

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.



# [12] 发明专利说明书

专利号 ZL 200480003313.7

H04N 5/92 (2006.01)  
G11B 20/10 (2006.01)  
G11B 20/12 (2006.01)  
G11B 27/00 (2006.01)  
G11B 27/34 (2006.01)

[45] 授权公告日 2009年4月1日

[11] 授权公告号 CN 100474915C

[22] 申请日 2004.1.30

[21] 申请号 200480003313.7

[30] 优先权

[32] 2003.1.31 [33] US [31] 60/443,876

[86] 国际申请 PCT/JP2004/000891 2004.1.30

[87] 国际公布 WO2004/068854 日 2004.8.12

[85] 进入国家阶段日期 2005.8.1

[73] 专利权人 松下电器产业株式会社

地址 日本大阪府

[72] 发明人 池田航 冈田智之 上坂靖

小塚雅之

[56] 参考文献

WO0036600A1 2000.6.22

CN1381148A 2002.11.20

JP1221073A 1989.9.4

审查员 袁野

[74] 专利代理机构 永新专利商标代理有限公司

代理人 韩宏

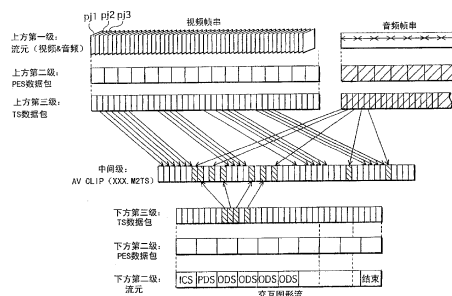
权利要求书 5 页 说明书 58 页 附图 48 页

[54] 发明名称

记录介质、再现装置、记录方法、程序以及再现方法

[57] 摘要

在 BD-ROM 上，记录的是将视频流和音频流多路复用到其中的数字流。视频流包括共同显示视频的多个图像。图形流包括储存状态控制信息 (ICS) 的 PES 数据包以及包含图形数据 (ODS) 的 PES 数据包。图形数据构成交互显示。状态控制信息定义控制从而根据再现进程和用户操作来改变呈现在交互显示上的按钮的状态。ICS 附有表示与交互显示同步的图像的显示定时的 PTS。ODS 附有表示图形数据解码时间的 PTS。在 ODS 中所示的定时先于显示定时。



1、一种记录介质的记录方法，包括：

第一步骤，用于产生视频流；

第二步骤，用于产生图形流；

第三步骤，用于通过多路复用所述视频流和图形流而产生数字流；以及

第四步骤，用于将所述数字流记录到所述记录介质上，

其中，

所述图形流包括多个显示组；

每个所述显示组包括结合在其中的图形数据和状态控制信息；

所述图形数据用于构成交互显示，以及所述状态控制信息用于使所述交互显示响应于所述视频流的再现进程和用户操作而改变为不同的状态；

所述第二步骤包括：

第一子步骤，用于产生每条状态控制信息，包括表示相对应的显示组的类型的类型信息；

第二子步骤，用于产生包括状态控制信息和图形数据的每个所述显示组；以及

第三子步骤，用于产生包括所述多个显示组的所述图形流；  
并且

所述图形流中包括的多条类型信息包括一条类型信息，其表示状态控制信息所属的显示组中包括的图形数据与所述图形流中前一个显示组中包括的图形数据一致。

2、根据权利要求1所述的记录介质的记录方法，其中：

所述图形流是数据包串；

该数据包串包括至少一组数据包，每一组数据包包括包含状态控制信息的数据包以及包含图形数据的数据包；以及

所述包含状态控制信息的数据包附有时间戳，该时间戳表示与所述交互显示同步显示的图像的显示定时。

3、根据权利要求1所述的记录介质的记录方法，其中：

所述图形流是数据包串；

该数据包串包括至少一组数据包，每一组数据包包括包含状态控制信息的数据包以及包含图形数据的数据包；

所述包含状态控制信息的数据包附有时间戳；以及

该时间戳表示在所述视频流的再现时间轴上的一个时间，在该时间之后所述交互显示可用于响应于用户操作的显示。

4、根据权利要求1所述的记录介质的记录方法，其中：

所述交互显示呈现  $n$  个按钮，并且所述  $n$  个按钮中的任意按钮  $i$  具有  $m$  个状态，其中  $i$ 、 $n$  和  $m$  是整数；

所述状态控制信息括  $n$  条按钮信息；

与按钮  $i$  相关联的一条按钮信息  $i$  包括卡嗒声控制信息；以及

所述卡嗒声控制信息表示在按钮  $i$  状态转换时将要再现的一条音频数据，并且表示通过用于音频数据再现的再现装置将要执行的控制。

5、一种用于再现数字流的再现装置，其中将视频流和图形流多路复用到该数字流中，所述再现装置包括：

视频解码器，用于对所述视频流进行解码以获得视频数据；以及

图形解码器，用于对所述图形流进行解码以获得交互显示，其中：

所述图形流包括多个显示组；

每个所述显示组包括图形数据和状态控制信息；

所述状态控制信息用于响应于所述数字流的再现进程和用户操作使所述交互显示改变为不同的状态；

每条状态控制信息包括表示相对应的显示组的类型的类型信息；

所述图形解码器包括：

处理单元，用于对包括在所述图形流中的图形数据进行解码以获得所述交互显示，以及

控制器，用于根据所述状态控制信息来控制所述交互显示的状态；

当执行正常再现时，所述控制器忽略一显示组中包括的图形数据，其中该显示组的类型信息表示该显示组中包括的图形数据与所述图形流中前一个显示组中包括的图形数据相一致；以及

当从在跳转操作中搜索的点开始执行再现时，所述控制器对一显示组进行解码，该显示组位于再现开始点之后并且具有表示该显示组中包括的图形数据与所述图形流中前一个显示组中包括的图形数据相一致的类型信息。

6、根据权利要求 5 所述的再现装置，其中：

所述图形流是数据包串；

该数据包串包括至少一组数据包，每一组数据包包括包含状态控制信息的数据包以及包含图形数据的数据包；

所述包含状态控制信息的数据包附有时间戳，该时间戳表示与所述交互显示同步显示的图像的显示定时；以及

参考附到所述包含状态控制信息的数据包的所述时间戳来执行所述处理单元的解码和所述控制器的控制。

7、根据权利要求 5 所述的再现装置，其中：

所述图形流是数据包串；

该数据包串包括至少一组数据包，每一组数据包包括包含状态控制信息的数据包以及包含图形数据的数据包；

所述包含状态控制信息的数据包附有时间戳；

该时间戳表示在所述视频流的再现时间轴上的一个时间，在该时间之后所述交互显示可用于响应于用户操作的显示；

所述再现装置还包括接收单元，用于接收用户操作；以及

当所述接收单元在当前再现点达到由附到所述包含状态控制信息的数据包的所述时间戳所显示的时间之后接收到用户操作时，所述控制器执行控制以呈现所述交互显示。

8、一种再现数字流的方法，其中将视频流和图形流多路复用到该数字流中，所述方法包括：

对所述视频流进行解码以获得视频数据；以及

对所述图形流进行解码以获得交互显示，其中：

所述图形流包括多个显示组；

每个所述显示组包括图形数据和状态控制信息；

所述状态控制信息用于响应于所述数字流的再现进程和用户操作而使所述交互显示改变为不同的状态，并且包括表示相对应的显示组的类型的类型信息；以及

所述对所述图形流进行解码包括：

对包括在所述图形流中的图形数据进行解码以获得所述交互显示，以及

根据所述状态控制信息来控制所述交互显示的状态，

所述控制所述交互显示的状态包括：

当执行正常再现时，忽略一显示组中包括的图形数据，其中该显示组的类型信息表示该显示组中包括的图形数据与所述图形流

中前一个显示组中包括的图形数据相一致；以及

当从在跳转操作中搜索的点开始执行再现时，对一个显示组进行解码，该显示组位于再现开始点之后并且具有表示该显示组中包括的图形数据与所述图形流中前一个显示组中包括的图形数据相一致的类型信息。

## 记录介质、再现装置、记录方法、程序以及再现方法

### 技术领域

本发明涉及一种记录介质，例如用于电影分销的 BD-ROM，本发明还涉及一种用于这种记录介质的再现装置。更具体而言，本发明涉及交互控制技术的改进。

### 背景技术

长期以来人们就一直希望实现这样一种交互控制，即当再现视频流时按钮出现在显示屏上，并且根据用户对按钮进行的操作来进行再现。DVD 是实现这种再现控制的突破性记录介质。通过使用时间戳建立按钮与视频流的同步出现，设置所述时间戳使按钮出现在视频流的再现时间轴上的特定点。

然而，为了实现交互控制，将用于呈现按钮的图形数据记录到记录介质上是不够的。需要根据用户的操作或视频流再现的进行来控制再现装置从而改变每一个显示在屏幕上的按钮的状态。为了实现交互控制，在 DVD 上记录将音频和视频流多路复用到其中的流（视频对象），并且将包含状态控制信息的 NAVI 包设置在每一个 VOB 的起始处。VOB 包括视频流的一个 GOP，并且还包括音频数据和图形数据，其是对视频流的补充，从 DVD 中同时读取这二者和 GOP。根据用户操作使用状态控制信息来改变每一个显示在屏幕上的按钮的状态。NAVI 包包含定义传输率和缓冲器大小的信息，每一个流在处理 GOP 时需要该信息。由于 DVD 将状态信息储存在 NAVI 包中，所以可以利用 GOP 的时间精确性来改变按钮状态。图 1 示出上述交互控制。在该图中，最低层显示 DVD 上的数据分配。图中所示状态控

制信息包含在 NAVI 包中。在 GOP 的时限期间状态控制信息仍然是有效的,NAVI 包属于该 GOP。每一个图形对象包含在 PES 数据包中,并且与图像同时显示,该图像与图形对象同步。例如在日本专利 NO.2813245 中公开了这种现有技术。

这里,应该注意的是,在 DVD 授权时,直到对视频、音频和图形数据进行编码并且准备多路复用到一个 VOB 中时才确定 GOP 和 VOB 的结构。多路复用是授权的最后阶段。即,在该最后阶段之前不能将状态控制信息合并到 VOB 中,由此在这之前的任何时候都不能对按钮如何在显示屏上变化进行测试。因此,经常出现这样的情况,即在出货之前发现有缺陷并且迫使开发商仓促地进行改正。另外,在测试时间不足的情况下,将复杂动画按钮合并到电影中是危险的。由于上述原因,在当前授权中通常合并相对简单的按钮,例如响应用户操作而改变颜色的按钮。

## 发明内容

本发明的目的是提供一种记录介质,该记录介质允许按钮状态改变以便在授权的早期对其进行测试。

当考虑 DVD 数据读取速度时将状态控制信息存储在每一个 VOB 的 NAVI 包中是一个有效方案。这是因为简单通过读取 NAVI 包来从 DVD 中读取状态控制信息,从而使得读取所需的带宽很低。

然而,当考虑 BD-ROM 的读取速度时,此方案就不是那么有效。BD-ROM 的读取速度远远高于 DVD 的读取速度,从而限制带宽不再重要。

鉴于记录介质的进步以及为了实现上述目的,根据本发明的记录介质将数字流记录在其上,视频流和图形流被多路复用到该数字流中。图形流包括结合在其中的图形数据和状态控制信息。图形数据用来构成交互显示。状态控制信息用来使交互显示响应视频流的再现进



程和用户操作而被改变为不同的状态。根据本发明的记录介质，用于使按钮状态改变的信息与图形数据结合成图形流。这样，一产生图形流，就进行有效性测试以检查按钮状态如何根据再现进程而改变。不再需要等待视频流编码或流多路复用的完成。由于可以在授权的早期（先于流多路复用）进行按钮状态改变的验证测试，所以可以减小这种令人不快的可能性：就在要发货前在记录介质中发现有错误并且迫使开发商仓促改正。而且，由于对图形流进行独立验证测试而与其他流无关，所以为将复杂动画按钮合并到电影中提供更好的环境。

这里，多个显示组中的每一个中的状态控制信息可以包括更新标志。当设为 ON 时，更新标志表示除按钮命令之外，对于状态控制信息和图形数据该显示组与紧挨在其之前的显示组一致。当设为 OFF 时，更新标志表示对于状态控制信息和图形数据该显示组与紧挨在其之前的显示组一致。一激活交互显示上的相关按钮就可以通过再现装置执行按钮命令。

利用上述状态结构，产生的标题使得根据按钮命令再现分支到多个再现路径中的一个。例如，标题是问答比赛，并且两个显示组提出供用户回答的问题。可以构造标题以使得当用户反应迟缓时，分支到用户的越来越多的不利的再现路径中。

这里， $n$  个按钮中的每一个可以具有指定给其的数字。 $n$  条按钮信息中的每一条可以包括指定给相关按钮的数字和标志，该标志表示利用所述数字按钮是否是用数字可选的。利用这种结构，可以将离散数字指定给按钮，这对于产生如棒球运动员名单这样的标题来说是方便的。

更具体地讲，准备这种按钮命令使得再现分支到显示棒球运动员比赛的情景。另外，将运动员号码指定给按钮命令。通过这种设置，通过运动员号码的数字输入接受到特定棒球运动员情景的再现路径。

附图说明

图 1 所示为 DVD 上的交互控制图；

图 2A 所示为根据本发明的记录介质的使用模式图；

图 2B 所示为遥控器 400 的键的示图，该遥控器 400 用于接收用户对交互显示的操作；

图 3 所示为 BD-ROM 的结构图；

图 4 示意性地示出 AV 剪辑的结构图；

图 5 所示为剪辑信息的内部结构图；

图 6 所示为播放列表信息的内部结构图；

图 7 示意性示出通过播放列表信息的间接参考图；

图 8A 所示为图形流的结构图；

图 8B 所示为 ICS 和 ODS 的内部结构图；

图 9 所示为由各种类型的功能段定义的逻辑结构图；

图 10A 所示为定义图形对象的 ODS 的数据结构图；

图 10B 所示为 PDS 的数据结构图；

图 11 所示为交互构成段的数据结构图；

图 12 所示为包含在 DS 中的 ODS 和 ICS 之间的关系图；

图 13 所示为在任意画面 pt1 显示时的合成屏幕图；

图 14 所示为 ICS 中的按钮信息设置的实例图；

图 15 所示为按钮 A-D 的按钮状态转换图；

图 16 所示为由 ODS11、21、31 和 41 定义的图形再定位的实例图；

图 17 所示为由与按钮 A 有关的 ODS11-19 定义的图形表示的实例图；

图 18 所示为包括在 DS 中的 ICS 和 ODS 的实例图；

图 19 所示为 DS 中的 ODS 和按钮状态组的顺序图；

图 20 所示为在由如图 19 所示的按钮状态组所定义的交互显示上的按钮状态转换图；

图 21 所示为 DS 中的 ODS 的顺序图；

图 22 所示为由 ICS 定义的同步显示的时序图；

图 23 所示为在多个 ODS 构成初始交互显示并且 default\_selected\_button\_number 是有效的情况下，DTS 和 PTS 的设置图；

