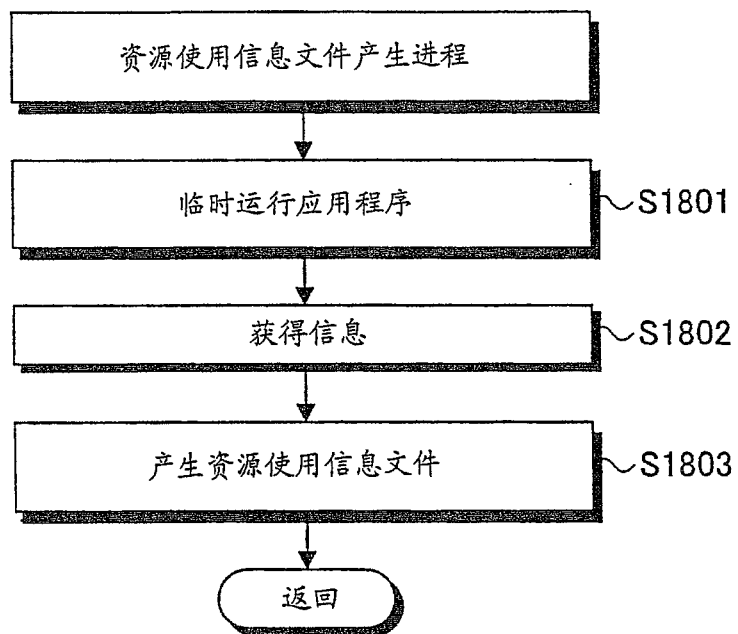


图 56

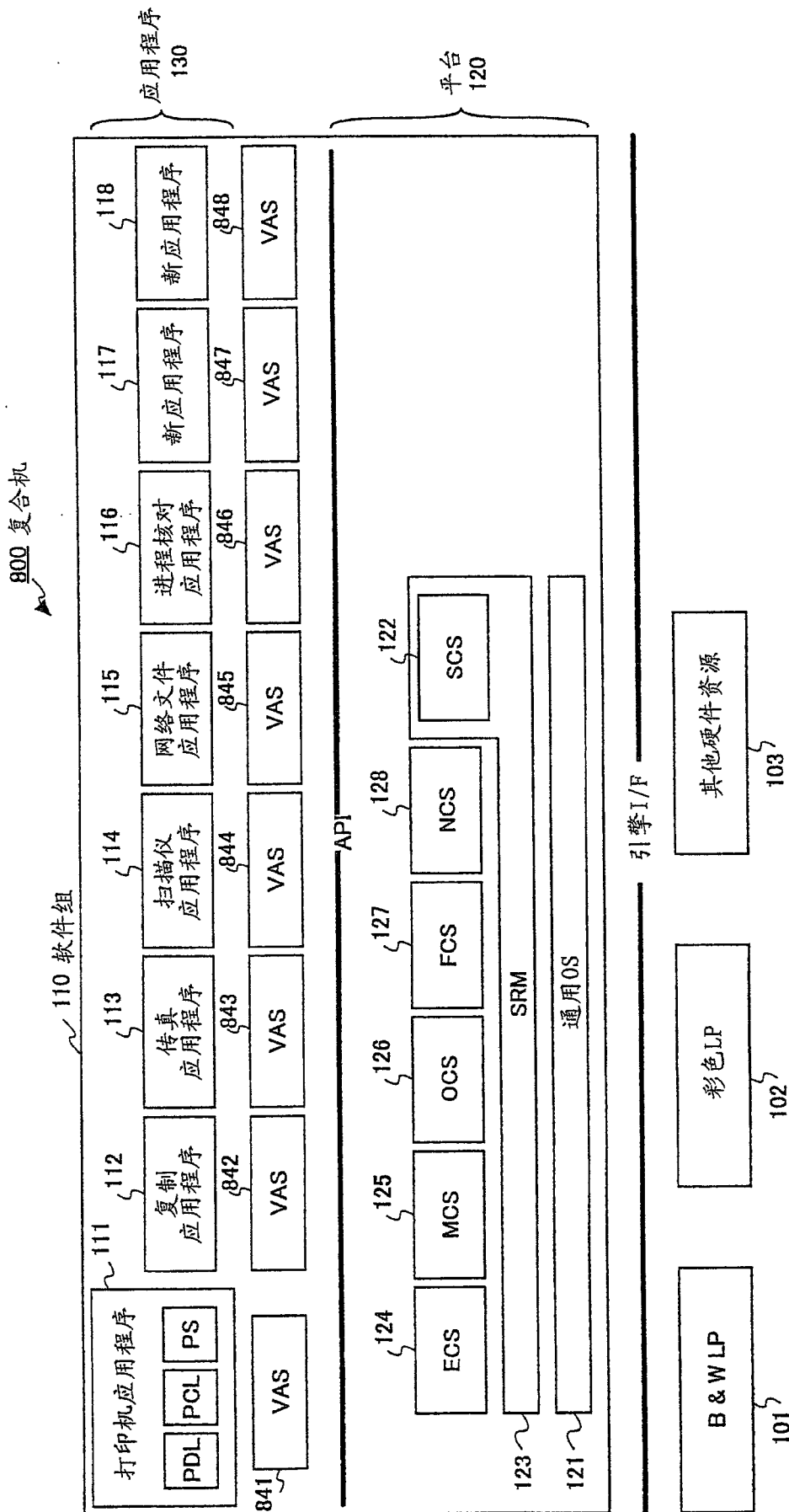


图 57

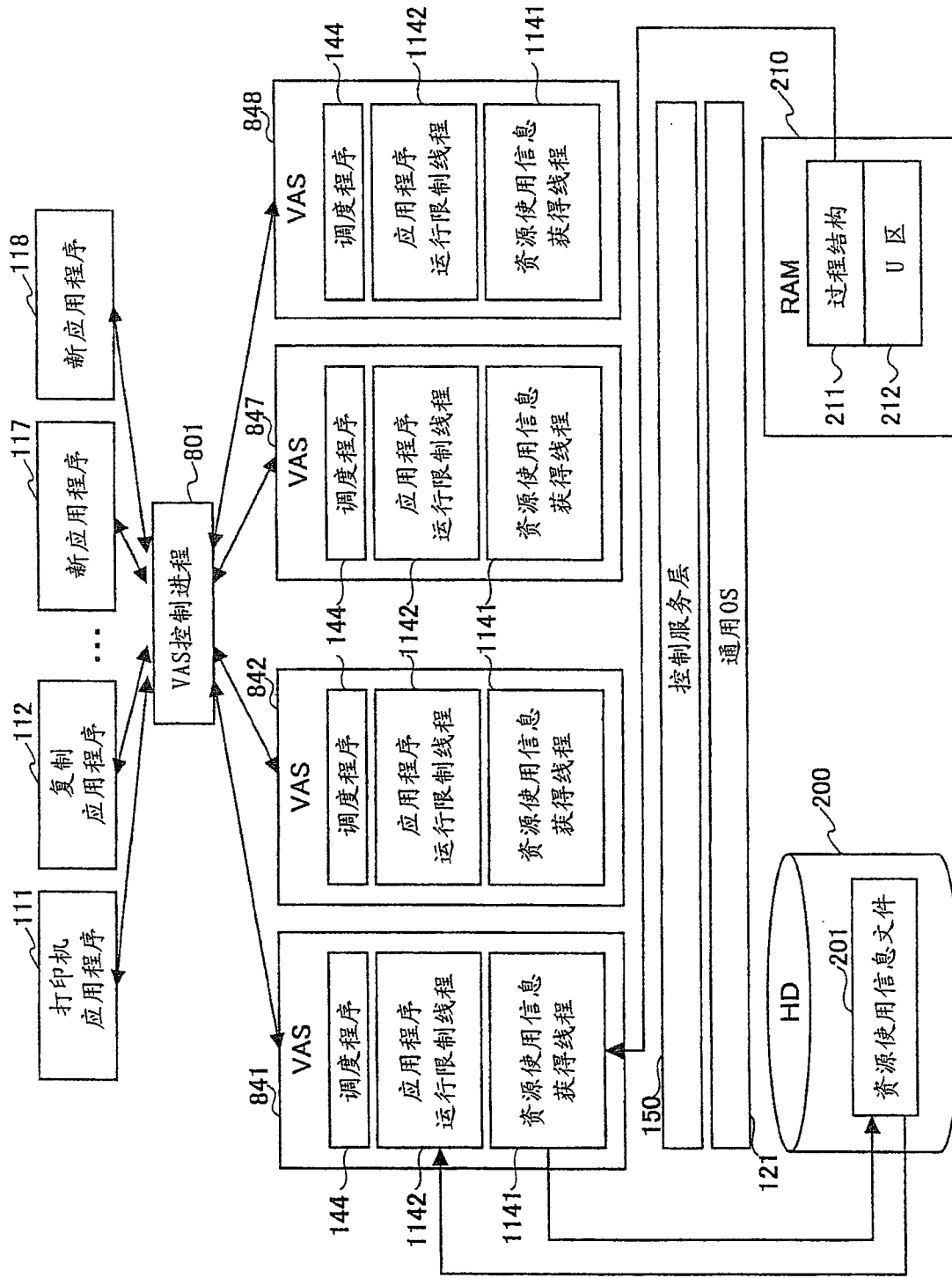


图 58

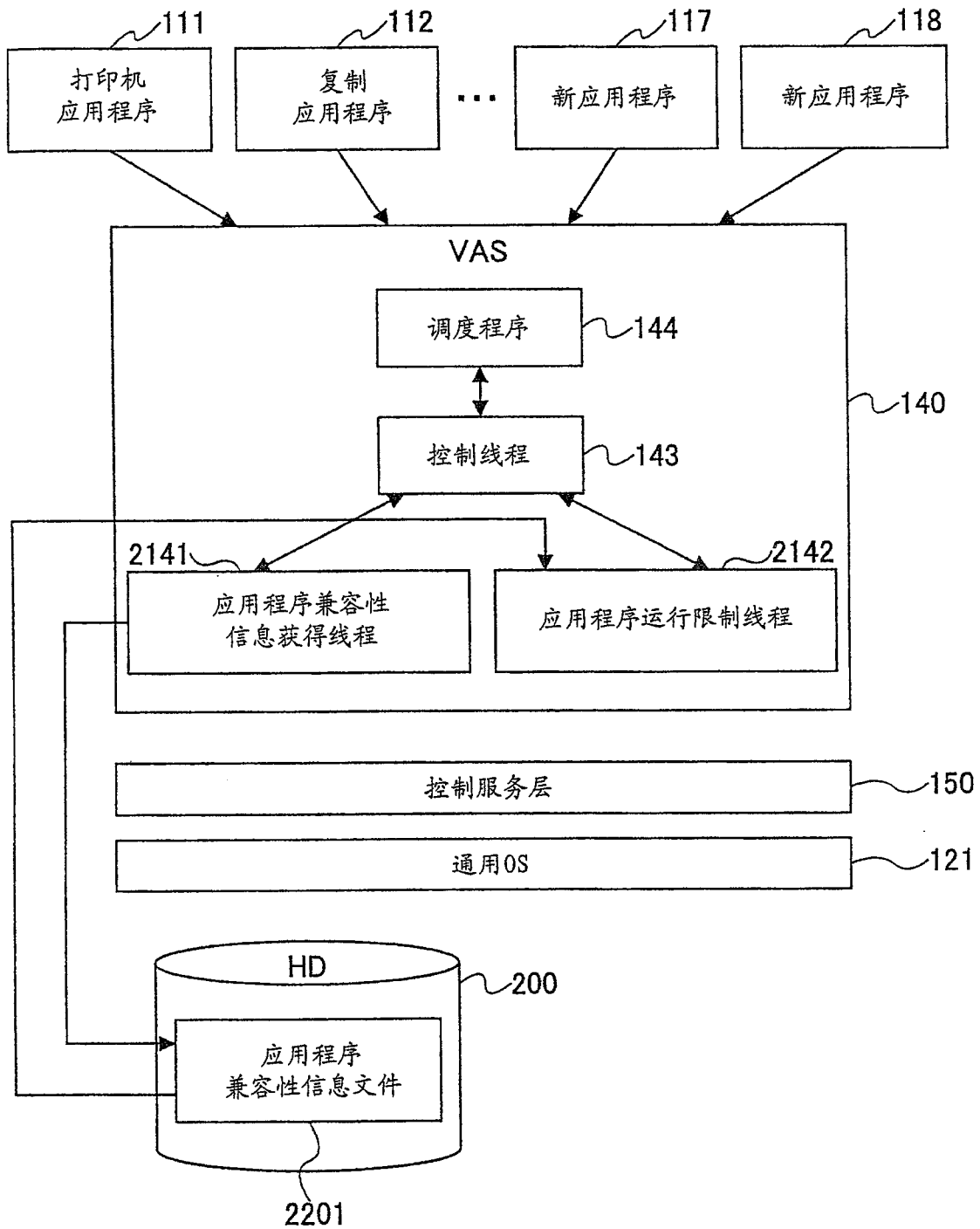


图 59

应用程序ID	不兼容应用程序ID
101	105,103
102	104
103	101
104	102
105	101

图 60

应用程序ID	属性	操作目标	应用程序名	版本
101	PRINTER TYPE	PRINTER	SP PRINT	1.1
102	AUTHENTICATION TYPE	OPERATION PANEL	VERIFICATION AP	2.0
103	BILLING TYPE	BILLING COUNTER	COUNT PLUS	1.01
104	AUTHENTICATION TYPE	OPERATION PANEL	AUTHENTICATION AP	1.5
105	PRINTER TYPE	PRINTER	SP PRINT	1.2
⋮	⋮	⋮	⋮	⋮