图 24 所示为在多个 ODS 构成初始交互显示并且 default\_selected\_button\_number 是无效的情况下，DTS 和 PTS 的设置图；

图 25 所示为再现装置的内部结构图；

图 26 是由再现装置执行的流水线处理的时序图；

图 27 是在没有指定缺省选择按钮的情况下，由再现装置执行的流水线处理的时序图；

图 28 是用于执行 LinkPL 功能的控制器操作的流程图；

图 29 是功能段装载处理的流程图；

图 30 所示为在跳过操作时如何执行装载处理的示图；

图 31 所示为如何将 DS 10 装载到再现装置的编码数据缓冲器 13 中的示图；

图 32 所示为正常再现的示图；

图 33 所示为当如图 32 所示执行正常再现时如何装载 DS 1、10、20 的示图；

图 34 是由图形控制器 17 执行的处理的主程序的流程图；

图 35 是基于时间戳的同步控制的流程图；

图 36 所示为图形平面 8 的流程图；

图 37 是用于自动激活缺省选择按钮的处理的流程图；

图 38 是用于动画显示的处理的流程图；

图 39 是 UO 处理的流程图；

图 40 是当前按钮变化处理的流程图；

图 41 是数字输入处理的流程图；

图 42 所示为能够产生卡嗒声的再现装置的内部结构图；

图 43 所示为用于再现卡嗒声的 ICS 的数据结构图；

图 44A 所示为包括 button\_info (1)和 button\_info (2)的状态控制信息的示图；

图 44B 所示为包括状态控制信息的 ICS 的读取处理的示图；

图 44C 所示为在交互显示上呈对角线排列的三个按钮（按钮 A、B 和 C）的例子以及与这些按钮有关的按钮状态信息的例子的示图；

图 45A 和 45B 所示为根据读到预载存储器 21 的 IC 的卡嗒声数据的再现控制的示图；

图 45C 所示为在交互显示上水平排列的三个按钮（按钮 A、B 和 C）的例子以及与这些按钮有关的按钮状态信息的例子的示图；

图 46 是根据本发明第三实施例的 BD-ROM 制造过程的流程图；

图 47 所示为根据本发明的一个修改的 ICS 的示图；以及

图 48 所示为定义用于遥控器的每一个键的卡嗒声的 ICS 的示图。

## 具体实施方式

### （第一实施例）

下面对根据本发明第一实施例的记录介质进行说明。首先，对记录介质的使用进行说明，其是实施本发明的一种形式。图 2A 示出根据本发明的记录介质的使用模式。在图 2 中，BD-ROM（蓝光光盘-只读存储器）100 是根据本发明的记录介质。BD-ROM 100 用于为家庭影院系统提供电影，该家庭影院系统包括再现装置 200、电视机 300 和遥控器 400。遥控器 400 接收指示改变交互显示状态的用户操作，并且与根据本发明的记录介质紧密相关。图 2B 示出遥控器 400 的键。如该图所示，遥控器 400 具有上移键、下移键、右移键和左移键。显示在交互显示上的每个按钮具有三种状态：正常状态；选择状态；以及激活状态。上移键、下移键、右移键和左移键用来接收用于使按钮

状态按照例如正常状态→选择状态→激活状态的顺序改变的用户操作。当按钮具有正常状态时，简单显示该按钮。当按钮具有选择状态时，该按钮当前被集中为用户操作但还没有被激活的结果。当按钮具有激活状态时，该按钮被激活。当在交互显示上特定按钮具有选择状态时，上移键用来将显示在当前所选择的按钮上方的按钮设为选择状态。下移键用来将显示在当前所选择的按钮下方的按钮设为选择状态。右移键用来将显示在当前所选择的按钮右侧的按钮设为选择状态。左移键将显示在当前所选择的按钮左侧的按钮设为选择状态。

激活键用来将具有选择状态的按钮设为激活状态（激活当前所选择的按钮）。数字键“0”到“9”用于数字选择从而将已经指定了输入值的按钮设为选择状态。“+10”键用来将数值 10 加在已输入的值上。注意“0”键和“+10”键都用来输入两位数的值。这样，“0”和“+10”键中的一个可以足以代替这二者。

在此结束对使用根据本发明的记录介质的说明。

接着，对记录介质的制作进行说明，其是实施本发明的另一种形式。可以通过改进 BD-ROM 的应用层来实现该记录介质。图 3 示出 BD-ROM 100 的实例结构。

在该图中，BD-ROM 100 显示在第四级上，并且 BD-ROM 的轨道显示在第三级上。在该图中，轨道向外延伸成为直线，尽管轨道实际上是从 BD-ROM 的中心向外盘旋的。轨道包括引入区、卷区、以及引出区。卷区具有物理层、文件系统层和应用层的层模型。在目录结构中，第一级显示 BD-ROM 应用层的格式（应用格式）。如所示那样，BD-ROM 在根目录下具有 BDMV 目录。BDMV 目录包含文件，例如 XXX.M2TS、XXX.CLPT 和 YYY.MPLS。可以通过创建如图所示的应用格式来制作本发明的 BD-ROM。

现在，对在应用格式中使用的文件进行说明。首先，说明 AV 剪辑（clip）（XXX.M2TS 文件）。

AV 剪辑 (XXX.N2TS) 是遵循 MPEG-TS (运动图像专家组-传输流) 格式的数字流, 并且通过多路复用视频流、一个或多个音频流、显示图形流、以及交互图形流来获得所述 AV 剪辑。视频流表示电影的视频。音频流表示电影的音频。显示图形流表示电影的子标题。交互图形流表示用于菜单再现的动态控制程序。图 4 示意性地示出 AV 剪辑的结构。

以如下方式获得中间级上的如图所示的 AV 剪辑。显示在上方第一级上的视频流包括多个视频帧 (图像 pj1、pj2、pj3、...), 并且显示在上方第一级上的音频流包括多个音频帧。分别将视频帧和音频帧转换为显示在上方第二级上的 PES 数据包。进一步将 PES 数据包转换为显示在上方第三级上的 TS 数据包。同样地, 分别将显示在下方第一级上的显示图形流和交互图形流转换为显示在下方第二级上的 PES 数据包, 并且进一步转换为显示在下方第三级上的 TS 数据包。多路复用这些显示在上方第三级和下方第三级上的 TS 数据包以形成 AV 剪辑。

按照与计算机文件相同的方式将如上产生的 AV 剪辑分成多个扩展区, 并且被记录到 BD-ROM 上。AV 剪辑包括一个或多个存取单元, 并且 AV 剪辑的再现可以跳到对应于存取单元的点。存取单元包括一个 GOP (图像组) 以及与 GOP 同时读取的音频帧, 并且是解码的最小单元。GOP 包括三种类型的图像: 双向预测图像 (B 图像), 使用与过去和未来图像的相关性对其进行编码; 预测图像 (P 图像), 使用与过去图像的相关性对其进行编码; 以及内图像 (I 图像), 使用其自身的空间频率特性而不参考帧之间的相关性来对其进行编码。

剪辑信息 (XXX.CLPI) 是每一个 AV 剪辑的管理信息。图 5 示出剪辑信息的内部结构。由于 AV 剪辑是通过多路复用视频流和音频流来获得的并且在称为存取单元的单元中存取的, 所以剪辑数据包括诸如视频流和音频流属性以及 AV 剪辑中的点等信息, 可以对其执行跳

过操作。在该图中，点划线表示引用剪辑信息的结构以便详细说明。如点划线 hn1 所示，剪辑信息 (XXX.CLPI) 包括视频流和音频流的“属性信息”、以及作为用于搜索存取单元的参考表的“EP\_map”。

如点划线 hn2 所示，属性数据包括视频流的属性 (视频属性)，属性的数量 (数量)，以及所有被多路复用在 AV 剪辑中的音频流的属性 (音频属性#1-#m)。如点划线 hn3 所示，视频属性包括显示其中对视频流进行编码的编码方法的信息 (编码)、构成视频流的图像的分辨率 (分辨率)、纵横比 (纵横)、以及帧频 (帧频)。

如点划线 hn4 所示，音频流属性 (音频属性#1-#m) 的每一个包括显示其中对相应的音频流进行编码的编码方法的信息 (编码)、音频流的信道数 (Ch.)、音频流的语言 (Lang.)、以及采样频率。

EP\_map 是用于通过时间来间接参考跳过操作中的可访问的点的地址的参考表。如点划线 hn5 所示，EP\_map 包括多个项 (存取单元#1 项、存取单元#2 项、存取单元#3 项、...)、以及项数 (数目)。如点划线 hn6 所示，每个项示出再现开始时间和相应存取单元的地址 (注意可以另外显示存取单元中的第一个 I 图象的大小 (I-大小))。通过位于存取单元的起始处的图像的时间戳 (显示时间戳) 来显示再现开始时间。通过相应 TS 数据包的序列号 (SPN: 源数据包号) 来显示地址。由于是变长编码，各包含 GOP 的存取单元在大小和再现周期上都不一致。然而，参考与存取单元相对应的项，可以搜索位于与任何给定的再现时间相对应的点上的存取单元，从而可以在所搜索的存取单元中从第一图像开始再现。注意文件名“XXX.CLPI”中的“XXX”与 AV 剪辑的名字相同，剪辑信息与该 AV 剪辑相对应。在该图中，AV 剪辑的文件名是“XXX”，其意味着剪辑信息 (XXX.CLPI) 与 AV 剪辑 (XXX M2TS) 相对应。在此结束对剪辑信息的说明。接下来，说明播放列表信息。

YYY.MPLS (播放列表信息) 是用作定义再现路径的播放列表的

表，并且包括多条播放项信息（播放项信息#1、#2、#3、...#n）和播放项信息的数量（数目）。图 6 示出播放列表信息的内部结构。播放列表信息示出用于由播放项信息各自定义的再现的一个或多个逻辑段。如点划线 hs1 指示引用播放项信息的结构以便详细显示。如该图所示，播放项数据包括：“clip\_information\_file\_name”，显示 AV 剪辑中的再现段的文件名，再现段的起始-时间和终止-时间属于该 AV 剪辑；“clip\_codec\_identifier”显示对 AV 剪辑进行编码的编码方法；“in\_time”显示与再现段的起始点相对应的时间；“out\_time”，显示与再现段的终止点相对应的时间。

播放项信息的一个特征是其符号惯例。即，使用 EP\_map 作为参考表，由通过时间的间接参考来定义再现段。图 7 是示意性地示出通过时间的间接参考。在该图中，AV 剪辑包括多个存取单元。剪辑信息中的 EP\_map 指定每个存取单元的扇区地址，如箭头 ay1、ay2、ay3、以及 ay4 所示。箭头 jy1、jy2、jy3、以及 jy4 中的每一个是间接参考存取单元的示意性表示。简而言之，每一条播放项信息具有经由 EP\_map 通过时间指定的参考(箭头 jy1、jy2、jy3、以及 jy4)、包含在 AV 剪辑中的相应存取单元的地址。

由一组播放项信息-剪辑信息-AV 剪辑指定的 BD-ROM 上的再现段称为“播放项”。由一组 PL 信息-剪辑信息-AV 剪辑指定的 BD-ROM 上的再现逻辑单元称为“播放列表（在下文中称为 PL）”。将记录在 BD-ROM 上的电影分割成 PL 的逻辑单元。由于将电影分割成逻辑单元，所以可以定义这种指定情景的 PL，在所有这些情景中特定人物出现。以这种方式，独立地根据主电影，可以容易地制造另一部电影，其中人物总是出现。

由于记录在 BD-ROM 上的电影具有上述的逻辑结构，电影中的特定情景的 AV 剪辑可以容易地“重复利用”或在另一部电影中使用。

接下来，对交互图形流进行说明。图 8A 所示为图形流的结构图。



在第一级上，示出构成 AV 剪辑的一串 TS 数据包。在第二级上，示出构成图形流的一串 PES 数据包。通过连接 TS 数据包的有效载荷形成显示在第二级上的 PES 数据包，所述 TS 数据包在显示在第一级上的 TS 数据包串中具有预定的 PID。注意不对显示图形流进行说明，因为它不是本发明的要点。

在第三级上，示出图形流的结构。图形流包括功能段，该功能段包括 ICS（交互合成段）、PDS（调色板定义段）、ODS（对象定义段）、以及 END（显示组段的结束）。在这些功能段中，ICS 是屏幕合成段，而 PDS、ODS、以及 END 是定义段。每一个功能段与 PES 数据包是一一对应，或一对多对应。即，在被转换为单一的 PES 数据包之后，一个功能段被记录在 BD-ROM 100 上，或被分割并被转换为多个 PES 数据包。

图 8B 所示为通过转换功能段来获得 PES 数据包的示图。如图 8B 所示，每一个 PES 数据包包括包头和有效负载。有效负载是功能段的实体，并且包头包含与该功能段有关的 DTS 和 PTS。在下文中，包含在含有功能段的 PES 数据包的头部中的 DTS 和 PTS 称为该功能段的 DTS 和 PTS。

这些不同类型的功能段定义如图 9 所示的逻辑结构。在该图中，功能段显示在第三级上，显示组显示在第二级上，并且时期（时期）显示在第一级上。

显示在第二级上的每一个显示组（以下为“DS”）是一组共同构成一个完整屏图形的功能段。虚线 hk2 表示 DS，第三级上的功能段属于该 DS。如从该图中所见的那样，一系列功能段，ICS-PDS-ODS-END 构成一个 DS。当从 BD-ROM 中读取这些构成 DS 的功能段时，再现装置就可以产生一屏图形。

显示在第一级上的时期指一段时间间隔，或者分配给所述时期的一组数据，在这段时间间隔内，在 AV 剪辑的再现时间轴上必须保持

存储器管理的连续性。这里提到的存储器包括用于存储一屏图形的图形平面以及用于存储未压缩图形数据的对象缓冲器。连续的存储器管理是指在整个时期中既不除去图形平面也不除去对象缓冲器，并且仅仅在图形平面的预定矩形区域中除去和显示图形（除去是指清除整个图形平面和整个对象缓冲器）。在整个时期中固定该矩形区域的大小和位置。只要在图形平面的该固定矩形区域中除去和显示图形，就确保无缝再现。也就是说，时期是 AV 剪辑的再现时间轴上的时间单位，在这期间，确保无缝再现。为了改变图形平面中的图形显示区域，必须确定再现时间轴上的改变点，并且根据该点向前设置新的时期。在这种情况下，在两个时期之间的边界不是无缝的。

这里使用的无缝再现表示利用预定数量的视频帧来完成图形的除去/显示。在交互图形流的情况下，视频帧的数量是四到五。根据固定区域与整个图形平面的比率以及对象缓冲器和图形平面之间的传输率来确定视频帧的数量。

在该图中，虚线 hk1 和 hk2 表示时期，显示在第二级上的功能段属于该时期。示出一系列 DS 构成一个显示在第一级上的时期，该系列 DS 是时期起始 DS、获得点 DS、以及正常情况 DS。这里，“时期起始”、“获得点”、以及“正常情况”是 DS 的类型。尽管在图 9 中获得点 DS 在正常情况 DS 之前，但是可以以相反的顺序设置它们。