图 61

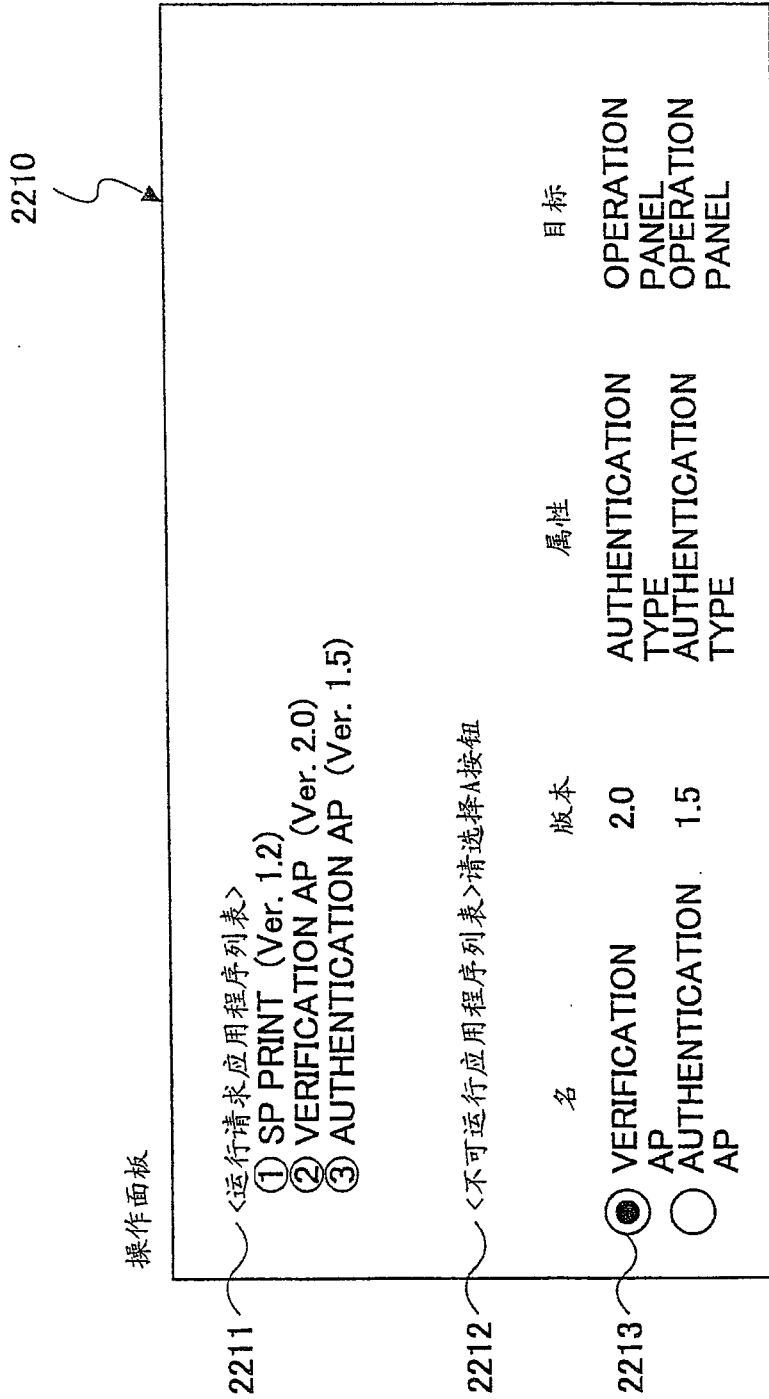


图 62

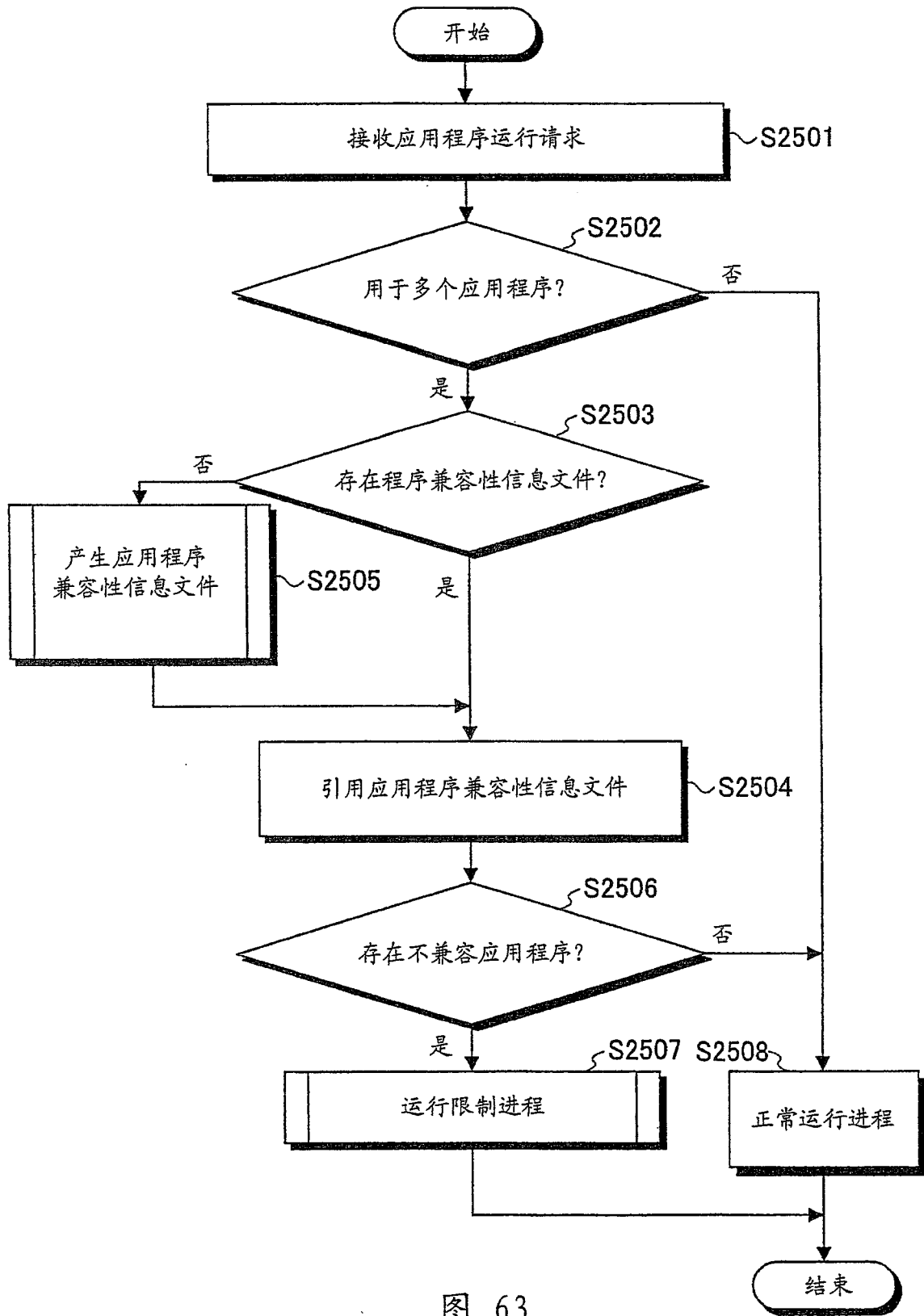


图 63

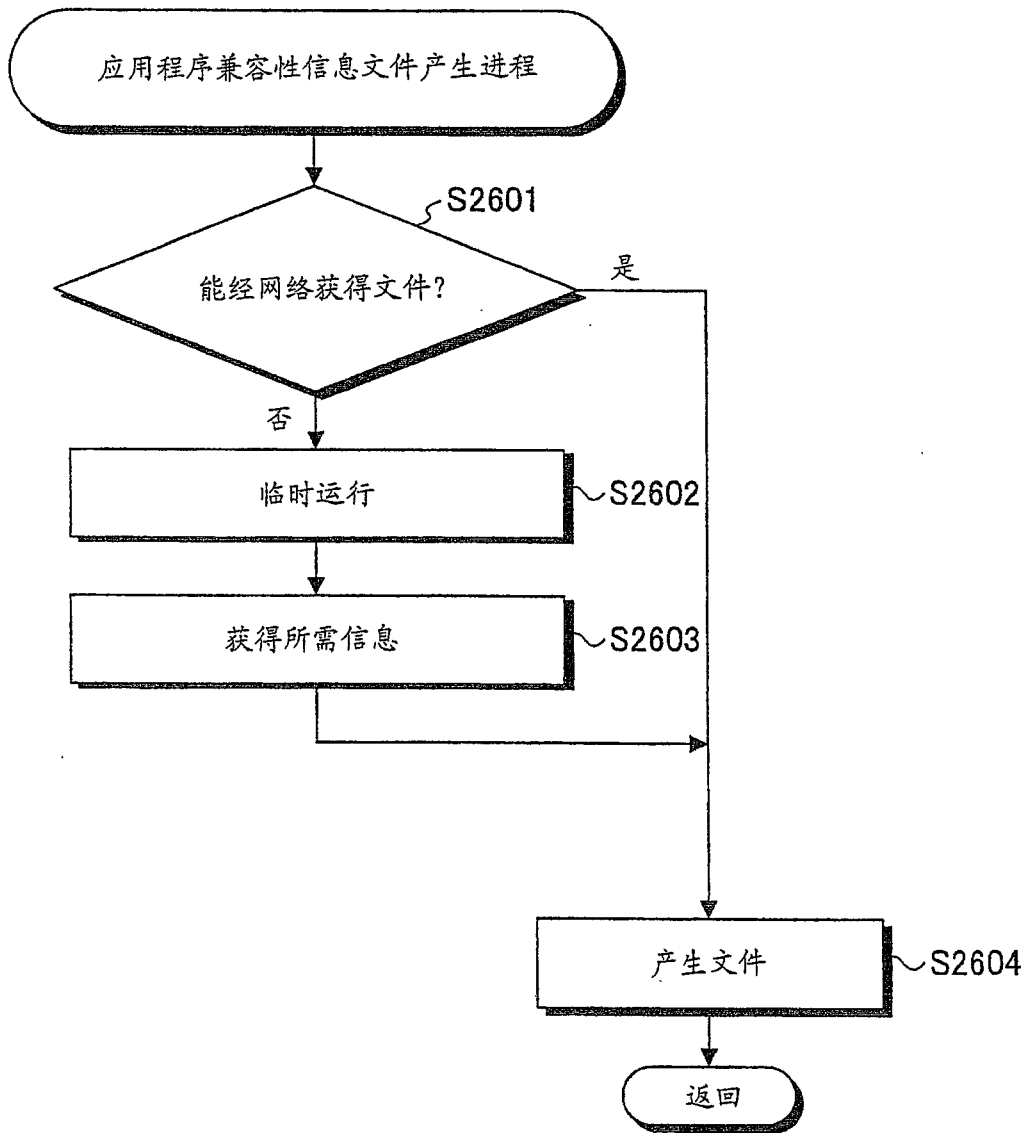


图 64

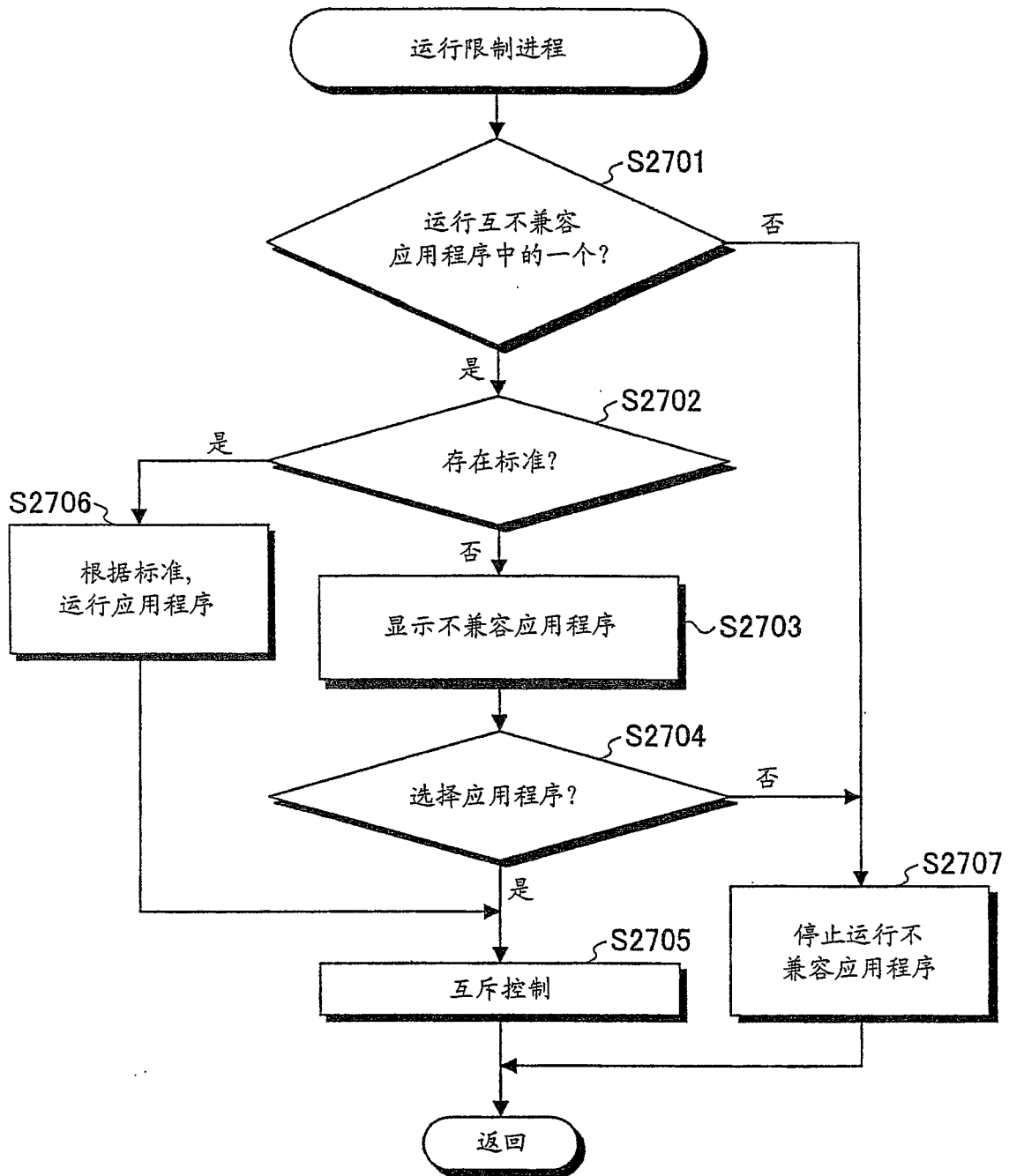