“时期起始” DS 提供显示效果“新显示”，并且表示新时期的开始。这样，时期起始 DS 包含下一个屏幕合成所需要的所有功能段。时期起始 DS 设置在 AV 剪辑中的点上，例如电影中一段的开始，其中可以对该 AV 剪辑执行跳过操作。

获得点 DS 提供显示效果“显示刷新”，并且与在前的时期起始 DS 有关。存在两种类型的获得点 DS：“复制”和“继承”。复制类型 DS 与在前的时期起始 DS 完全一致。继承 DS 继承在前的时期起始 DS 的功能段，但具有不同的按钮命令。尽管不是时期的开始，但是

获得点 DS 包含下一个屏幕合成所需要的所有功能段。这样，当从获得点 DS 开始再现时，可以可靠地显示图形。也就是说，获得点 DS 能够从时期的中点进行屏幕合成。

获得点 DS 设置在可以进行跳过操作的点中。这种点的例子包括可以通过时间搜索而指定的一个点。时间搜索是以分钟/秒为单位来确定再现点位置的操作，该再现点与用户输入的时间相对应。由于以诸如十分钟和十秒的相对较大的单位进行用户输入，所以可搜索的再现点位于 10 分钟和 10 秒的间隔内。通过将获得点 DS 设置在这种通过时间搜索可搜索到的点中，当进行时间搜索时可以平稳地显示图形流。

正常情况 DS 提供显示效果“显示更新”，并且只包含与在前屏幕合成的差异。例如，如果由 DS (v) 定义的按钮具有与由紧前一个 DS (u) 定义的按钮相同的图形表示，但具有不同的状态控制。在这种情况下，DS (v) 只包含 ICS 或 ICS 和 PDS，并且用作正常情况 DS。通过这种设置，不要求正常情况 DS 包含重叠的 ODS，这导致存储在 BD-ROM 上的数据量的减少。由于正常情况 DS 只包含差异，所以不能单独利用正常情况 DS 显示图形。

由上述 DS 定义的交互显示呈现出 GUI 组成部分。DS 的交互性指的是能够根据用户操作来改变每个 GUI 组成部分的状态。在该实施例中，通过用户操作而进行交互的 GUI 组成部分称作按钮。每个按钮具有正常状态、选择状态和激活状态。使用多个未压缩的图形数据给出每个按钮状态，其被称为“图形对象”。一个按钮的一种状态与多个用于显示动画的图形对象有关。

现在，对定义段进行说明（ODS：对象定义段和 PDS：调色板定义段）。

ODS 是定义图形对象的信息，这将在后面进行说明。由于记录在 BD-ROM 上的 AV 剪辑的特征在于可与高清晰度电视相比的高图像

质量，图形对象具有  $1920 \times 1080$  像素的高分辨率。通过 8 位长的给定值定义每个像素的颜色，该给定值表示红色差分量（Cr 值）、蓝色差分量（Cb 值）、亮度分量（Y 值）、以及透明度（T 值）。该结构允许将每个像素设为 16777216 种颜色中任意 256 种颜色的一种。

ODS 具有如图 10A 所示的数据结构。如该图所示，ODS 包括以下字段：“segment\_type”，表示该段的类型是 ODS；“segment\_length”表示 ODS 的数据长度；“object\_id”：在时期中唯一地标识与 ODS 有关的图形对象；“object\_version\_number”：在时期中表示 ODS 的版本；“last\_in\_sequence\_flag”；以及“object\_data\_fragment”，包含与部分或所有图形对象相对应的连续的一系列字节。

object\_id 字段在时期中唯一地标识与 ODS 有关的图形对象。在由多个 ODS 定义的多个图形对象构成一系列动画的情况下，将序列 object\_id 值分配给这些 ODS。

现在更加详细地参考 last\_in\_sequence\_flag 字段和 object\_data\_fragment 字段。由于对 PES 数据包的有效负载的限制，因此存在单个 ODS 不能执行构成一个按钮的一个未压缩图形数据的情况。如果情况果真如此，那么分割图形数据，并且由 object\_data\_fragment 字段中的 ODS 定义每一段。这里，除了随后一段的每一段的大小相等。即，最后一段的大小小于或等于以前的段的大小。包含这些图形对象的段的 ODS 顺次出现在 DS 中。last\_in\_sequence\_flag 表示图形对象的结束。尽管上述 ODS 数据结构是以将各段没有空隙地存储在连续的 PES 数据包中的方法为基础，但是可以更换地将这些段存储在 PES 数据包中，从而使 PES 数据包之间留有一些空隙。这里结束对 ODS 的描述。

接下来，说明 PDS。PDS 是定义用于颜色转换的调色板的信息。图 10B 示出 PDS 的数据结构。如该图所示，每个 PDS 包括以下字段：“segment\_type”，表示当设为值“0x15”时该段的类型是 PDS；

“segment\_length”，表示 PDS 的数据长度；“pallet\_id”，唯一地标识包含在 PDS 中的调色板；“palette\_version\_number”，在时期中表示 PDS 的版本；以及“pallet\_entry”，表示色差红 (Cr\_value)、色差蓝 (Cb\_value)、亮度 (Y\_value)、以及透明度 (T\_value)。

接下来，说明 ICS。ICS 是定义交互显示的组成的功能段。ICS 具有如图 11 所示的数据结构。如该图所示，ICS 包括以下字段：“segment\_type”、“segment\_length”、“composition\_number”、“composition\_state”、“command\_update\_flag”、“composition\_time\_out\_pts”、“selection\_time\_out\_pts”、“UO\_mask\_table”、“animation\_frame\_rate\_code”、“default\_selected\_button\_number”、“default\_selected\_button\_number”、以及一组“button\_info (1)、(2)、(3)”。

将 composition\_number 字段设为 0-15 中的一个值，该值表示更新 DS，ICS 属于该 DS。

composition\_state 字段表示以 ICS 开始的 DS 是正常情况 DS、获得点 DS、或时期开始 DS。

command\_update\_flag 字段表示从在先前 ICS 中定义的那些按钮命令改变该 ICS 中的按钮命令。例如，ICS 所属的 DS 是获得点 DS，该 ICS 通常等于在紧前一个 ICS 中定义的那些按钮命令。然而，通过将 command\_update\_flag 设为 ON，该 ICS 可以定义与在前一个 ICS 中定义的那些按钮命令不同的按钮命令。当不同的按钮命令与相同的图形对象相关时，command\_update\_flag 设为 ON。

composition\_time\_out\_pts 字段表示交互显示的结束时间。在结束时，交互组成不再有效，并且由此不再显示。利用再现时间轴上的视频流的帧准确度优先表示 composition\_time\_out\_pts。

selection\_time\_out\_pts 字段表示有效按钮选择周期的终止时间。在

selection\_time\_out\_pts 时, 激活由 default\_selected\_button\_number 指定的按钮。 selection\_time\_out\_pts 的值小于或等于 composition\_time\_out\_pts 的值。利用再现时间轴上的视频流的帧准确度表示 selection\_time\_out\_pts。

UO\_mask\_table 在 DS 期间确定用户操作的允许/禁止, ICS 属于该 DS。当将该字段设为“禁止”时, 用户对再现装置的相应的操作是无效的。

animation\_frame\_rate\_code 字段规定应用于动画按钮的帧频。通过视频帧频除以 animation\_frame\_rate\_code 字段的值给出动画帧频。当将该字段设为“00”时, 针对处于非动画状态下的每个按钮仅仅显示由 start\_object\_id\_XXX 指定的图形对象。

当交互显示的表现开始时, default\_selected\_button\_number 字段表示缺省选择的按钮数。当将该字段设为值“0”时, 自动激活由存储在再现装置的寄存器中的按钮数指定的按钮。在另一方面, 当将该字段设为非“0”值时, 该值表示有效的按钮数。

当在由 selection\_time\_out\_pts 字段所定义的时间之前没有按钮被用户激活时, default\_selected\_button\_number 字段表示被自动激活的按钮。当将该字段设为“FF”时, 在由 selection\_time\_out\_pts 字段所定义的时间激活当前选择的按钮。当将该字段设为“00”时, 没有按钮被自动激活。如果将该字段设为不同于“FF”和“00”的值时, 该值被认为是有效按钮数。

button\_info 字段提供定义显示在交互显示上的按钮的信息。在该图中, 点划线 hp1 表示引用 button\_info (i) 的数据结构以便更详细地显示。button\_info (i) 包含关于由 ICS 定义的 button (i) 的信息。以下, 对构成 button\_info (i) 的信息项进行说明。

button\_number 字段表示在 ICS 中唯一标识 button (i) 的值。

numerically\_selectable\_flag 字段表示是否可以用数字选择 button

(i)。

`auto_action_flag` 字段表示是否自动激活 `button (i)`。当将 `auto_action_flag` 设为 ON (位值为“1”) 时, 不将 `button (i)` 转变为选择状态而是直接转变为激活状态。在另一方面, 当将 `auto_action_flag` 设为 OFF (位值为“0”) 时, 不将 `button (i)` 转变为激活状态而是在被用户选择时转变为选择状态。

`button_horizontal_position` 字段和 `button_vertical_position` 字段分别规定交互显示中的 `button (i)` 的左上方像素的水平和垂直位置。

当 `button (i)` 处于选择状态时, `upper_button_numbe` 字段规定在按上移键时接收选择状态的按钮的按钮数。如果将该字段设为与 `button (i)` 相同的按钮数, 则忽略用户对上移键的操作。

同样地, 当 `button (i)` 处于选择状态时, `lower_button_number` 字段、`left_button_number` 字段、`right_button_number` 字段分别规定在按下移键、左移键、以及右移键时接收选择状态的按钮的按钮数。如果将这些字段设为与 `button (i)` 相同的按钮数, 则忽略用户对相应键的操作。

`start_object_id_normal` 规定连续分配给一组 ODS 的 `object_id` 中的第一个, 该组 ODS 构成处于正常状态下的 `button (i)` 的动画。

`end_object_id_normal` 字段规定连续分配给一组 ODS 的 `object_id` 中的最后一个, 该组 ODS 构成处于正常状态下的 `button (i)` 的动画。如果 `end_object_id_normal` 字段规定与 `start_object_id_normal` 相同的值, 则由该值标识的图形对象的静态图像表现为 `button (i)`。

`repeat_normal_flag` 字段规定是否将连续地重复处于正常状态下的 `button (i)` 的动画。

`start_object_id_selected` 字段规定连续分配给一组 ODS 的 `object_id` 中的第一个, 该组 ODS 构成处于选择状态下的 `button (i)` 的动画。如果 `end_object_id_selected` 字段规定与 `start_object_id_selected` 相同

的值，则由该值标识的图形对象的静态图像表现为 button (i)。

end\_object\_id\_selected 字段规定连续分配给一组 ODS 的 object\_id 中的最后一个，该组 ODS 构成处于选择状态下的 button (i) 的动画。

repeat\_selected\_flag 字段规定是否将连续地重复处于选择状态下的 button (i) 的动画。

start\_object\_id\_activated 字段规定连续分配给一组 ODS 的 object\_id 中的第一个，该组 ODS 构成处于激活状态下的 button (i) 的动画。

end\_object\_id\_activated 字段规定连续分配给一组 ODS 的 object\_id 中的最后一个，该组 ODS 构成处于激活状态下的 button (i) 的动画。

接下来，对按钮命令进行说明。

当激活 button (i) 时，执行每一个按钮命令。

按钮命令包括指示再现装置再现 PL 或播放项的一个命令。这种指示再现装置再现 PL 或播放项的命令称作 LinkPL 命令。当执行该命令时，从由第二参数 (argument) 指定的点开始再现由第一参数指定的播放列表。

格式：LinkPL (第一参数、第二参数)

第一参数是指定将要被再现的 PL 的 PL 数。第二参数作为再现开始点指定 PL 中的播放项、章节、或标志。

用于指定播放项作为再现开始点的 LinkPL 功能是“PlayItem () 处的 LinkPL”。

用于指定章节作为再现开始点的 LinkPL 功能是“Chapter () 处的 LinkPL”。

用于指定标志作为再现开始点的 LinkPL 功能是“Mark () 处的 LinkPL”。

按钮命令还可以包括指示再现装置获得或设置装置状态的命令。



通过 64 个播放器状态寄存器（它们的值称作 PSR）和 4096 个通用寄存器（它们的值称作 GPR）来表示再现装置的状态。利用如下命令 (i) - (iv)，获得设置到寄存器的特定值或寄存器的值。

(i) 获得播放器状态寄存器命令的值

格式：获得播放器状态寄存器的值（参数）

该功能返回用参数指定的播放器状态寄存器的值。

(ii) 设置播放器状态寄存器命令的值

格式：设置播放器状态寄存器的值（第一参数、第二参数）

该功能将用第一参数指定的播放器状态寄存器的值设置为第二参数。

(iii) 获得通用寄存器命令的值

格式：获得通用寄存器的值（参数）

该功能返回用参数指定的通用寄存器的值。

(iv) 设置通用寄存器命令的值

格式：设置通用寄存器的值（参数）

该功能将用第一参数指定的通用寄存器的值设置为第二参数。

在此结束对 ICS 结构的说明。接下来，对由 ICS 定义的交互控制的具体例子进行说明。例子涉及如图 12 所示的 ODS 和 ICS。图 12 所示为包含在 DS (n) 中的 ODS 和 ICS 之间的关系。DS (n) 包括 ODS 11-19、21-29、31-39、41-49。ODS 11-19 用于反映处于三个状态中的每一个状态下的按钮 A。ODS 21-29 用于反映处于三个状态中

的每一个状态下的按钮 B。ODS 31-39 用于反映处于三个状态中的每一个状态下的按钮 C。ODS 41-49 用于反映处于三个状态中的每一个状态下的按钮 D（参见该图中的右括弧）。ICS 中的 `button_info` (1)、(2)、(3) 和 (4) 提供对按钮 A-D 的状态控制的说明（参见该图中的箭头 `bh1`、`bh2`、`bh3` 和 `bh4`）。

假设将与包含在如图 13 所示的视频流中的图像 `pt1` 的显示时间同步地执行由 ICS 定义的控制。在这种情况下，图像 `pt1` 叠置 (`gs1`) 在由按钮 A-D 组成的交互显示 `tm1` 上从而产生合成屏 `gs2`。以这种方式，与特定的视频图像同步显示由多个按钮组成的交互显示。这样，ICS 使得可以以对于用户来说更逼真的方式显示按钮。

图 14 示出实现如图 15 所示的按钮 A-D 的状态转换的例子。在图 15 中，箭头 `hh1` 和 `hh2` 各在视觉上表示由 `button_info` (1) 中的 `neighbor_info` ( ) 定义的状态转换。在 `neighbor_info` ( ) 中，`lower_button_number` 字段指定按钮 C。因此，当按钮 A 处于选择状态时，用户对下移键进行操作（图中的 `up1`），按钮 C 接收选择状态（图中的 `sj1`）。同样地，`right_button_number` 字段指定按钮 B。因此，当按钮 A 处于选择状态时，用户对右移键进行操作（图中的 `up2`），按钮 B 接收选择状态（图中的 `sj2`）。

图 15 中的箭头 `hh3` 在视觉上表示由 `button_info` (3) 中的 `neighbor_info` ( ) 定义的状态转换。在 `neighbor_info` ( ) 中，`upper_button_number` 字段指定按钮 A。因此，当按钮 C 处于选择状态时，用户对上移键进行操作（图中的 `up3`），使按钮 A 返回到选择状态。

接下来，对按钮 A-D 的图形显示进行说明。ODS 11、21、31 以及 41 各显示图 16 所示的图像。与按钮 A 有关的 ODS 11-19 显示图 17 所示的图像。`start_object_id_normal` 和 `end_object_id_normal` 字段分别指定 ODS 11 和 13。因此，通过序列 ODS 11-13 以动画来显示处于

正常状态下的按钮 A。另外，在 `button_info (1)` 的 `selected_state_info ( )` 中，`start_object_id_selected` 和 `end_object_id_selected` 字段分别指定 ODS 14 和 16。因此，通过序列 ODS 14-16 以动画来显示处于选择状态下的按钮 A。当用户将按钮 A 设为选择状态时，为按钮 A 显示的图像从 ODS 11-13 改变为 ODS 14-16。通过将 `repeat_normal_flag` 和 `repeat_selected_flag` 设为值“1”，连续重复 ODS 11-13 和 ODS 14-16 的动画，如在图中用“→A”、“A→”、“→B”、“B→”所表示。

如上所述，按钮 A-B 的每一个都与可以用动画显示的 ODS 序列有关。这样，利用如上描述控制的 ICS，实现这样的按钮状态控制，即用作按钮的人物图像响应于用户的操作来改变它的面部表情。

接下来，对 `numerically_selectable_flag` 字段的应用进行说明。图 18 所示为包括在 DS 中的 ICS 和 ODS 的例子的示图。ODS 31-33 表示三个棒球运动员的图像以及他们的名字和运动员号码，如图的上半部分所示。属于 DS 的 ICS 包括三条按钮信息，`button_info (1)`、`(2)` 和 `(3)`。`button_info (1)`、`(2)` 和 `(3)` 的 `start_object_id` 字段分别指定 ODS 31、ODS32 和 ODS33。同样地，分别将 `button_info (1)`、`(2)` 和 `(3)` 的 `button_number` 字段设为值“99”、“42”和“94”。这里，将 `button_info (1)`、`(2)` 和 `(3)` 的 `numerically_selectable_flag` 字段都设为值“1”。通过上述设置，由 `button_info (1)`、`(2)` 和 `(3)` 定义的每一个按钮都是用数字可选择的。当用户在遥控器 400 上输入值“99”时，用 Beginner's Luck 先生的图像显示的按钮接收选择状态。可以通过连续按“4”键和“9”键或者通过连续按“9”键一次和“+10”键四次来输入值“99”。或者，当用户输入值“42”时，Careless Mistake 先生的按钮接收选择状态。当用户输入值“94”时，Dead Stock 先生的按钮接收选择状态。

这里，将 `button_info (1)`、`(2)` 和 `(3)` 的 `auto_action_flag` 字段设为值“1”。在这种情况下，当被选择时，上述三个按钮接收激活状态

而不是选择状态，并且执行包括在 `button_info` 中的按钮命令（LinkPL（PL#21）、LinkPL（PL#22）、LinkPL（PL#23））。结果，再现所执行的按钮命令与其链接的 PL#21、#22 或#23。如果那些 PL 定义了上述运动员击球和投球的情景，则在输入与一个运动员号码相对应的数值时再现这些情景中的每一个。由于利用输入运动员号码，按钮是可选择的，所以对用户来说是很容易识别的，所以改善了用户的可操作性。

接下来，对 DS 中的 ODS 的顺序进行说明。如上所述，通过 ICS 将属于一个 DS 的 ODS 与一个按钮的每一个状态联系起来。根据每一个 ODS 与其关联的按钮状态来确定 DS 中的 ODS 的顺序。

更详细地，将属于一个 DS 的 ODS 分组成：（1）用于表示处于正常状态下的按钮的 ODS；（2）用于表示处于选择状态下的按钮的 ODS；以及（3）用于表示处于激活状态下的按钮的 ODS。用于表示相应状态的每一组被称为“按钮状态组”。按照例如正常状态→选择状态→激活状态的顺序设置按钮状态组。如上所述，根据 ODS 与其关联的按钮状态确定 DS 中的 ODS 的顺序。

图 19 所示为 DS 中的 ODS 顺序的示意图。在该图中，在第二级上显示 DS 中的以下三个按钮状态组：用于正常状态表示的按钮状态组（用于正常状态的 ODS）；用于选择状态表示的按钮状态组（用于选择状态的 ODS）；以及用于激活状态表示的按钮状态组（用于激活状态的 ODS）。在该图中，按照正常状态→选择状态→激活状态的顺序设置按钮状态组。确定这一顺序从而使得再现装置首先读取构成第一交互显示的交互合成，然后读取只在更新后显示的交互合成。

图 19 示出由按钮状态组参照的在第一级上的图形对象  $A_n$ 、 $B_n$ 、 $C_n$ 、 $D_n$ 、 $A_s$ 、 $B_s$ 、 $C_s$ 、 $D_s$ 、 $A_a$ 、 $B_a$ 、 $C_a$  和  $D_a$ 。如在  $A_n$ 、 $B_n$ 、 $C_n$  和  $D_n$  中的数字下标  $n$  表示处于正常状态下的各按钮。同样地，如在  $A_s$ 、 $B_s$ 、 $C_s$ 、 $D_s$  中的数字下标  $s$  表示处于选择状态下的各按钮，如在  $A_a$ 、 $B_a$ 、 $C_a$ 、和  $D_a$  中的数字下标  $a$  表示处于激活状态下的各按

钮。该图在第二级上示出按钮状态组，显示在第一级上的图形对象属于所述按钮状态组。在该图中，存在不止一个附有诸如“1”和“n”的数字下标的 ODS，用以读取“ODS 1”和“ODS n”。然而，应该注意的是包括在 N-ODS、S-ODS 和 A-ODS 中的每一个 ODS 1 是不同的。这对于具有相似参考数字的附图来说是一样的。

图 20 所示为在由图 19 所示的按钮状态组定义的交互显示上的按钮状态转换。

如在该图中所示，交互显示具有多个状态，包括“初始状态”、“通过第一个用户动作的显示更新”以及“通过第二个用户动作的显示更新”。图中的箭头表示触发状态转变的用户动作。四个按钮 A、B、C 和 D 各具有正常状态、选择状态以及激活状态。为了呈现初始交互显示，需要为显示准备好用于处于正常状态下的三个按钮和处于选择状态下的一个按钮的图形对象。

当没有指定缺省选择按钮时，没有确定首先选择按钮 A-D 中的哪一个。即使如此，一完成对用于每一个按钮的正常状态和选择状态的图形对象进行的解码就可以进行初始交互显示。鉴于这一观察，按照正常状态→选择状态→激活状态的顺序设置本实施例中的按钮状态组，如图 19 所示。通过这种设置，即使尚未读取用于激活状态的 ODS 并对其进行解码，也可以呈现初始交互显示。作为结果，缩短了开始读取 DS 和完成呈现初始交互显示之间的持续时间。

参考图 21，对如图 16、17 所示的 ODS 的顺序进行说明。图 21 所示为 DS 中的 ODS 的顺序。在该图中，用于正常状态的 ODS 包括 ODS 11-13、21-23、31-33 以及 41-43。用于选择状态的 ODS 包括 ODS 14-16、24-26、34-36 以及 44-46。用于激活状态的 ODS 包括 ODS 17-19、27-29、37-39 以及 47-49。ODS 11-13 用于图 17 所示的人物图像的动画显示。同样地，ODS 21-23、31-33 以及 41-43 用于其他人物图像的动画显示。这样，通过将由上述 ODS 组成的按钮状态组作为第一按

钮状态组放置在 DS 中，即使在完成 DS 的读取之前也准备好了用于显示的初始交互显示。这确保了没有延迟地显示由动画按钮组成的交互显示。

接下来，对于当在多个按钮状态组中存在对 ODS 顺序的多重参考的 ODS 顺序进行说明。术语多重参考表示通过两条或多条 ICS 中的 `normal_state_info`、`selected_state_info` 以及 `activated_state_info` 参考相同的 `object_id`。例如，利用所述多重参考，用于反映处于正常状态下的按钮的特定图形对象通常用来反映处于选择状态下的另一个按钮。也就是说，共用图形对象，从而可以减少 ODS 的数量。这里，引出了关于具有多重参考的 ODS 属于哪一个按钮状态组的问题。更具体地，当一个 ODS 与一个处于正常状态下的按钮以及另一个处于选择状态下的按钮有关时，需要确定 ODS 是属于用于正常状态的按钮状态组还是属于用于选择状态的按钮状态组。在这种情况下，将 ODS 放置在第一个出现的按钮状态组中。例如，如果按钮状态组按照正常状态→选择状态→激活状态的顺序出现，则将正常状态和选择状态组都参考的 ODS 放置在正常状态组中。此外，将选择状态和激活状态组都参考的 ODS 放置在选择状态组中。在此结束对多重参考的 ODS 顺序进行的说明。

现在，对用于选择状态的按钮状态组中的 ODS 的顺序进行说明。在用于选择状态的按钮状态组中，应该将哪一个 ODS 放置在开始处取决于是否确定缺省的选择按钮。当将 `default_selected_button_number` 字段设为非“0”的有效值时，指定缺省的选择按钮。在这种情况下，将与缺省的选择按钮有关的 ODS 放置在按钮状态组的开始处。

当将 ICS 中的 `default_selected_button_number` 字段设为值“00”时，没有指定缺省的选择按钮。例如，在将 DS 多路复用到其中的 AV 剪辑用作多个再现路径的合并点的情况下，将

default\_selected\_button\_number 字段设为值“00”。假设先前的再现路径与段 1、2 和 3 相对应，则用作合并点的 DS 用于表示与段 1、2 和 3 有关的按钮。在这种情况下，不能在 default\_selected\_button\_number 字段中指定特定的按钮。

理想地，当在段 1 的再现之后呈现交互显示时，应该选择与段 2 有关的按钮作为缺省。同样地，在段 2 的再现之后，应该选择与段 3 有关的按钮作为缺省，并且在段 3 的再现之后应该选择与段 4 有关的按钮。也就是说，当需要根据采用哪一个再现路径来选择不同的按钮作为缺省时，将 default\_selected\_button\_number 字段设为值“0”，以使其失效。在这种情况下，不需要将特定的按钮状态组 ODS 放置在开始处，因为没有选择特定的按钮作为缺省。

在此结束对 ODS 的顺序进行的说明。接下来，对在 AV 剪辑的再现时间轴上如何分配具有上述的 ICS 和 ODS 的 DS 进行说明。时期是再现时间轴上的一段时间间隔，在此期间存储器管理是连续的。由于每一个时期包括一个或多个 DS，所以如何在 AV 剪辑的再现时间轴上分配那些 DS 是重要的。这里提到的 AV 剪辑的再现时间轴是用于定义单独图像的解码时间和显示时间的的时间轴，其中所述单独图像构成多路复用在 AV 剪辑中的视频流。用 90 KHz 的时间准确度表示再现时间轴上的解码时间和显示时间。该 90 KHz 的时间准确度对应于 NTSC 信号、PAL 信号、Dolby AC-3 以及 MPEG 音频的帧频的公倍数。附加在 DS 中的 ICS 和 ODS 上的 DTS 和 PTS 指定用于实现同步控制的再现时间轴上的定时。也就是说，通过利用附加在 DS 中的 ICS 和 ODS 上的 DTS 和 PTS 进行同步控制来在再现时间轴上分配 DS。

首先，对利用 ODS 的 DTS 和 PTS 进行同步控制进行说明。

DTS 示出一个时间，具有 90 KHz 的时间准确度，在该时间，必须开始对 ODS 进行解码。PTS 示出完成解码的截止时间。

ODS 的解码不能立即完成，而要花费一些时间。为了明确显示对 ODS 进行解码的开始和结束时间，ODS 的 DTS 和 PTS 示出解码开始时间和解码的截止时间。

PTS 的值示出需要完成 ODS (j) 的解码以及在再现装置的对象缓冲器中需要使作为结果的未压缩图形对象有效的截止时间。

通过具有 90 KHz 的时间准确度的 DTS (DS<sub>n</sub> [ODS]) 来示出属于 DS 的任意 ODS (j) 的解码开始时间。这样，通过将可用于解码的最长时间加到 DTS (DS<sub>n</sub> [ODS]) 值，来确定 ODS (j) 的解码截止时间。

使 SIZE (DS<sub>n</sub> [ODS j]) 表示 ODS (j) 的大小，并且 Rd 表示 ODS 解码速率。然后解码所需的最长时间（单位为秒）是 SIZE (DS<sub>n</sub> [ODS j]) //Rd。

通过将这一最长时间转换为 90 KHz 的准确度并且将结果加到 ODS (j) 的 DTS，用 90 KHz 的准确度计算由 PTS 显示的解码截止时间。

属于 DS (n) 的 ODS (j) 的 PTS 可以由下列公式表示：

$$\text{PTS (DS [ODS j])} = \text{DTS (DS}_n \text{ [ODS j])} + 90000 \times (\text{SIZE (DS}_n \text{ [ODS j]) //Rd})$$

接下来，对 ICS 中的 PTS 的值进行说明。

将 ICS 中的 PTS 设为通过将 (1) 在 DS(n) 的初始显示所需要的所有 ODS 当中解码时间最迟的 ODS 的 PTS 值；与 (2) 清除图形平面所用去的时间；以及与 (3) 将已解码的图形对象转移到图形平面所用去的时间相加起来而获得的值。

当在 ICS 中指定 default\_selected\_button\_number 时，一完成 ODS 的解码就可以呈现初始交互显示，其中所述 ODS 与处于正常状态下



的每一个按钮以及与处于选择状态下的特定的缺省按钮有关。用于反映处于选择状态下的每一个按钮的 ODS 称作 S-ODS，并且解码开始时间是最早（即，在这种情况下，用于反映缺省选择按钮的 ODS）的 S-ODS 称作 S-ODSsfirst。指定该 S-ODSsfirst 的 PTS 值作为解码开始时间是最早的 ODS 的 PTS 值，并且用作 ICS 中的 PTS 的参考值。

当在 ICS 中没有指定 default\_selected\_button\_number 时，不知道将首先选择哪一个按钮。这样，直到准备好反映处于正常状态以及选择状态下的每一个按钮时初始交互显示才可以用于显示。在用于反映包括在初始交互显示的每一个按钮的选择状态的 S-ODS 当中，解码开始时间最迟的 ODS 称作 S-ODSslast。指定该 S-ODSslast 的 PTS 值作为解码开始时间最迟的 ODS 的 PTS 值，并且用作 ICS 中的 PTS 的参考值。

使  $PTS(DS_n [S-ODSsfirst])$  表示 S-ODSsfirst 的解码截止时间，然后将  $PTS(DS_n [ICS])$  设为通过 (2) 清除图形平面所用去的时间与 (3) 将已解码的图形对象反映到图形平面上所用去的时间，所获得的值。