图 65

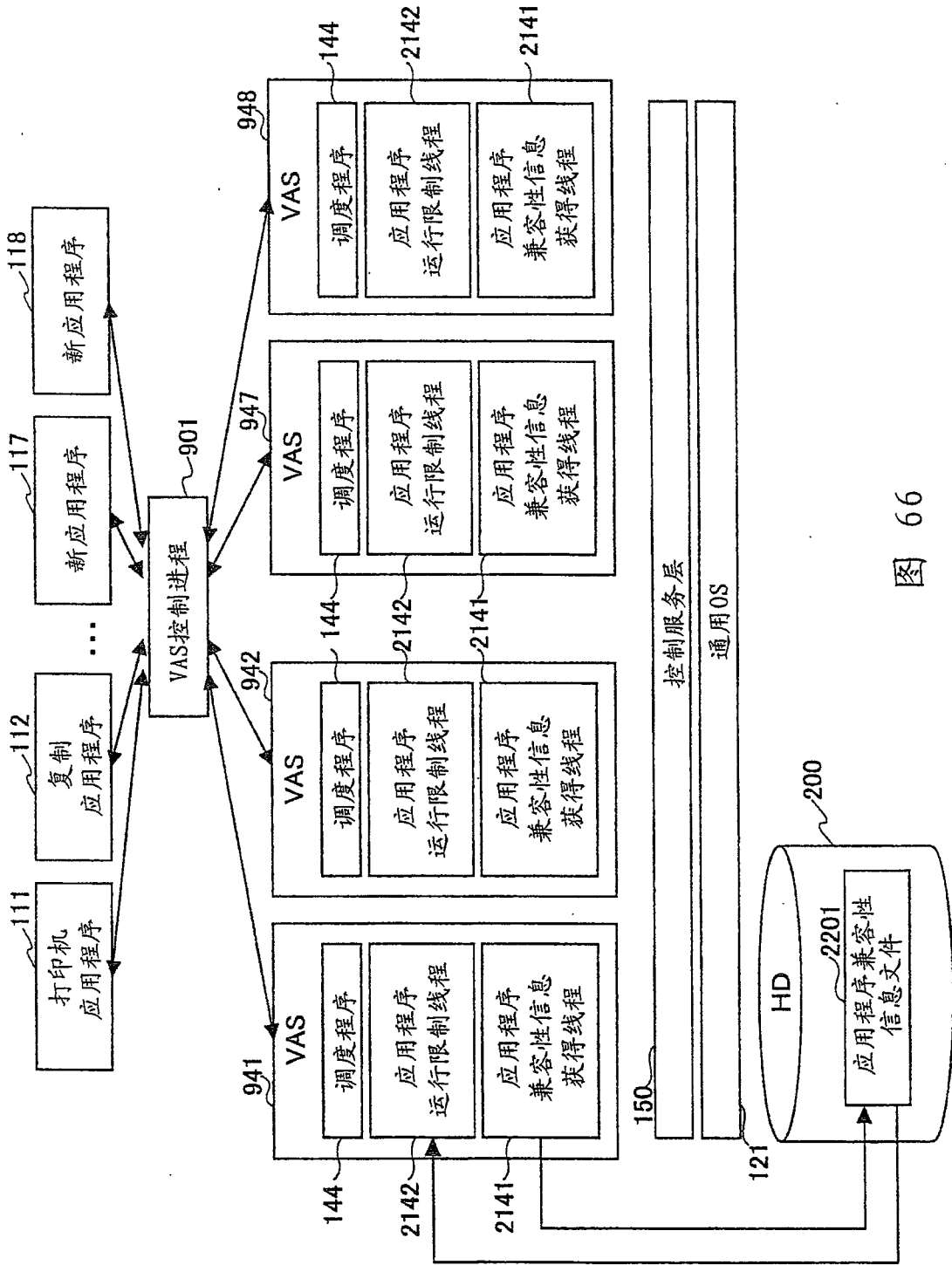


图 66

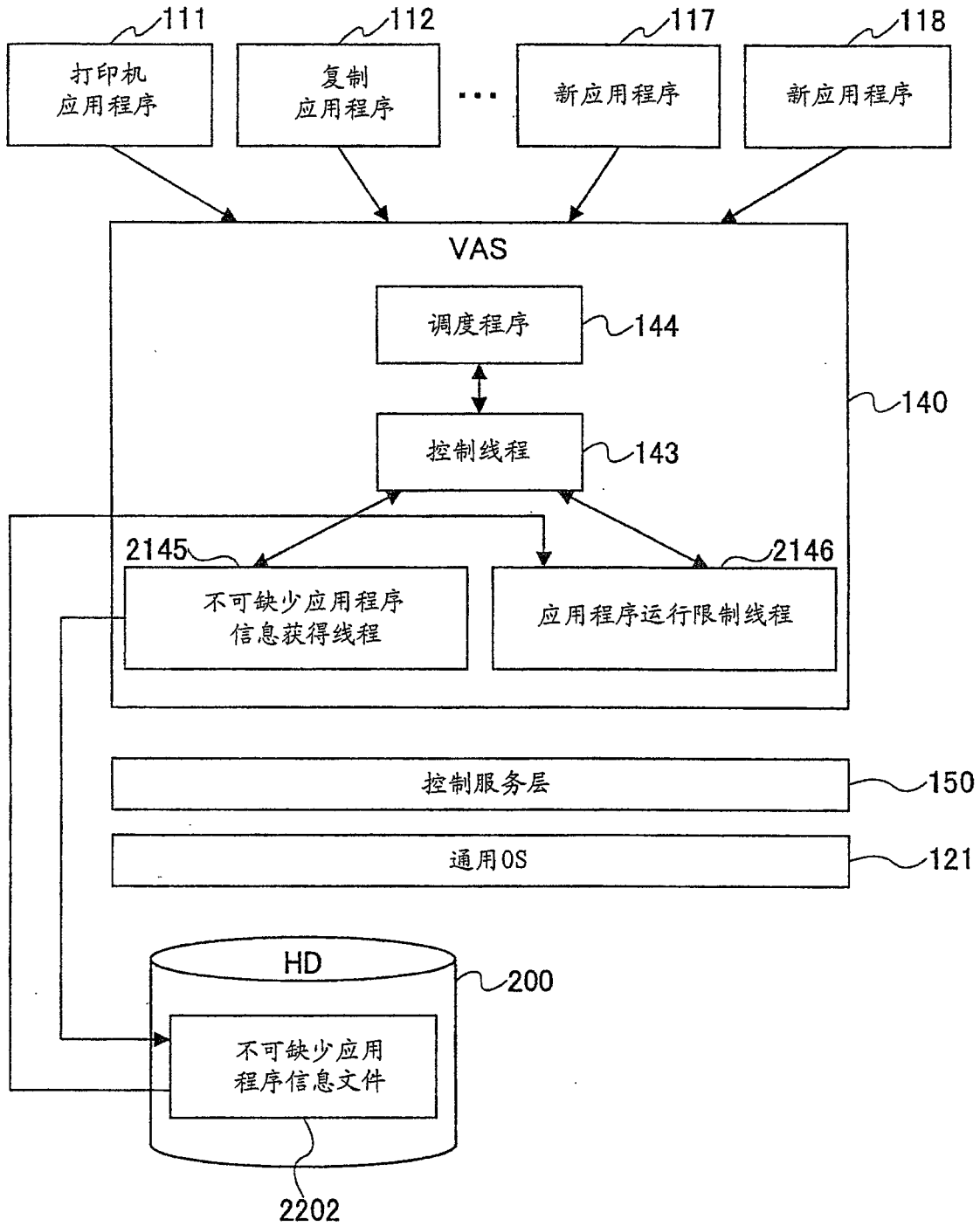


图 67

不可缺少应用程序信息文件

应用程序ID	存在或不存在不可缺少应用程序	应用程序类型	相应的应用程序ID
101	x	COPY	—
102	x	PRINTER	—
103	○	BROWSER	105
104	x	FAX	—
105	○	PLUG-IN	103
⋮	⋮	⋮	⋮

图 68

操作面板

[运行请求的应用程序]	[存在或不存在不可缺少应用程序]	[不可缺少应用程序]	[存在或不存在运行]
1.BROWSER	○	PLUG-IN	x
2.PRINTER	x	—	○