使 video\_width 和 video\_height 表示图形平面的矩形区域的宽度和高度，在所述图形平面中反映图形对象。当图形平面的显示速率是 128 Mbps 时，通过  $8 \times video\_width \times video\_height // 128000000$  来获得清除图形平面所用去的时间。利用 90 KHz 的时间准确度，通过  $90000 \times (8 \times video\_width \times video\_height // 128000000)$  来表示上述清除图形平面所用去的时间 (2)。

使  $\Sigma SIZE(DS_n [ICS.BUTTON[i]])$  表示由包含在 ICS 中的每一条按钮信息指定的图形对象的全部尺寸。当图形平面的显示速率是 128 Mbps 时，通过  $\Sigma SIZE(DS_n [ICS.BUTTON[i]]) // 128000000$  来获得显示图形平面所用去的时间。利用 90 KHz 的时间准确度，通过  $90000 \times (\Sigma SIZE(DS_n [ICS.BUTTON[i]]) // 128000000)$  来表示上

述显示图形平面所用去的时间 (3)。

根据上述公式，PTS (DS<sub>n</sub> [ICS]) 由下列公式表示：

$$\begin{aligned} \text{PTS (DS}_n \text{ [ICS])} \geq & \text{PTS (DS}_n \text{ [S-ODSsfirst])} + \\ & 90000 \times (8 \times \text{video\_width} \times \text{video\_height} // 128000000) + \\ & 90000 \times (\sum \text{SIZE (DS}_n \text{ [ICS.BUTTON}[i])}) // 128000000 \end{aligned}$$

注意当 ICS 中的 default\_selected\_button\_number 是有效时，上述公式是可应用的。否则，需要满足下列公式。

$$\begin{aligned} \text{PTS (DS}_n \text{ [ICS])} \geq & \text{PTS (DS}_n \text{ [S-ODSslast])} + \\ & 90000 \times (8 \times \text{video\_width} \times \text{video\_height} // 128000000) + \\ & 90000 \times (\sum \text{SIZE (DS}_n \text{ [ICS.BUTTON}[i])}) // 128000000 \end{aligned}$$

通过上述方式设置 PTS 和 DTS，实现同步显示。图 22 示出这种控制的例子。在该例子中，与视频流中的图像 py1 的显示时间同步显示按钮。为此目的，需要将 ICS 中的 PTS 设为与图像数据 py1 的显示点相对应的值。注意设置 ICS 中的 DTS 以显示由 PTS 所示的时间之前的时间。

需要在由 ICS 中的 PTS 所示的时间中减去平面清除的持续时间 cd1 和对象转移的持续时间 td1 而计算的时间之前完成对构成 DS (n) 的初始交互显示的 ODS 进行的解码。这样，需要将解码时间最迟的 ODS 的 PTS 设为与在该图中用黑色星号所标记的时间点相对应的值。此外，对 ODS 进行解码要用去持续时间 dd1，从而需要将 ODS 的 DTS 设为比 PTS 早持续时间 dd1 的时间。

图 22 所示的例子是一种简化的情况：仅仅一个 ODS 用于与视频数据的重叠。在多个 ODS 用于呈现初始交互显示的情况下，需要如图 23 所示的那样设置 ICS 和 ODS 的 PTS 和 DTS。

图 23 所示为在由多个 ODS 构成初始交互显示以及指定了缺省的选择按钮的情况下的 DTS 和 PTS 设置。假设对 S-ODSsfirst 进行解码用去持续时间 dd1, S-ODSsfirst 是在呈现初始交互显示所需要的所有 ODS 当中解码时间是最早的 ODS。将该 S-ODSsfirst 的 PTS ( $DS_n [S-ODSsfirst]$ ) 设为与持续时间 dd1 的结束相对应的值。

此外, 为了呈现初始交互显示, 必须清除图形平面并且必须传输已解码的图形对象。这样, 必须将 ICS 的 PTS ( $DS_n [ICS]$ ) 设为与在最早时通过将 PTS ( $DS_n [S-ODSsfirst]$ ) 的值、平面清除的持续时间 ( $90000 \times (8 \times video\_width \times video\_height // 128000000)$ ), 以及已解码的对象传输的持续时间 ( $90000 \times (\sum SIZE (DS_n [ICS.BUTTON[i]]) // 128000000)$ ) 相加起来而计算的时间相对应的值。

图 24 所示为在由多个 ODS 构成初始交互显示以及没有指定缺省的选择按钮的情况下的 DTS 和 PTS 设置。这里, 将 PTS ( $DS_n [S-ODSslast]$ ) 设为与对 S-ODSslast 进行解码所用去的持续时间 dd2 的结束相对应的值, S-ODSslast 是在呈现初始交互显示所需要的所有 ODS 当中解码时间是最迟的 ODS。

此外, 为了呈现初始交互显示, 必须清除屏幕并且必须传输已解码的图形对象。这样, 需要将 ICS 的 PTS ( $DS_n [ICS]$ ) 设为与在最早时通过将 PTS ( $DS_n [S-ODSslast]$ ) 的值、屏幕清除的持续时间 ( $90000 \times (8 \times video\_width \times video\_height // 128000000)$ ), 以及已解码的对象传输的持续时间 ( $90000 \times (\sum SIZE (DS_n [ICS.BUTTON[i]]) // 128000000)$ ) 相加起来而计算的时间相对应的值。这里结束关于由 ICS 定义的同步控制的描述。

在 DVD 的情况下, 在 VOB 的时间内交互控制是有效的, 其对应于视频流的 GOP。然而, 在 BD-ROM 的情况下, 可以利用包括在时期中的 ICS 的 PTS 和 DTS 来任意设置交互控制的有效周期。也就

是说，BD-ROM 的交互控制与 GOP 无关。

注意通过 ICS 的 PTS 的同步控制不仅包括在再现时间轴上的特定点显示按钮，而且还包括在在再现时间轴上的特定持续时间内能得到弹出式菜单用于显示。弹出式菜单是在按遥控器 400 的菜单键时显示的菜单。在 AV 剪辑中特定图像的显示时间能得到弹出式菜单用于显示。这种控制也包括在由 ICS 的 PTS 定义的同步控制中。类似于用于按钮显示的 ODS，对用于弹出式菜单显示的 ODDS 进行解码，并且在图形平面上显示已解码的对象。除非完成对图形平面的显示，否则不能对来自用户的菜单调用作出反应。为了允许弹出式菜单的同步显示，ICS 的 PTS 示出可得到弹出显示的时间。

在此结束对根据本发明的第一实施例的记录介质进行的说明。下面说明根据本发明的第一实施例的再现装置。图 25 示出再现装置的内部结构。在该内部结构的基础上在工业上制造根据本发明的再现装置。再现装置大概是由三个部分组成：系统 LSI、驱动装置、以及微型计算机系统。可以通过将这些部分安装在机柜上或装置的衬底上来制造该再现装置。系统 LSI 是集成电路，包括用于实现再现装置功能的各种处理单元。以上述方式制造的再现装置包括 BD 驱动器 1、轨道缓冲器 2、PID 滤波器 3、传输缓冲器 4a、4b 和 4c、外围电路 4d、视频解码器 5、视频平面 6、音频解码器 7、图形平面 8、CLUT 单元 9、加法器 10、图形解码器 12、编码数据缓冲器 13、外围电路 13 a、流图形处理器 14、对象缓冲器 15、合成缓冲器 16、图形控制器 17、UO 控制器 18、播放器寄存器组 19 以及控制器 20。

BD 驱动器 1 执行 BD-ROM 的装载、读取和弹出，从而访问 BD-ROM。

轨道缓冲器 2 是 FIFO（先入先出）存储器。相应地，按照与 TS 数据包到达相同的顺序从轨道缓冲器 2 中除去从 BD-ROM 读取的 TS 数据包。

PID 滤波器 3 对从轨道缓冲器 2 输出的 TS 数据包进行滤波。更具体地，PID 滤波器 3 仅仅将具有预定 PID 的 TS 数据包传给传输缓冲器 4a、4b 和 4c。在 PID 滤波器 3 内没有所需的缓冲。相应地，无延迟地将进入 PID 滤波器 3 的 TS 数据包写到传输缓冲器 4a、4b 和 4c。

传输缓冲器 4a、4b 和 4c 是 FIFO 存储器，用于存储从 PID 滤波器 3 输出的 TS 数据包。

外围电路 4d 具有布线逻辑，用于将从传输缓冲器 4a、4b 和 4c 读取的 TS 数据包转变为功能段。然后将功能段存储在编码数据缓冲器 13。

视频解码器 5 对从 PID 滤波器 3 输出的 TS 数据包进行解码以获得未压缩的图像，并且将获得的图像写到视频平面 6 上。

视频平面 6 是用于视频数据的平面存储器。

音频解码器 7 对从 PID 滤波器 3 输出的 TS 数据包进行解码，并且输出未压缩的音频数据。

图形平面 8 是具有一屏存储区的平面存储器，并且能够存储一屏未压缩的图形。

CLUT 单元 9 根据在 PDS 中定义的 Y、Cr、和 Cb 值来转换图形平面 8 上的未压缩图形的指针颜色。

加法器 10 使由 CLUT 单元 9 转换的未压缩图形乘以在 PDS 中定义的 T 值（透明度）。然后加法器 10 为在作为结果的未压缩图形和视频平面 6 上的未压缩图像数据中的相应像素执行加法，以输出合成图像。

图形解码器 12 对图像流进行解码以获得未压缩图形，并且将未压缩图形作为图形对象提供给图形平面 8。作为对图形流进行解码的结果，子标题和菜单出现在屏幕上。该图形解码器 12 包括编码数据缓冲器 13、外围电路 13 a、流图形处理器 14、对象缓冲器 15、合成缓冲器 16 以及图形控制器 17。

编码数据缓冲器 13 用于存储功能段连同它们的 DTS 和 PTS。通过从存储在传输缓冲器 4a、4b 和 4c 中的每一个 TS 数据包除去 TS 数据包的头部和 PES 数据包的头部以及顺次设置有效负载来获得功能段。与 PES 数据包一致地存储包含在所除去的 TS 数据包的头部和 PES 数据包的头部中的 DTS 和 PTS。

外围电路 13a 具有布线逻辑，该布线逻辑用于将数据从编码数据缓冲器 13 传输到流图形处理器 14，以及将数据从编码数据缓冲器 13 传输到合成缓冲器 16。更具体地，当当前时间达到 ODS 的 DTS 时，外围电路 13a 将 ODS 从编码数据缓冲器 13 传输到流图形处理器 14。此外，当当前时间达到由 ICS 或 PDS 的 DTS 所示的时间时，外围电路 13a 将 ICS 或 PDS 从编码数据缓冲器 13 传输到合成缓冲器 16。

流图形处理器 14 对 ODS 进行解码以获得具有指针颜色的未压缩图形，并且将未压缩图形作为图形对象传输到对象缓冲器 15。由流图形处理器 14 进行的解码在由与 ODS 有关的 DTS 所示的时间开始，并且在由也与 ODS 有关的 PTS 所示的解码截止时间结束。上述图形对象解码的解码速率  $R_d$  等于流图形处理器 14 的输出速率。

对象缓冲器 15 存储由流图形处理器 14 解码的图形对象。

合成缓冲器 16 用于存储 ICS 和 PDS。

图形控制器 17 对存储在合成缓冲器 16 中的 ICS 进行解码，并且根据解码结果在由附加在 ICS 上的 PTS 所指定的时间执行控制。

UO 控制器 18 检测用户在遥控器和再现装置的前面板上进行的操作，并且将显示所检测到的用户操作（以下称为 UO（用户操作））的信息输出到控制器 20。

播放器寄存器组 19 是一组设置在控制器 20 中的寄存器，并且包括 32 个播放器状态寄存器和 32 个通用寄存器。以下示出播放器状态寄存器各值（PRS）的含义。“PSR (x)”表示第 x 个播放器状态寄存器的值。

- PSR (0): 保留
- PSR (1): 将要被解码的音频流数
- PSR (2): 将要被解码的子标题图形流数
- PSR (3): 由用户设置的角度数
- PSR (4): 当前再现目标的标题数
- PSR (5): 当前再现目标的段数
- PSR (6): 当前再现目标的 PL 数
- PSR (7): 当前再现目标的播放项数
- PSR (8): 显示当前再现点的时间信息
- PSR (9): 导航定时器的计数
- PSR (10): 当前处于选择状态下的按钮数
- PSR (11) - (12): 保留
- PSR (13): 由用户设置的父级
- PSR (14): 针对视频的再现装置设置
- PSR (15): 针对音频的再现装置设置
- PSR (16): 用于音频的语言码
- PSR (17): 用于子标题的语言码
- PSR (18): 用于菜单说明的语言码
- PSR (19) - (63): 保留

在每一次显示属于 AV 剪辑的图像时更新 PSR (8)。即，当再现装置显示新图像时，更新 PSR (8) 从而保持与新显示图像的显示开始时间（即，显示时间）相对应的值。参考 PSR (8)，当前再现点是已知的。

控制器 20 通过与图形解码器 12 的双向数据交换执行完整控制。从控制器 20 输出到图形解码器 12 的数据是由 UO 控制器 18 接收的

UO。从图形解码器 12 输出到控制器 20 的数据是包含在 ICS 中的按钮命令。

具有上述结构的再现装置的部件执行流水线中的处理。

图 26 所示为 ODS 解码的流程图。在该图中，第四级示出记录在 BD-ROM 上的 DS。第三级示出通过编码数据缓冲器 13 的 ICS、PDS 以及 ODS 的读取持续时间。第二级示出通过流图形处理器 14 的 ODS 的解码持续时间。第一级示出通过图形控制器 17 进行处理的持续时间。在该图中示出每一个 ODS 的解码开始时间，作为 DTS11、DTS12、和 DTS13。需要在由相应的 DTS 所指定的解码开始时间前将每一个 ODS 读到编码数据缓冲器 13。因此，最迟就在到编码数据缓冲器 13 的 ODS1 的解码持续时间  $dp1$  之前，完成 ODS1 的读取。同样地，最迟就在 ODS2 的解码持续时间  $dp2$  之前，完成 ODS(n)的读取。

在该图中示出每一个 ODS 的解码截止时间，作为 PTS11、PTS12 和 PTS13。在 PTS11 所示的时间前完成通过流图形处理器 14 的 ODS1 的解码，并且在 PTS12 所示的时间前完成 ODS (n) 的解码。如上所述，在由 ODS 的相应 DTS 所示的时间前将每一个 ODS 读到编码数据缓冲器 13，以及对所读取的 ODS 进行解码并在由 ODS 的相应 PTS 所示的时间前装载到对象缓冲器 15。通过单个流图形处理器 14 流水线地执行上述处理。

在指定缺省选择按钮的情况下，一完成对用于正常状态的所有 ODS 和用于选择状态的第一个 ODS 进行的解码，就可在对象缓冲器 15 上得到呈现初始交互显示所需的所有图形对象。在该图中，在 PTS13 所示的时间，可得到呈现初始交互显示所需的所有图形对象。

在第一级，图形控制器 17 需要持续时间  $cd1$  以清除图形平面 8，并且为将对象缓冲器 15 上的图形提供给图形平面 8 需要持续时间  $td1$ 。由 ICS 中的 `button_horizontal_position` 和 `button_vertical_position` 字段指定图形平面 8 中用于显示的位置。也就是说，在通过将平面清



除持续时间  $cd1$  以及已解码的图形对象的显示持续时间  $td1$  加到 ODS 的  $PTS13$  值上而计算的时间的结尾在图形平面 8 上获得构成交互显示的未压缩图形。CLUT 单元 9 对未压缩图形进行颜色转换，并且加法器 10 将图形覆盖在用视频平面 6 保存的未压缩图像上。作为结果，产生合成图像。

通过上述设置，不用等待完成 ODS 的解码就呈现出初始交互显示，其中所述 ODS 属于用于选择状态和激活状态的按钮状态组。这样，与只在对 DS 中的所有 ODS 进行解码之后才呈现初始交互显示的情况相比，所述呈现提早了持续时间  $hy1$ ，如图所示。

注意在该图中，存在不止一个附有诸如“1”和“ $n$ ”的数字下标的 ODS，用以读取“ODS 1”和“ODS  $n$ ”。然而，包括在 N-ODS、S-ODS 和 A-ODS 中的每一个 ODS 1 是不同的。这对于具有相似参考数字的附图来说是一样的。

在图形解码器 12 中，当图形控制器 17 清除和显示图形平面 8 时，流图形处理器 14 连续解码（显示在第二级上的 ODS ( $n$ )、ODS 1、ODS ( $n$ ) 的解码持续时间）。由于即使在图形控制器 17 清除和显示图形平面 8 期间也连续对剩余的 ODS 进行解码，与不是这种情况相比，可较早地完成对剩余 ODS 进行的解码。这样，用户一操作就可以立即执行使用那些剩余 ODS 的交互显示的更新。作为这种流水线处理的结果，可以迅速地呈现初始交互显示和更新显示。

图 26 中的例子涉及指定缺省选择按钮的情况。图 27 所示为在没有指定缺省选择按钮的情况下，再现装置执行流水线处理的流程图。在这种情况下，一完成对用于正常状态和用于选择状态的所有 ODS 进行的解码，就可得到呈现初始交互显示所需的所有图形对象，并且将作为结果的对象装载到图形平面 8。利用上述设置，不用等待完成对用于激活状态 ODS 进行的解码就呈现出初始交互显示。这样，与只在对 DS 中的所有 ODS 进行解码之后才呈现交互显示的情况相比，

所述呈现提早了持续时间  $hy_2$ ，如图所示。

在此结束对再现装置的内部结构进行的说明。接下来，对控制器 20 和图形解码器 12 的实现进行说明。可以通过通用 CPU 执行如图 28 和 29 所示的步骤程序来实现控制器 20。以下，参考图 28 和 29 对控制器 20 所执行的操作进行说明。

图 28 是用于执行 LinkPL 功能的控制器 20 所执行的操作的流程图。当对包括 LinkPL 功能的命令进行解码时，控制器 20 就如图 20 所示工作。

在该流程图中，将要处理的播放项和存取单元被分别表示为  $PI(y)$  和存取单元  $(v)$ 。如该流程图所示，再现装置读取由 LinkPL 的参数所指定的 PL (.mpls) (步骤 S1)，并且指定当前 PL 中的第一个 PI 作为  $PI(y)$  (步骤 S2)。然后，再现装置读取由  $PI(y)$  中的 `clip_information_file_name` 字段所指定的剪辑信息 (步骤 S3)。

一读取剪辑信息，再现装置就使用包含在剪辑信息中的 `EP_map` 将  $PI(y)$  中的 `In_time` 字段的值转换为地址 (步骤 S4)。由所得到的地址指定的存取单元被指定为存取单元  $(v)$  (步骤 S5)。再现装置使用包含在剪辑信息中的 `EP_map` 还将 `Out_time` 字段的值转换为地址 (步骤 S6)。由所得到的地址指定的存取单元被指定为存取单元  $(w)$  (步骤 S7)。

一旦指定存取单元  $(v)$  和  $(w)$ ，再现装置就指示 BD 驱动器读取从  $(v)$  到  $(w)$  的存取单元 (步骤 S8)，并且还指示视频解码器 5、音频解码器 7、图形解码器 12 对驻留在对应于  $PI(y)$  中的 `In_time` 字段到 `Out_time` 字段的地址中的数据进行解码 (步骤 S9)。

在步骤 S11 中，对该流程图进行结束判断从而察看  $PI(y)$  是否等于  $PI(z)$ 。如果步骤 S11 的结果为“是”，则终止该流程图的处理。如果不是，将下一个播放项指定为  $PI(y)$  (步骤 S12)，并且处理返回到步骤 S3。重复步骤 S1- S10 直到步骤 S11 的结果为“是”。

在步骤 S10 中，在读取存取单元的同时，将功能段装载到编码数据缓冲器 13。

图 29 是为装载功能段执行的操作的流程图。在该流程图中，段(K)是表示利用存取单元读取的段(ICS、ODS 或 PDS)的变量，并且忽略标志表示是忽略段(K)还是装载段(K)。在该流程图中，在将忽略标志初始化为 0(步骤 S20)之后，为段(K)执行步骤 S21 到步骤 S24 以及步骤 S27 到步骤 S35 的循环(步骤 S25 和 S26)。

在步骤 S21 中，再现装置判断段(K)是否是 ICS。如果段(K)是 ICS，则处理转到步骤 S27 和 S28 中的判断。

在步骤 S27 中，判断 ICS 中的 `segment_type` 字段是否被设为表示获得点 DS 的值。如果段(K)属于获得点 DS，则处理转到步骤 S28。在另一方面，如果段(K)属于时期开始 DS 或正常情况 DS 的任意一个，则处理转到步骤 S33。

在步骤 S28 中，判断紧前一个 DS 是否存储在编码数据缓冲器 13 中。如果步骤 S27 的结果为“是”，则执行步骤 S28。如果执行跳过操作，则紧前一个 DS 没有存储在编码数据缓冲器 13 中。在这种情况下，需要从获得点 DS 开始显示，从而处理转到步骤 S30(步骤 S28: 否)。

另一方面，在紧前一个 DS 存储在编码数据缓冲器 13 中(步骤 S28: 是)的情况下，将忽略标志设为“1”(步骤 S29)，并且处理转到步骤 S31。

在步骤 S31 中，判断 `command_update_flag` 字段是否被设为“1”。如果设为“1”(步骤 S31: 是)，则当忽略其它数据时将按钮信息的按钮命令装载到编码数据缓冲器 13 中(步骤 S32)。如果 `command_update_flag` 字段被设为“0”，则处理转到步骤 S22。因此，忽略表示获得点的 ICS(步骤 S24)。

通过将忽略标志设为“1”，步骤 S22 的结果为“否”，从而忽略属

于获得点 DS 的所有功能段。

在步骤 S33 中，判断 ICS 中的 `segment_type` 字段是否表示正常情况 DS。如果段 (K) 属于时期开始 DS，则在步骤 S30 中将忽略标志设为“0”。

如果将忽略标志设为“0”（步骤 S22：是），则将段 (K) 装载到编码数据缓冲器 13 中（步骤 S23）。

另一方面，如果段 (K) 属于正常情况 DS，则处理转到步骤 S34。类似于步骤 S28，在步骤 S34 中，判断紧前一个 DS 是否存储在编码数据缓冲器 13 中。如果存储了前一个 DS，则将忽略标志设为“0”（步骤 S30）。否则，将忽略标志设为“1”（步骤 S35）。通过如上设置忽略标志，如果前一个 DS 没有存储在再现装置的编码数据缓冲器 13 中，则忽略属于正常情况 DS 的功能段。

在每一条按钮信息 (1)、(2) 和 (3) 中的 `auto_action_flag` 字段被设为“1”的情况下，三个按钮进入到激活状态而不是选择状态，并执行按钮信息中的按钮命令 (LinkPL(PL#21)、LinkPL(PL#22)、以及 LinkPL(PL#23))。例如，假设按钮命令与其联系的 PL#21、PL#22 以及 PL#23 是棒球运到员击球和投球的情景，那么一旦数字输入相应的运到员号码就再现那些情景。由于通过输入运到员号码进行选择，这对于用户来说是容易识别的，所以进一步提高了用户的可操作性。

现在，对在图 30 所示的例子中如何读取 DS 进行说明。在该例子中，利用视频数据多路复用三个 DS (DS1、DS10 和 DS20)。在 DS1 中，其是三个 DS 中的第一个，`segment_type` 字段表示时期开始，将 `command_update_flag` 字段设为“0”，并且包括按钮命令 LinkPL(PL#5)。

DS10 是对 DS1 的复制。在 DS10 中，将 `segment_type` 字段设为表示获得点 DS 的值，将 `command_update_flag` 字段设为“0”，并且

包括按钮命令 LinkPL(PL#5)。

从 DS1 继承 DS20, 并且它的 segment\_type 字段值表示获得点 DS。DS20 与 DS1 的不同之处在于按钮命令 LinkPL(PL#10), 从而对按钮命令不同的信号将 command\_update\_flag 设为“1”。

假设跳过操作执行到 AV 剪辑中的图像 pt10, 其中利用视频数据多路复用上述三个 DS。在这种情况下, 对最靠近跳终点的 DS10 进行处理, 如图 29 所示。具体地, 步骤 S27 中的判断结果是: segment\_type 是获得点 DS, 但前一个 DS 没有存储在编码数据缓冲器 13 中。相应地, 将忽略标志设为 0。作为结果, 将 DS10 装载到编码数据缓冲器 13 中, 如图 31 所示。另一方面, 假设跳过操作执行到晚于 DS10 的点上的图像 (图 30 中的箭头 hst1)。在这种情况下, 将 DS10 之后的 DS20 装载到编码数据缓冲器 13 中 (图 31 中的箭头 hst2)。

图 33 示出当执行正常再现时如何装载 DS1、DS10 和 DS20, 如图 32 所示。在三个 DS 中, 将 segment\_type 字段表示时期开始 DS 的 DS1 装载到编码数据缓冲器 13 中 (步骤 S23)。然而, 对于 segment\_type 字段表示获得点 DS 的 DS10, 将忽略标志设为“1” (步骤 S29)。因此, 不将 DS10 的功能段装载到编码数据缓冲器 13 中而是将其忽略 (步骤 S24)。至于 DS20, segment\_type 字段表示获得点 DS, 但将 command\_update\_flag 设为“1”。这样, 步骤 S31 的结果为“是”。因此, 将按钮命令装载到编码数据缓冲器 13 中, 由此取代当前存储在 DS 的 ICS 中的按钮命令。然而, 将忽略标志设为“1”, 由此不装载除按钮命令之外的数据, 而是将其忽略。

通过上述操作, 在显示 DS20 之前, 和与 DS10 相同的图形有关的按钮命令 LinkPL(#5) 已经被 LinkPL(#10) 取代了。通过这种取代, 可以执行和与再现相同的按钮变化有关的按钮命令这样的控制。

接下来, 对通过图形控制器 17 执行的处理进行说明。图 34 示出由图形控制器 17 执行的处理的主程序。流程图显示重复以下三个操

作：时间戳同步（步骤 S35）；动画显示（步骤 S36）；以及 UO 处理（步骤 S37）。

图 35 是根据时间戳执行同步控制的流程图。在该流程图中，在步骤 S43-S47 中判断是否已经发生了特殊事件。每一个事件的发生调用相应子程序，其中在执行完预定的步骤之后所述子程序返回到主程序。

在步骤 S43 中，判断当前再现点是否已经达到 ODS 的 PTS 所示的时间。如果判断是肯定的，则执行到图形平面 8 的显示（步骤 S51），然后处理返回到主程序。

在步骤 S45 中，判断当前再现点是否已经达到 ICS 的 PTS。如果判断是肯定的，则使图形平面 8 开始输出所存储的内容。输出到进行颜色转换的 CLUT 单元 9。在颜色转换之后，交互合成与存储在视频平面 6 中的内容重叠，由此呈现初始交互显示（步骤 S52）。然后，将变量  $q$  设为值“1”（步骤 S53），并且处理返回到主程序。注意变量  $q$  是全局变量（即，在多个流程图中始终有效的变量），并且将在后面对其含义进行说明。

在步骤 S46 和 S47 中，判断当前再现点是否已经达到由 ICS 中的时间信息所指定的时间。

在步骤 S46 中，判断当前再现点是否已经达到由 `selection_time_out_pts` 所示的时间。如果判断结果是肯定的，则激活由 `default_activated_button_number` 所指定的按钮，然后处理返回到主程序（步骤 S54）。

在步骤 S47 中，判断当前再现点是否已经达到由 `composition_time_out_pts` 所指定的时间。如果判断结果是肯定的，则在清除图形平面之后处理返回到主程序（步骤 S55）。在此结束对根据时间戳的同步进行的说明。在同步中，步骤 S51 和 S54 调用子程序。下面参考附图 36 对步骤 S51 的子程序进行说明。

图 36 是图形平面 8 的显示的流程图。在呈现初始交互显示之前需要完成到图形平面 8 的显示。在该图中，ODS<sub>x</sub> 表示具有与当前再现点相对应的 PTS 的 ODS，由 PRS (10) 显示。在步骤 S61-S63 中，判断该 ODS 是否是初始交互显示所需要的 ODS 中的最后一个。如果该 ODS 是最后一个，则执行步骤 S64-S72。

在步骤 S61 中，判断 `default_selected_button_number` 字段是否具有有效值。如果该字段是有效的，则在步骤 S63 中判断 ODS<sub>x</sub> 是否是 S-ODS<sub>first</sub>。如果步骤 S63 的结果为“否”，则终止该流程图的处理并且返回到主程序。

如果步骤 S61 的结果为“否”，则在步骤 S62 中判断 ODS<sub>x</sub> 是否是 S-ODS<sub>last</sub>。如果步骤 S62 的结果为“否”，则终止该流程图的处理并且返回到主程序。

在步骤 S64 中，判断 ICS 中的 `segment_type` 字段是否表示时期开始 DS。如果表示时期开始 DS，则执行步骤 S65 以清除图形平面 8，并且执行步骤 S66-S72。清除图形平面 8 所用去的这段时间是图 23 和 24 所示的持续时间 `cd1`。如果 `segment_type` 字段不表示时期开始 DS，则跳过步骤 S65 转到步骤 S66-S72。

步骤 S66-S72 形成一个循环，其中依次处理 ICS 中的每条按钮信息（步骤 S66 和 S67）。在该循环中，要处理的当前按钮信息由 `button_info (p)` 表示。在步骤 S68 中，判断 `button_info (p)` 是否对应于由 `default_selected_button_number` 指定的缺省选择按钮。

如果 `button_info (p)` 不对应于缺省选择按钮，则在存储在对象缓冲器 15 中的图形对象当中，将在 `button_info (p)` 的 `start_objcdt_id_normal` 字段中指定的图形对象指定为图形对象 (p)（步骤 S69）。

另一方面，如果 `button_info (p)` 对应于缺省选择按钮，则在存储在对象缓冲器 15 中的图形对象当中，将由 `button_info (p)` 的

start\_objecdt\_id\_selected 字段所指定的图形对象指定为图形对象 (p) (步骤 S70)。然后, 将按钮 (p) 指定为当前按钮 (步骤 S71)。当前按钮指的是在当前呈现的交互显示中具有选择状态的按钮。再现装置将当前按钮的 ID 存储在 PRS (10) 中。