<LAUNCH ERROR MESSAGE>
PLEASE LAUNCH BOTH OF BROWSER AND PLUG-IN

图 69

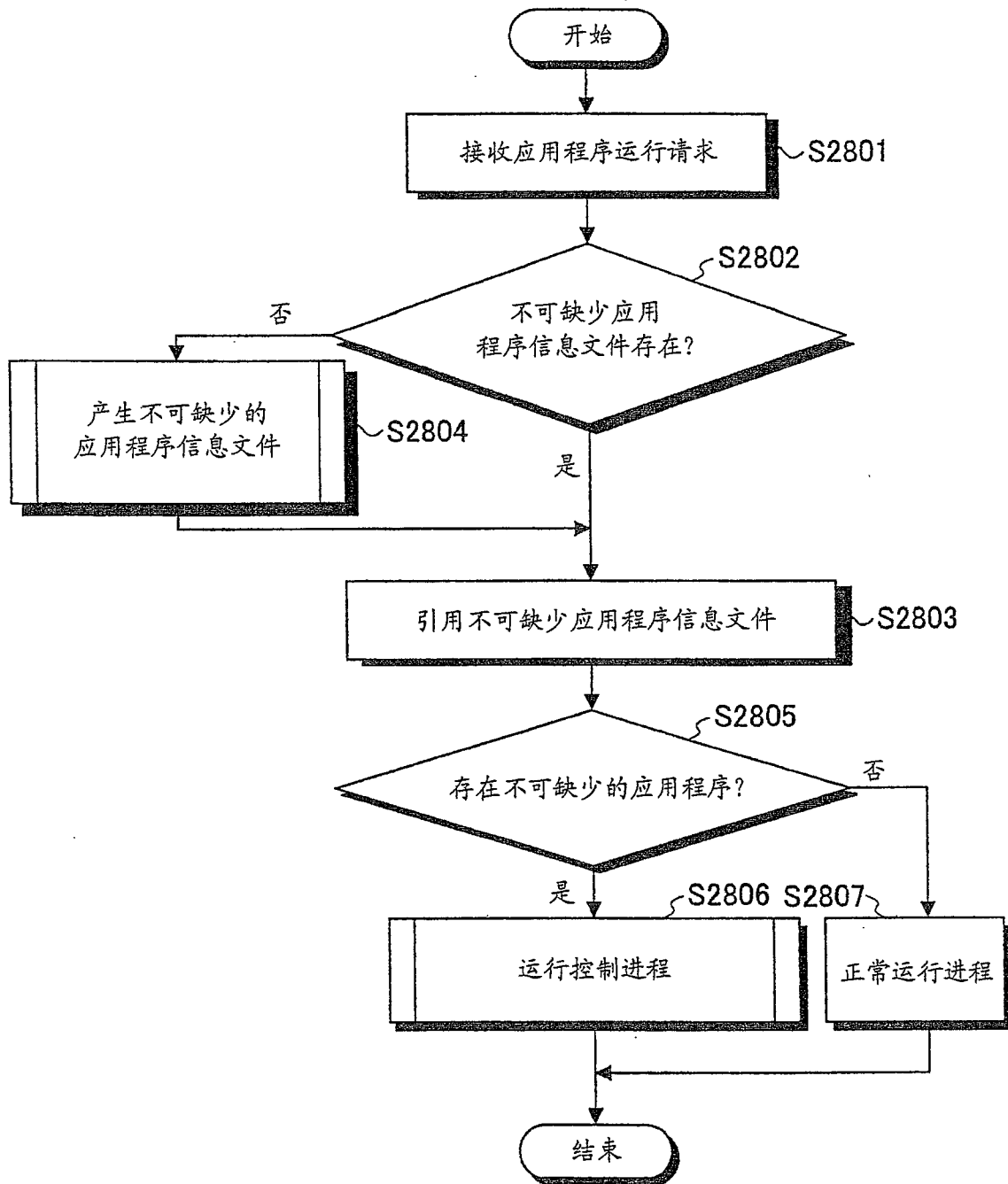


图 70

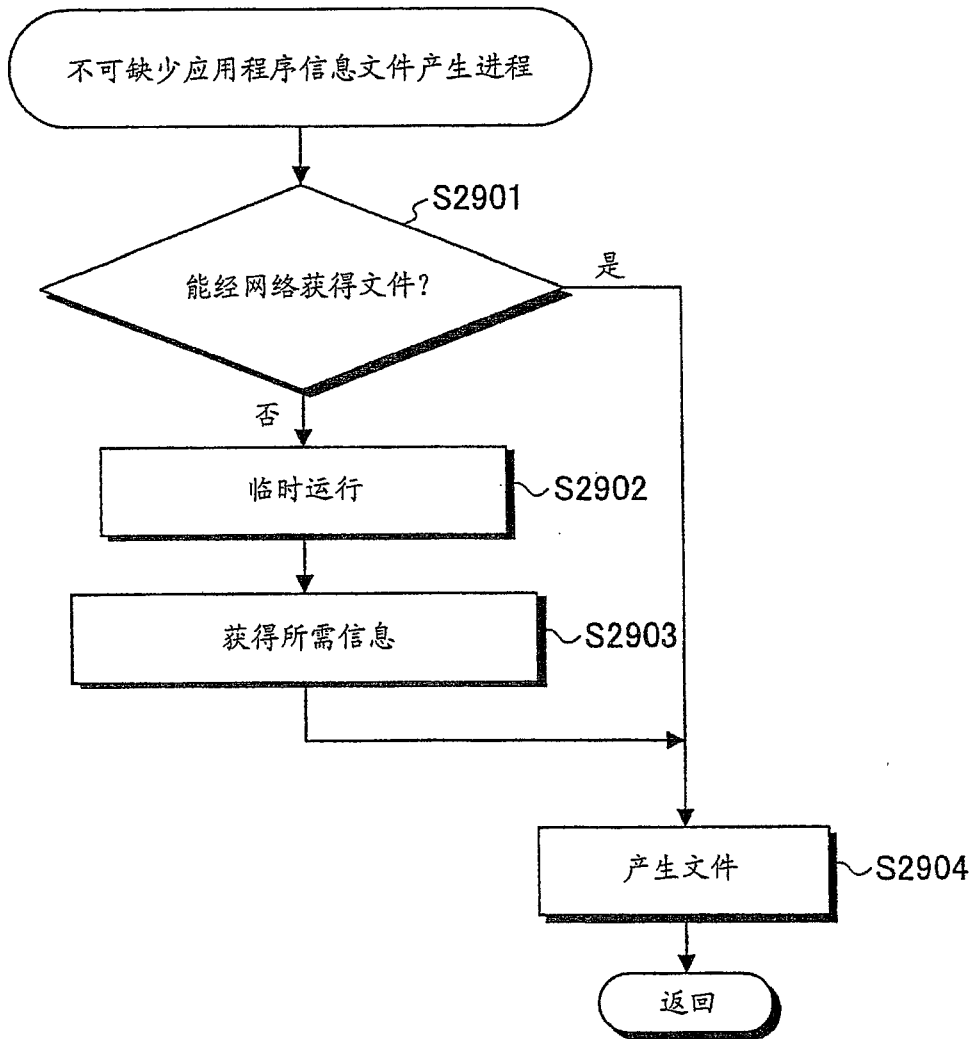


图 71

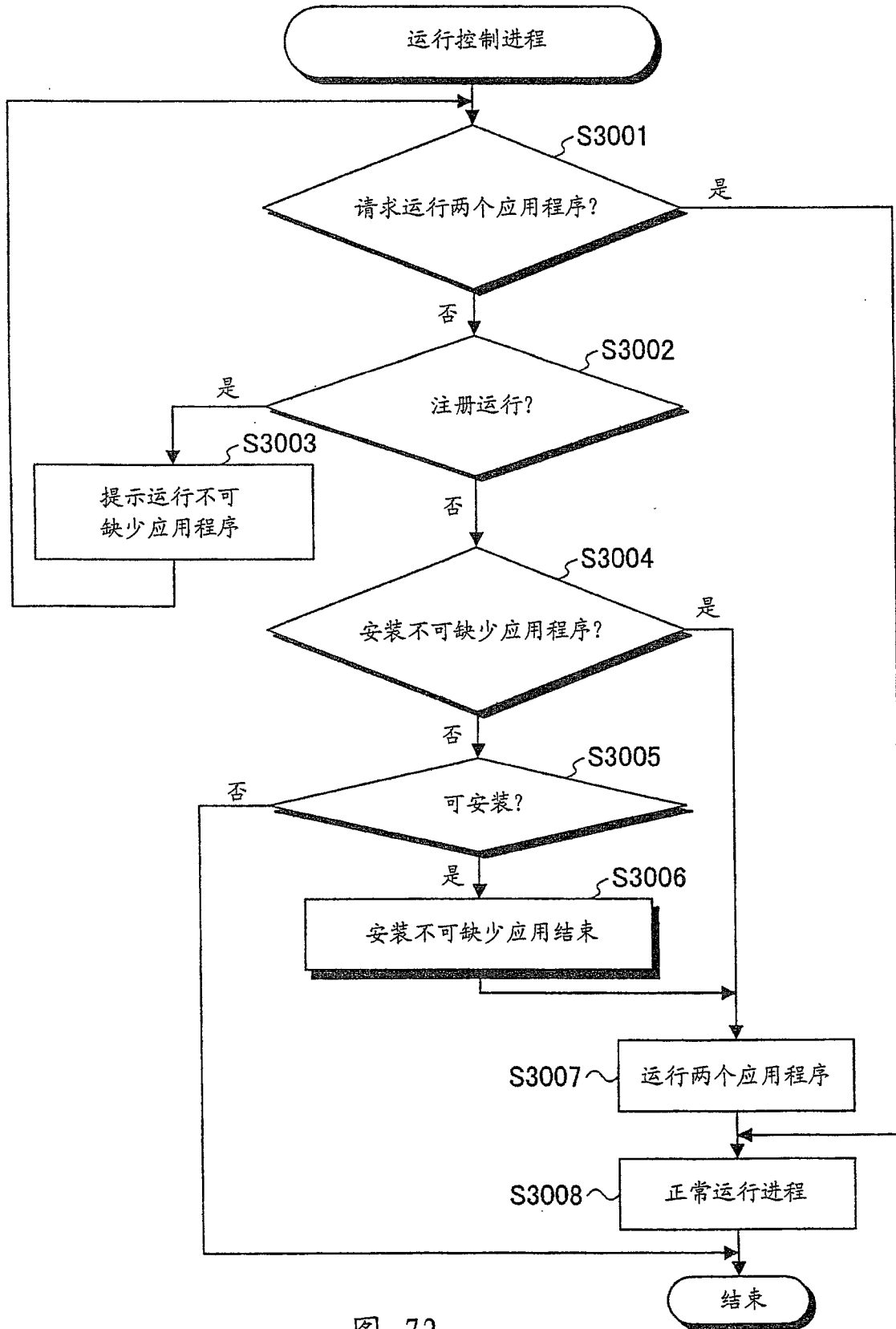


图 72

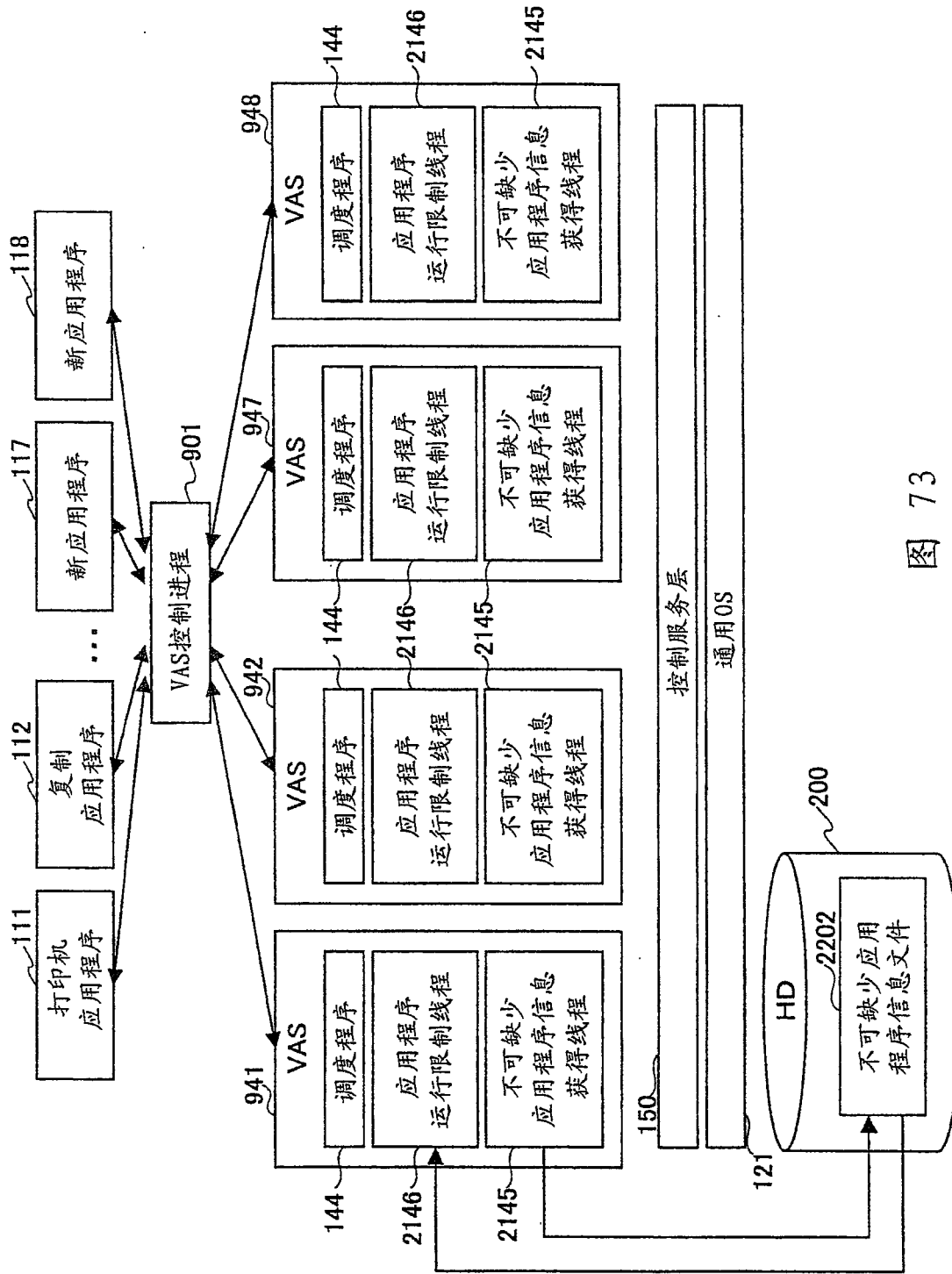


图 73

产品ID	不兼容或不可缺少关系的应用程序
101	105(-3),103(+1)
102	101(-3)
103	101(+1)
104	105(-2),106(+2)
⋮	⋮