一旦通过步骤 S69 和 S70 指定图形对象 (p), 则将图形对象 (p) 显示到图形平面 8 上由 button\_horizontal\_position 和 button\_vertical\_position 字段指定的位置上 (步骤 S72)。针对 ICS 中的每条按钮信息重复上述处理。在这种情况下, 在与每一个按钮状态有关的多个图形对象当中, 针对图形平面 8 上的每一个按钮显示第一个图形对象。用于显示存储在对象缓冲器 15 中的所有图形对象所用去的这段时间是图 23 和 24 所示的持续时间 td1。在此结束对步骤 S51 进行的说明。接下来, 参考图 37, 对在步骤 S54 中调用的子程序进行说明。

图 37 是用于自动激活缺省选择按钮的处理的流程图。首先, 判断 default\_activated\_button\_number 字段是否被设为“00”或“FF”(步骤 S91)。如果该字段被设为“00”, 则处理返回到主程序而不执行任何操作。另一方面, 如果 default\_activated\_button\_number 被设为“FF”, 则将当前按钮指定为按钮 (u) (步骤 S95)。

如果该字段的值既不是“00”也不是“FF”, 则将由 default\_activated\_button\_number 字段指定的按钮指定为按钮 (u) (步骤 S92) 并激活按钮 (u) (步骤 S93)。通过显示来自 button\_info (u) 中的 start\_object\_id\_activated 字段所指定的图形对象到 end\_object\_id\_activated 字段所指定的图形对象的一序列图形对象来执行按钮状态转换。button\_horizontal\_position 和 button\_vertical\_position 字段表示图形平面 8 上的显示位置。然后, 执行与按钮 (u) 有关的按钮命令 (步骤 S94), 并且处理返回到主程序。

通过上述操作, 处于选择状态下的按钮在预定的持续时间结束时



进入到激活状态。在此结束对图 37 所示的流程图进行的说明。

接下来，对菜单中的按钮动画显示进行说明（步骤 S36）。图 38 是用于动画显示的处理的流程图。

通过将 在 每 条 `button_info` 的 `start_object_id_normal` 和 `start_object_id_selected` 字段中指定的图形对象显示到图形平面 8 上，来呈现初始显示。为了以动画显示按钮，对于步骤 S35-S37 的循环的每一次重复，通过用按钮的任意一帧（第  $q$  个图形对象）重写每一个按钮来更新图形平面 8。更具体地讲，通过依次显示由每条 `button_info` 中的 `normal_state_info` 和 `selected_state_info` 指定的图形对象来更新图形平面 8。这里，变量  $q$  用来指在每条按钮信息中的 `normal_state_info` 和 `selected_state_info` 字段中指定的单独图形对象。

现在，参考图 38，对用于进行动画显示的处理进行说明。在该流程图中，在以相同数量的帧显示每个按钮的动画的前提下，对显示处理进行说明。这是为了使说明简明。采用不同数量的帧显示按钮需要更复杂的处理。此外，还是为了使说明简明，假设 `repeat_normal_flag` 和 `repeat_selected_flag` 字段都被设为要求连续重复动画的值。

在步骤 S80 中，判断是否呈现初始显示。如果没有呈现初始显示，则处理返回到主程序而不执行任何操作。如果已经呈现初始显示，则执行步骤 S81-S90。步骤 S81-S90 形成一个循环，其中针对包含在 ICS 中的每条 `button_info` 重复步骤 S83-S87（步骤 S81 和 S82）。

在步骤 S83 中，判断 `button_info (p)` 是否对应于当前按钮。

如果 `button_info (p)` 不对应于当前按钮，则将变量  $q$  加在 `button_info(p)` 中的 `start_objecdt_id_normal` 的值上以获得  $ID(q)$ （步骤 S84）。

如果 `button_info(p)` 对应于当前按钮，则将变量  $q$  加在 `button_info (p)` 中的 `start_objecdt_id_selected` 字段的值上以获得  $ID (q)$ （步骤 S85）。

在存储在对象缓冲器 15 中的图形对象当中，将图形对象 (q) 显示到图形平面 8 上由 button\_info (p) 中的 button\_horizontal\_position 和 button\_vertical\_position 字段所指定的位置上(步骤 S87)。针对 ICS 中的每条 button\_info 重复上述操作 (步骤 S81 和 S82)。

通过上述循环，在用于显示处于选择状态下的当前按钮和处于正常状态下的其它按钮的图形对象当中，针对每一个按钮将第 q 个图形对象显示在图形平面 8 上。

在步骤 S88 中，判断 end\_object\_id\_normal 字段的值是否已经达到由 start\_objecdt\_id\_normal 字段+ q 所获得的值 (步骤 S89)。如果判断的结果是肯定的，则将变量 q 初始化为“0”并且处理返回到主程序 (步骤 S89)。如果判断的结果是否定的，则将变量 q 加 1，并且处理返回到主程序 (步骤 S90)。

通过步骤 S80-S90，在每次执行步骤 S35-S37 时使用新的图形对象来更新交互显示上的每个按钮的显示。通过多次重复步骤 S35-S37，以动画显示每个按钮。当以动画显示按钮时，图形控制器 17 判断关于以由 animation\_frame\_rate\_code 显示的速率呈现图形对象。当可应用时也在其它的流程图中进行这种判断。在此结束对用于动画显示的处理进行的说明。接下来，参考图 39，对主程序的步骤 S37 所示的 UO 处理进行说明。

图 39 是 UO 处理的流程图。在该流程图中，在步骤 S100-S103 判断是否已经发生特定事件。每个事件的发生都调用相应的子程序，所述子程序在执行完预定步骤后返回到主程序。在步骤 S100 中，判断 UO\_mask\_table 字段是否被设为“1”。如果该字段被设为“1”，则处理返回到主程序而不执行任何操作。

在步骤 S101 中，判断是否按了上移键、下移键、左移键和右移键中的任何一个键。在按下这些键的任何一个时，另外一个键被相应地指定为当前按钮 (步骤 S104)，随后判断新指定的当前按钮的

auto\_action\_flag 是否被设为“01”（步骤 S108）。如果 auto\_action\_flag 没有被设为“01”，则处理返回到主程序。另一方面，如果 auto\_action\_flag 被设为“01”，则处理转到步骤 S105。

在步骤 S102 中，判断是否按了激活键。在按下激活键时，在步骤 S105 中激活当前按钮。通过将当前按钮的 start\_object\_id\_activated 字段所指定的图形对象到 end\_object\_id\_activated 字段所指定的图形对象的一序列图形对象显示在图形平面 8 上来执行这种按钮状态转换。由当前按钮的 button\_horizontal\_position 和 button\_vertical\_position 字段指定图形平面 8 中的显示位置。通过这种设置，实现这样的控制：用作按钮的人物看上去响应用户的操作而移动。在按钮状态转变之后，执行与当前按钮有关的按钮命令（步骤 S106）。

在步骤 S103 中，判断是否进行了数字输入。如果进行了数字输入（步骤 S107），则根据输入的数字执行操作（步骤 S107），并且处理返回到主程序。在图 39 中，步骤 S104 和 S107 各调用子程序。图 40 和 41 示出子程序。现在，对那些流程图进行说明。

图 40 是当前按钮变化处理的流程图。在以相同数量的帧显示每个按钮动画的前提下，对该流程图进行说明。这是为了使说明简明。采用不同数量的帧显示按钮需要更复杂的处理。首先，确定当前按钮的 neighbor\_info 中的 upper\_button\_number、lower\_button\_number、left\_button\_number 以及 right\_button\_number 中的哪一个与被按的键相对应（步骤 S110）。

使按钮 (i) 表示当前按钮，而按钮 (j) 表示下一个被指定为当前按钮的按钮（步骤 S111）。在步骤 S112 中，判断在步骤 S111 中被指定的按钮 (j) 是否等于按钮 (i)。如果是，则处理返回到主程序而不执行任何操作。如果不是，则执行步骤 S113-S120。在步骤 S113-S120 中，使按钮 (j) 进入到选择状态，并且使按钮 (i) 回到正常状态。以动画显示每一个按钮状态。为此，首先将变量  $r$  初始化为“0”。变

量  $r$  用来指动画的单独帧。在步骤 S113-S1119 中，将与处于正常状态下的按钮 (i) 有关的图形对象中的第  $r$  个和与处于选择状态下的按钮 (j) 有关的图形对象中的第  $r$  个反复显示到图形平面上。

具体地，通过将变量  $r$  加到 `button_info(i)` 的 `start_objecdt_id_normal` 字段的值上来计算 ID ( $r$ ) (步骤 S114)。在以上述方式计算 ID ( $r$ ) 之后，在存储在对象缓冲器 15 中的图形对象当中，将具有 ID ( $r$ ) 的图形对象显示到图形平面 8 上由 `button_info (i)` 的 `button_horizontal_position` 和 `button_vertical_position` 字段所示的位置上 (步骤 S115)。

接下来，通过将变量  $r$  加到 `button_info (j)` 的 `start_objecdt_id_selected` 字段的值上来计算 ID ( $r$ ) (步骤 S116)。在以上述方式获得 ID ( $r$ ) 之后，在存储在对象缓冲器 15 中的图形对象当中，将具有 ID ( $r$ ) 的图形对象显示到图形平面 8 上由 `button_info (j)` 的 `button_horizontal_position` 和 `button_vertical_position` 字段所示的位置上 (步骤 S117)。

在步骤 S118 中，判断通过将变量  $r$  加到 `start_object_id_normal` 字段的值上而获得的 ID 是否等于 `end_object_id_normal` 字段的值。如果这两个值不相等，则增加变量  $r$  (步骤 S120)，并且处理返回到步骤 S114。重复步骤 S114-S120 直到步骤 S118 的判断结果为“是”。通过重复上述步骤，实现按钮状态转变从而用作按钮的人物响应用户的操作而移动。当步骤 S118 的判断结果为“是”时，现在将按钮 (j) 指定为当前按钮 (步骤 S119)，并且处理返回到主程序。

图 41 是数字输入处理的流程图。首先，判断是否存在一条 `button_number` 与输入数字匹配的 `button_info (j)` (步骤 S121)。接下来，判断 `button_info (j)` 中的 `numerically_seletable_flag` 字段是否被设为“1” (步骤 S122)。如果步骤 S121 和 S122 的结果都为“是”，则判断 `button_info(j)` 中的 `auto_action_flag` 字段是否被设为“01” (步

骤 S123)。

如果 `auto_action_flag` 字段的值不是“01”，则从与按钮 (j) 有关的 `start_object_id_selected` 字段所指定的图形对象到 `end_object_id_selected` 字段所指定的图形对象的一序列图形对象依次显示在由与按钮 (j) 有关的 `button_horizontal_position` 和 `button_vertical_position` 字段所指定的位置上(步骤 S124)。作为结果，使按钮 (j) 进入到选择状态。然后，将按钮 (j) 指定为当前按钮(步骤 S125)，并且处理返回到主程序。

另一方面，如果 `auto_action_flag` 字段的值是“01”，则在步骤 S126 中激活当前按钮。通过将 从与当前按钮有关的 `start_object_id_activated` 字段所指定的图形对象到 `end_object_id_activated` 字段所指定的图形对象的一序列图形对象依次显示在图形平面 8 上由与当前按钮有关的 `button_horizontal_position` 和 `button_vertical_position` 字段所指定的位置上来执行这种按钮状态转换。在步骤 S127 中，执行与按钮 (j) 有关的按钮命令，并且处理返回到主程序。

如果步骤 S121-S123 中任意一个的结果为“否”，则处理返回到主程序。

在此结束图形控制器 17 为同步显示执行的 处理。为了呈现交互显示，例如由用户操作引起的弹出显示，流图形处理器 14 和图形控制器 17 执行以下操作，其与针对同步显示而执行的步骤相似。通过这样做，图形平面 8 得到所需要的图形对象。在得到图形对象之后，直到当前再现点达到由附在 ICS 上的 PTS 所指定的时间才执行操作。如果 UO 控制器 18 在当前再现点已经过了由 PTS 指定的时间之后接收菜单调用的用户操作，则将存储在图形平面 8 中的图形对象输出到 CLUT 单元 9，其中将所输出的图形对象与视频平面重叠。与 UO 同步执行的输出导致响应于菜单调用而呈现弹出显示。

如上所述,根据本发明实施例,将包括 ICS 和 ODS 的每一个时期多路复用在 AV 剪辑中。采用这种 AV 剪辑,容易地对交互控制进行描述从而再现装置与视频特定帧的显示同步执行特定操作。简而言之,本发明的 AV 剪辑适用于描述紧密限制到视频帧的显示时间的交互控制。此外,由于将时期多路复用在 AV 剪辑中,所以即使当要求再现数百个视频段时,也没有必要将所有相关的时期存到存储器中。利用视频包从 BD-ROM 中依此读取时期。这样,在当前再现的视频段的整个持续时间内,与该视频段有关的 ICS 都驻留在存储器中。一完成视频段,就从存储器中除去已再现的时期,然后将对应于下一个视频段的 ICS 装载到存储器中。由于将时期多路复用在 AV 剪辑中,所以即使时期的数量超过好几百也可以使所需要的存储器的大小保持在最小值。

## 第二实施例

本发明的第二实施例涉及对在将按钮状态改变为选择状态和激活状态时再现的卡嗒声作出的改进。例如,假设用户与出现在电影中用作按钮的人物进行交互,如图 16 和 17 所示。在这种情况下,如果作为响应用户操作的卡嗒声来再现相应人物的声音,则用户可以本能地识别用户正在操作哪一个按钮。通过这种设置,用户可以更容易地识别按钮操作。这里,出现了如何同时再现卡嗒声和主声音的问题。这里使用的主声音是指电影的声音,例如人物的演讲以及电影中的背景音乐。将承载主声音的音频流与视频流和图形流多路复用到 AV 剪辑中。音频解码器 7 对音频流进行解码。在再现卡嗒声时,需要对主音频进行消音。为此,需要暂停音频解码器 7 的操作。然而,暂停音频解码器 7 有产生噪声的危险。

为了避免由于暂停音频解码器 7 而产生噪声,再现装置具有如图 42 所示的内部结构。该图所示的结构与图 25 所示的结构相似,只是

增加了预载存储器 21 和混合器 22。

预载存储器 21 预先存储未压缩的 LPCM 数据以便作为卡嗒声而对其进行再现。

混合器 22 将存储在预载存储器 21 中的未压缩的 LPCM 数据与音频解码器 7 的输出进行混合。以包括在图形解码器 12 中的图形控制器 17 所指示的速率对数据进行混合（参见图 25）。由于根据混合参数再现卡嗒声，所以不必暂停音频解码器 7 的输出。

在此结束对根据第二实施例的再现装置结构进行的说明。

为了同时再现主声音和卡嗒声，需要将记录在 BD-ROM 上的解压缩 LPCM 数据预先装载到预载存储器 21。然而，不幸的是，解压缩 LPCM 数据相对较大。例如，不到十秒的 16-bit/48 KHz LPCM 音频数据具有一兆字节的大小。