图 74

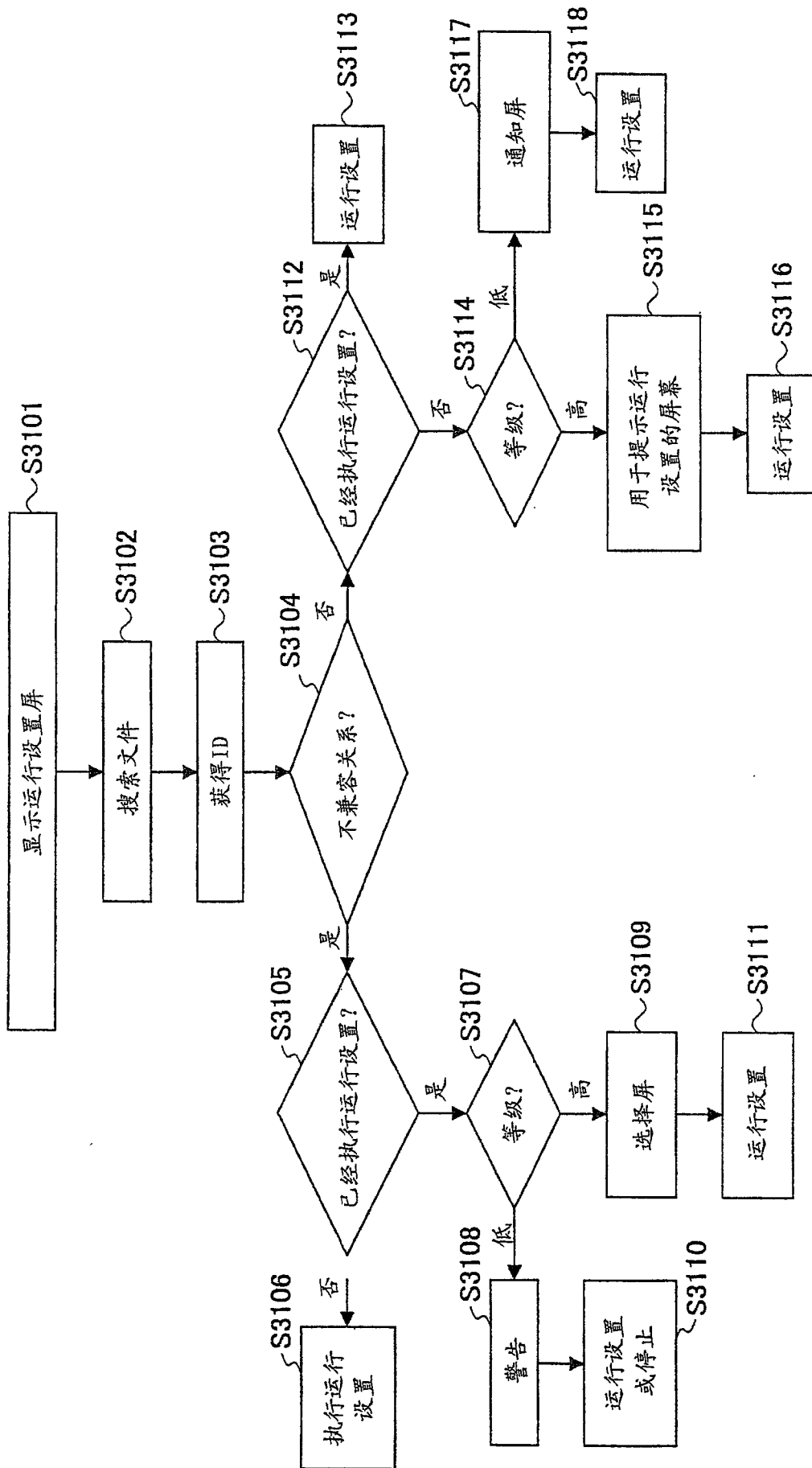


图 75

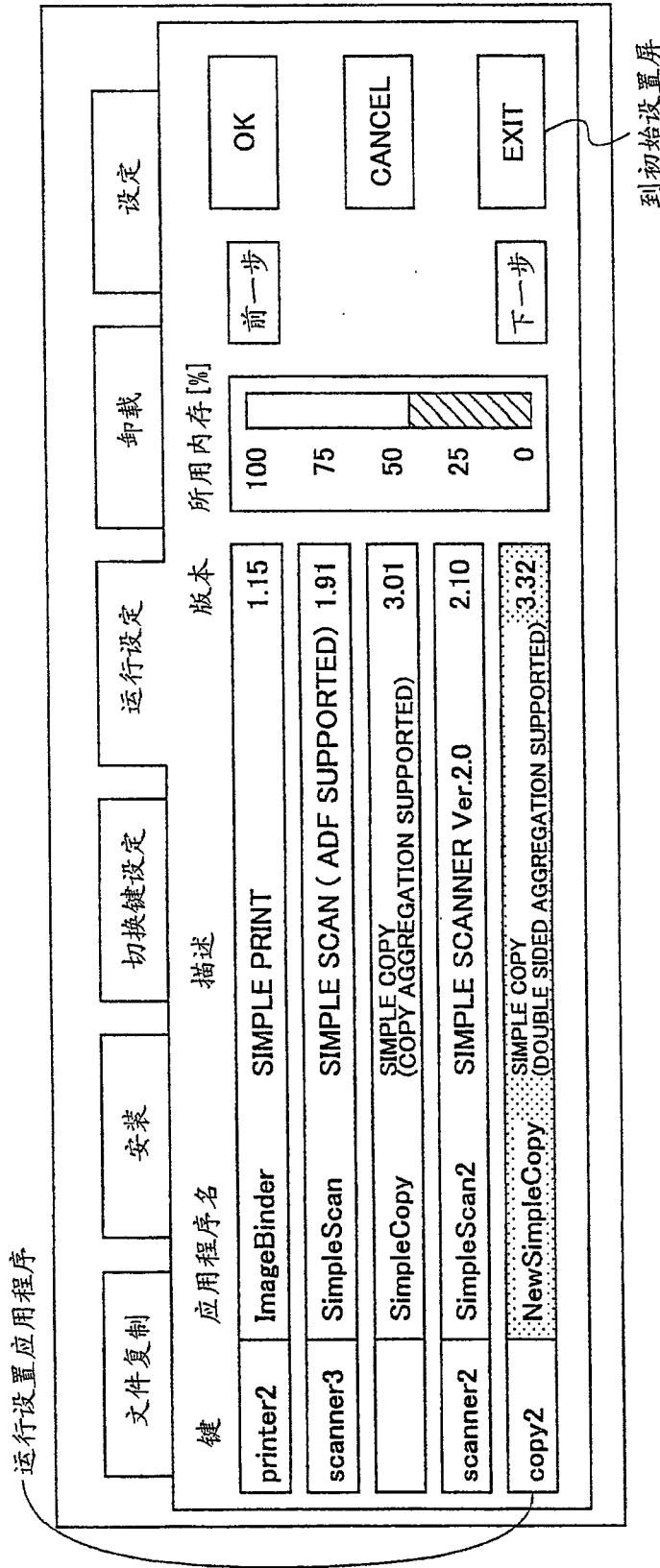


图 76

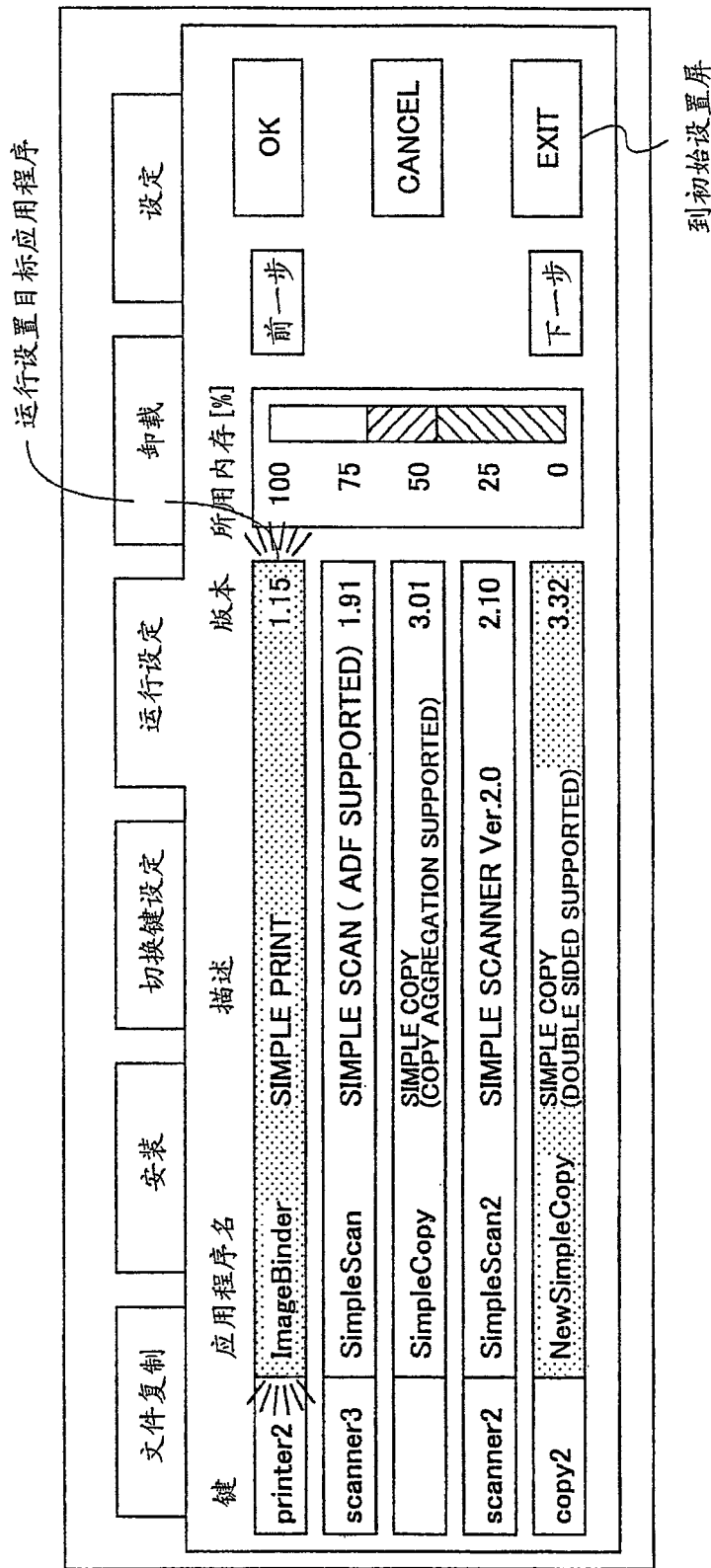


图 77

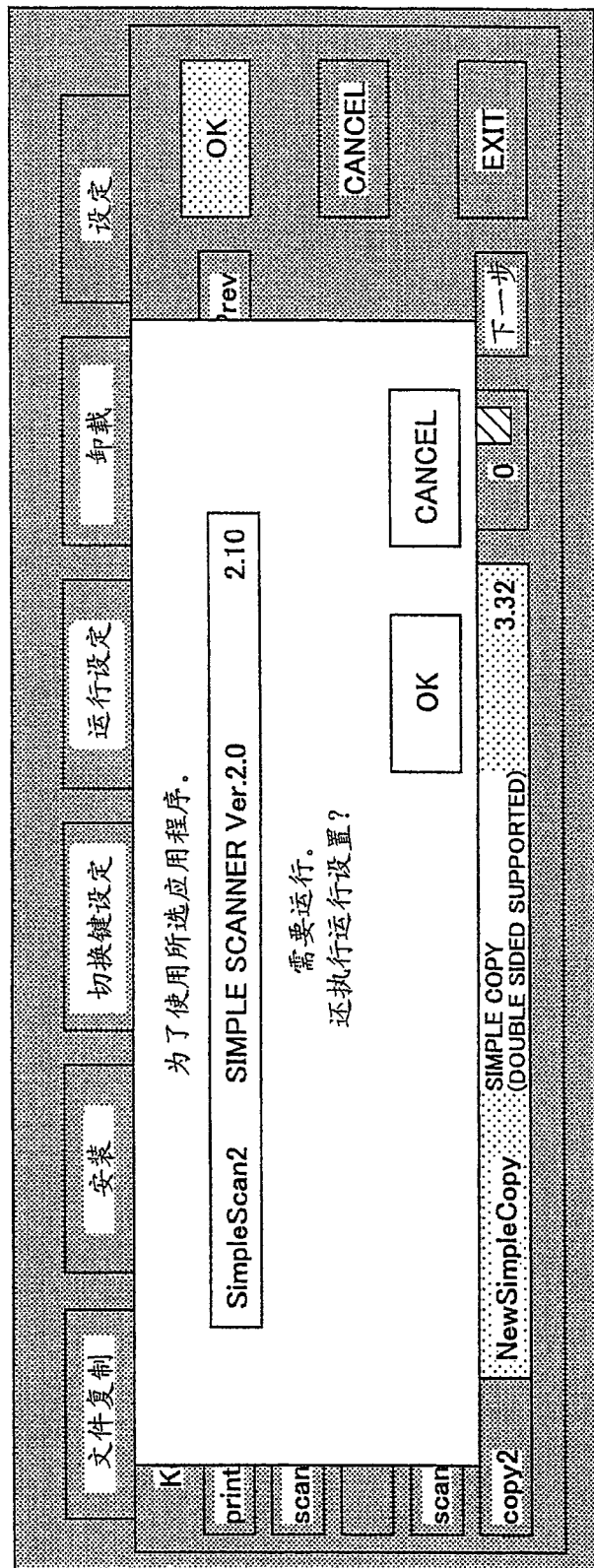


图 78

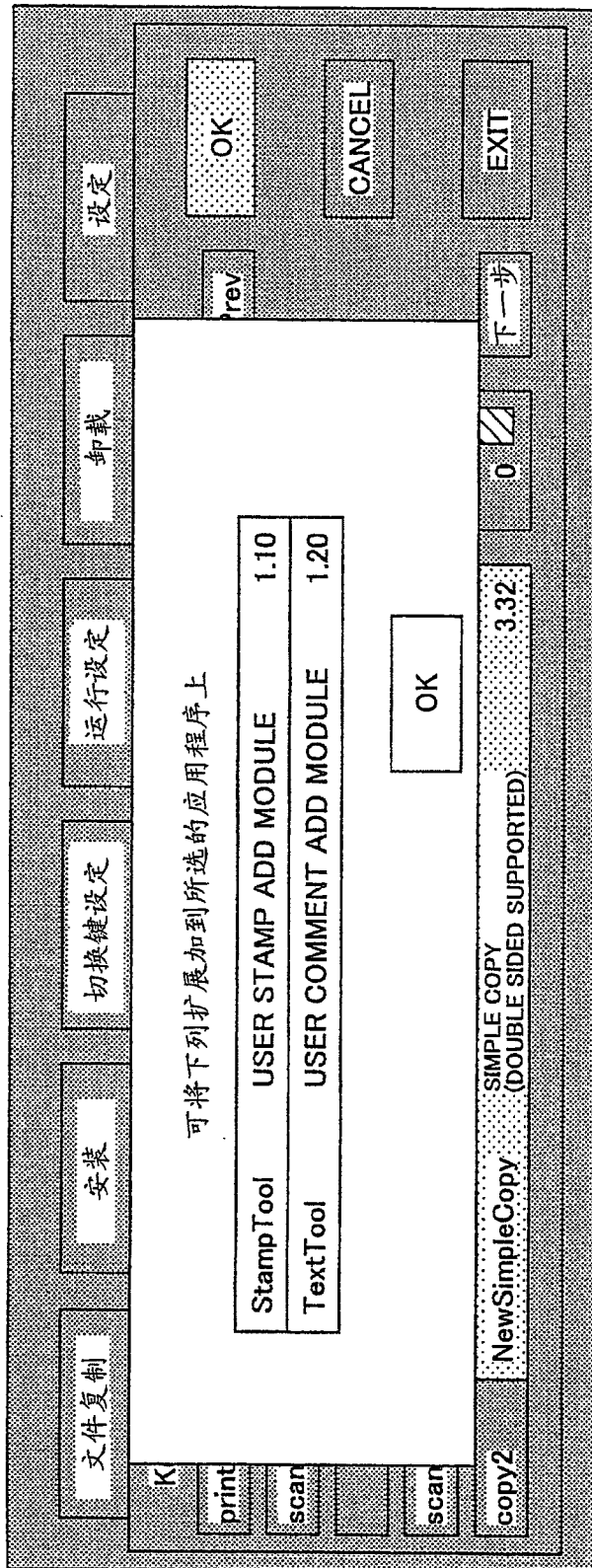


图 79

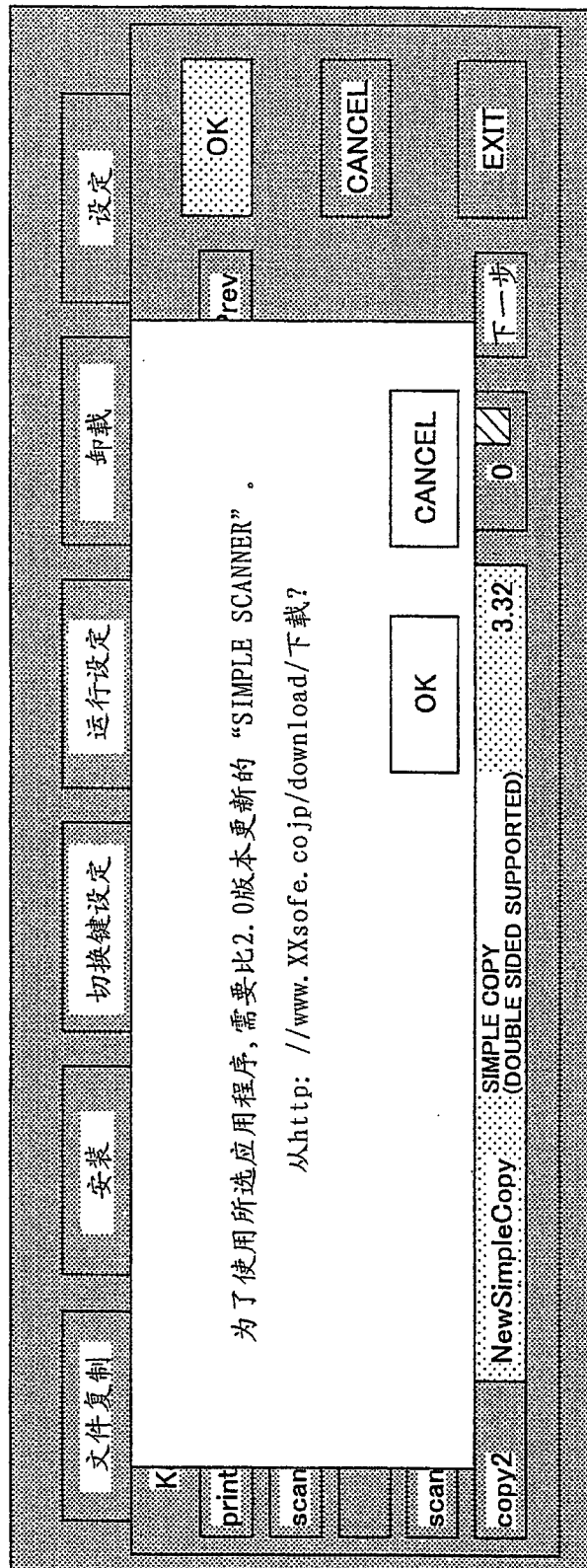


图 80

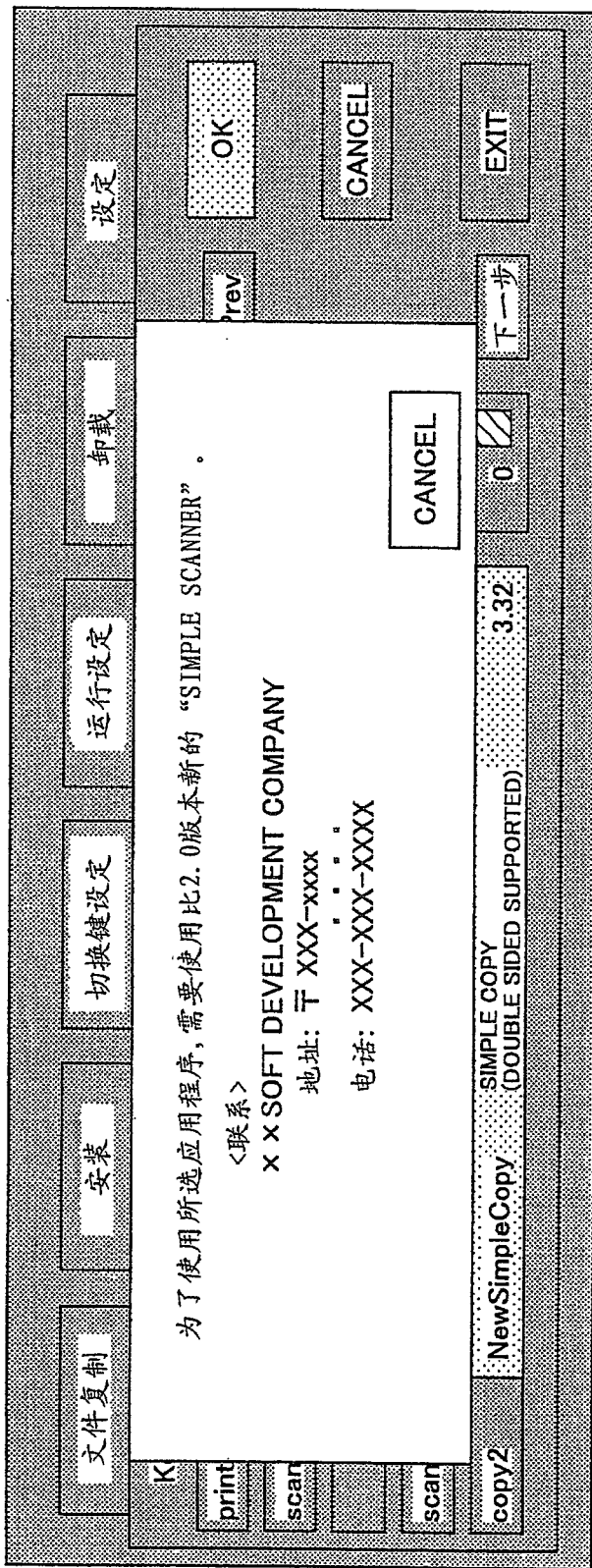


图 81