为了满足减小预载存储器 21 大小的要求，根据本发明的 ICS 具有如图 43 所示的数据结构。图 43 示出用于以上述方式再现卡嗒声的 ICS 的数据结构。该图中的数据结构与图 11 所示的数据结构的不同之处在于 button\_info。在每条 button\_info 中，selected\_state\_info（）和 activated\_state\_info（）各额外具有“audio\_specification\_info”字段和“audio\_reproduction\_control\_info”字段。

audio\_specification\_info 字段表示音频数据的文件名或 ID，当将与 button\_info 有关的按钮设为不同的状态时再现装置读取所述音频数据并将其再现为卡嗒声。将要装载到预载存储器 21 的卡嗒声数据由 button\_info 的 selected\_state\_info（）和 activated\_state\_info（）指定。将装载到预载存储器的卡嗒声数据提供给混合器 22。

audio\_reproduction\_control\_info 包括多个混合参数。每一个混合参数表示每一个音频分量与主声音混合的速率。每一个混合参数具有 0—1.0 范围内的值。卡嗒声数据在再现之前乘以由混合参数所示的值。在音频数据包括 R 和 L 分量的情况下，audio\_reproduction\_control\_info

字段表示 R 和 L 分量的分开的混合参数，从而指示混合器 22 以由那些参数所示的混合速率对音频数据进行混合。

通过提供 `audio_reproduction_control_info`，当 R 分量表示按钮 B 的卡嗒声时，未压缩 LPCM 数据的 L 分量表示按钮 A 的卡嗒声。以这种方式，可以共同存储两个不同按钮的两种不同的卡嗒声作为一个未压缩 LPCM 数据。

除了上述共同存储的卡嗒声之外，`button_info (1)` 还只包括为 R 分量的输出而设定的 `audio_reproduction_control_info`，而 `button_info (2)` 还只包括为 L 分量的输出而设定的 `audio_reproduction_control_info`。通过这种设置，在按钮 A 的状态转变到选择状态的同时，根据 `button_info (1)` 的 `audio_reproduction_control_info` 来再现未压缩 LPCM 数据的 L 分量，从而再现按钮 A 的卡嗒声。

同样地，在按钮 B 的状态转变到选择状态的同时，根据 `button_info (2)` 的 `audio_reproduction_control_info` 来再现未压缩 LPCM 数据的 R 分量，从而再现按钮 B 的卡嗒声。

参考图 44 和 45，对使用上述 ICS 和再现装置再现卡嗒声的特殊例子进行说明。本例涉及图 44A 和 44B 所示的状态控制信息。图 44A 所示的状态控制信息包括 `button_info (1)` 和 `button_info (2)`。如图中箭头 `sy1` 和 `sy2` 所示，`button_info (1)` 和 `button_info (2)` 指定同一个卡嗒声数据，其是立体声。`button_info (1)` 的 `audio_reproduction_control_info` 包括 L 分量的混合参数，而 `button_info (2)` 的 `audio_reproduction_control_info` 包括 R 分量的混合参数。

图 44B 示出包括状态控制信息的 ICS 的读取过程。在 ICS 之前，将卡嗒声数据装载到预载存储器 21。

图 45A 和 45B 示出根据读到预载存储器 21 的 ICS 的卡嗒声数据的再现控制。在与 `button_info (1)` 有关的按钮 A 处于选择状态的情



况下，图形解码器 12 控制音频解码器 7，从而根据 `button_info` (1) 的 `audio_reproduction_control_info` 再现卡嗒声数据。作为结果，再现立体卡嗒声数据的 L 分量。另一方面，在与 `button_info` (2) 有关的按钮 B 处于选择状态的情况下，图形解码器 12 控制音频解码器 7，从而根据 `button_info` (2) 的 `audio_reproduction_control_info` 来再现卡嗒声数据。作为结果，再现立体卡嗒声数据的 R 分量。

通过上述控制，当选择相应按钮时，以立体声组成的一个卡嗒声数据通过再现 L 分量用作按钮 A 的卡嗒声并通过再现 R 分量用作按钮 B 的卡嗒声。

在此结束该特殊例子，其中共同存储多个卡嗒声作为一个未压缩 LPCM 数据。或者，可以设置本发明的 `button_info` 从而以声音来自不同方向的方式产生按钮操作的卡嗒声。下面参考图 45 对这种控制的特殊例子进行说明。图 45C 示出与三个水平排列的按钮（按钮 A、按钮 B、和按钮 C）有关的三条 `button_info` 的例子。与按钮 A 有关的 `button_info` 处在左侧，具有用于 L 分量的混合参数 1.0。与按钮 B 有关的 `button_info` 处在中间，具有用于 L 和 R 分量的混合参数 0.5。与按钮 C 有关的 `button_info` 处在右侧，具有用于 R 分量的混合参数 1.0。利用上述混合参数，当选择左按钮 A 时，从左侧的扬声器输出卡嗒声。当选择右按钮 C 时，从右侧的扬声器输出卡嗒声。当选择中间按钮 B 时，从两个扬声器同等地输出卡嗒声。如上所述，可以设置 `button_info` 从而根据按钮在屏幕上的位置从不同的方向输出卡嗒声。来自对应于所按按钮位置的方向的卡嗒声增加了按钮操作的真实性。

如上所述，根据本实施例，将用于多个按钮中的每一个按钮的卡嗒声结合为一个立体卡嗒声数据。利用 `audio_specification_info` 和 `audio_reproduction_control_info`，来再现同一个卡嗒声数据作为用于不同按钮的不同卡嗒声。通过这种结合，减少卡嗒声数据量，由此可以减小用于装载卡嗒声数据的预载存储器 21 的容量。

尽管以上给出的特殊例子是以立体卡嗒声数据为基础的，但是卡嗒声数据也可以是未压缩 5.2-信道音频数据。图 44C 示出与图 45C 所示相似的例子。然而，在图 44C 的例子中，针对 5.2-信道音频数据设置混合变量。这样，除了 L 和 R 分量之外，5.2-信道音频数据还包括中间、左后、以及右后分量。在交互显示上按钮 A、B 和 C 在对角上排列。在该例子中，与按钮 A 有关的 button\_info 具有用于 L 分量的混合参数 1.0。与按钮 C 有关的 button\_info 具有用于右后分量的混合变量参数 1.0。与按钮 B 有关的 button\_info 具有分别用于 L、R、中间、左后和右后分量的混合参数 0.1、0.1、0.4、0.2、和 0.2。通过这种设置，当选择左按钮 A 时，从右侧输出卡嗒声。当选择按钮 B 时，从左侧输出卡嗒声。当选择中间按钮 B 时，从所有方向输出卡嗒声。如上所述，根据所按按钮的位置改变卡嗒声来自的方向。这种设置增加了按钮操作的真实性（注意在上述例子中，或者可以将与按钮 B 有关的 button\_info 设为具有用于中间分量的参数 1.0 和用于每一个其它分量的参数 1.0）。

此外，除了音频解码器 7 之外，还可以提供另外一个音频解码器用于卡嗒声数据。在这种情况下，预载存储器 21 预先存储压缩的音频数据。响应按钮状态转换，用于卡嗒声数据的音频解码器从预载存储器 21 提取压缩的音频数据并对所提取的数据进行解码。提供用于卡嗒声数据的音频解码器允许预载存储器 21 存储压缩的音频数据，从而可以减小预载存储器 21 的容量。

### 第三实施例

本发明的第三实施例涉及 BD-ROM 的制造过程。图 46 是根据本发明的 BD-ROM 制造过程的流程图。

制造过程包括记录视频、声音等的材料制造步骤（步骤 S201），使用授权装置产生应用格式的授权步骤（步骤 S202），以及产生

BD-ROM 的原始主盘和执行冲压和压焊以完成 BD-ROM 的压盘步骤（步骤 S203）。

BD-ROM 授权步骤包括以下步骤 S204-S209。

首先，在步骤 S204 中，将视频材料、音频材料以及辅助图形材料分别编码到视频流、音频流、以及图形流中。接下来，在步骤 S205 中，执行图形流的验证测试。如在第一实施例中所述，除了用于显示按钮的图形数据之外，图形流还包括按钮的状态控制信息。这使得可以独立验证图形流。如果检测到有任何错误（步骤 S206：否），则对图形流进行适当的修改，所述图形流仍然没有与其它流进行多路复用（步骤 S207），并且再次执行图形流的验证测试。

如果通过图形流的验证测试没有检测到错误（步骤 S206：是），则通过对材料进行编码而在步骤 S208 中获得的视频、音频以及图形流被交错复用转换为一个数字流。在随后的步骤 S209 中，根据用于 BD-ROM 的情节产生必要的信息，并且改编情节和数字流以遵循 BD-ROM 格式。

根据本实施例，将定义按钮状态转换的 ICS 结合为具有图形数据的图形流。这样，不必等待对视频流进行编码或完成流的多路复用。而是，图形流一产生就准备好验证测试以查看当再现进行时按钮状态如何改变。由于在授权的较早阶段可以执行按钮状态转换的验证测试，所以避免了正好在发货之前检测到错误的可能性，这种可能性迫使开发商仓促地进行改正。图形流的验证测试独自为将移动复杂的动画按钮并入到电影中提供了更好的环境。

#### 补充注释

本发明的应用不限于上述的特定例子。利用以下修改（A）到（O）中的任何一个可以实施本发明。对上述实施例及其修改进行扩展和归纳的说明的本申请的每一个权利要求的发明包括以下内容。扩展和归

纳的程度反映了在提交本申请时相关技术领域的技术状态。然而，在每一个权利要求中所述的发明关注本发明相关问题的解决方法。这样，每一个发明的范围不超出本领域技术人员公认用于解决问题的方法的范围。因此，所附权利要求书中的每一项所述的发明实质上与以上的详细说明相一致。

(A) 上述实施例对 BD-ROM 用作记录介质的情况进行了说明。然而，本发明的主要特征在于记录在记录介质上的图形流，其不依赖于 BD-ROM 的物理特性。因此，本发明可应用于能够记录动态情景和图形流的任何记录介质。这种记录介质的例子包括：光盘，例如 DVD-ROM、DVD-RAM、DVD-RW、DVD-R、DVD+RW、DVD+R、CD-R 或 CD-RW；磁光盘，例如 PD 或 MO；半导体存储卡，例如压缩快闪卡、智慧 (SmartMedia) 卡、记忆棒 (Memory Stick) 卡、多媒体卡或 PCMCIA 卡；磁盘，例如软盘、超级磁盘、Zip 盘、或 Klik!；可移动硬盘驱动器，例如 ORB、Jaz、SparQ、SyJet、EZFley 或微型硬盘；以及非移动硬盘驱动器。

(B) 上述实施例对在 BD-ROM 上再现装置对 AV 剪辑进行解码并且将所解码的 AV 剪辑输出到电视机的情况进行了说明。作为可选择的方法，再现装置可以仅装配有 BD 驱动器，而剩余的构造元件在电视机中提供。在这种情况下，可以将再现装置和电视机合并到用 IEEE 1394 连接器连接的家庭网络中。此外，上述实施例对再现装置连接到电视机的情况进行了说明，但是再现装置也可以与显示设备结合。此外，再现装置可以只包括与处理的基本部分有关的元件。作为发明在本说明书中对那些再现装置都进行了说明。因此，与其模式无关，在第一、第二、或第三实施例中所述的根据再现装置的内部结构制造再现装置的行为被认为是实施本发明的行为。此外，收费散发(即

销售)或不收费散发(即作为礼物)、出租、以及引入所述再现装置的任何一种行为都是实施本发明的行为。同样地,使用店面展示、商品目录、或小册子来提供所述再现装置的散发或租借的行为是实施本发明的行为。

(C) 使用流程图所示程序的信息处理实际上是使用硬件资源来实现的。因此,描述流程图所示的操作过程的程序本身就是发明。上述实施例对将程序合并到再现装置的情况进行了说明,但是可以使用程序而与再现装置无关。执行程序的行为包括(1)制造行为;(2)收费散发或不收费散发行为;(3)租借行为;(4)引入行为;(5)通过双向电子通讯网络提供给公众的行为;以及(6)使用店面展示、商品目录、或小册子来提供散发或租借的行为。

(D) 在每一个流程图中以时间顺序执行的步骤的时间因素被认为是本发明的必要因素。由于情况如此,所以认为这些流程图所示的过程公开了再现方法。如果通过以时间顺序执行各步骤来执行每一个流程图所示的处理从而取得预期的目的和效果,则这就是实施本发明的记录方法的行为。

(E) 当将 AV 剪辑记录在 BD-ROM 上时,可以将扩展头部加到 AV 剪辑中的每一个 TS 数据包中。扩展头部被称为 TP\_extra\_header,包括 arrival\_time\_stamp 和 copy\_permission\_indicator,并且具有 4 字节的数据长度。以 32 个数据包为单元对具有 TP\_extra\_header 的 TS 数据包(以下为“EX TS 数据包”)进行分组,并且将每一组写到三个扇区。由 32 个 EX TS 数据包组成的一组具有 6144 个字节( $=32 \times 192$ ),这等价于三个扇区的大小,是 6144 个字节( $=2048 \times 3$ )。包含在三个扇区中的 32 个 EX TS 数据包被称为对准单元。

在采用 IEEE 1394 连接器连接的家庭网络中，再现装置 200 以如下方式传送对准单元。再现装置从对准单元的 32 个 EX TS 数据包中的每一个中除去 TP\_extra\_header，根据 DTCP 规范对每个 TS 数据包的主体进行加密，并且输出被加密的 TS 数据包。当输出 TS 数据包时，再现装置在相邻的 TS 数据包之间插入同步数据包。根据由 TP\_extra\_header 的 arrival\_time\_stamp 所示的时间确定插入同步数据包的点。再现装置 200 输出 DTCP\_descriptor 以及 TS 数据包。DTCP\_descriptor 对应于 TP\_extra\_header 中的 copy\_permission\_indicator。通过提供表示“拷贝禁止”的 DTCP\_descriptor，当在采用 IEEE 1394 连接器连接的家庭网络中使用 TS 数据包时，可以防止将 TS 数据包记录到其它设备上。

(F) 上述实施例对将 BD-ROM 格式的 AV 剪辑用作数字流的情况进行了说明，但是还可以采用 DVD-Video 格式或 DVD-Video 记录格式的 VOB（视频对象）实现本发明。VOB 是按照 ISO/IEC 13818-1 标准并通过多路复用视频流和音频流而获得的程序流。并且，AV 剪辑中的视频流可以是 MPEG4 视频流或 WMV 视频流。此外，AV 剪辑中的音频流可以是线性 PCM 音频流、Dolby AC-3 音频流以及 MP3 音频流或 MPEG-AAC 音频流。

(G) 可以利用对通过模拟广播的模拟图像信号广播进行编码来获得上述实施例所述的视频编辑。

或者，可以对记录在录像带上的模拟/数字图像信号进行编码以获得内容。此外，可以对由摄影机直接捕获到的模拟/数字图像信号进行编码以获得内容。由分布服务器分配的数字工作也是可应用的。

(H) ICS 可以定义在暂停时间执行的操作。由在第一实施例中所

述的 `composition_time_out_pts` 指定 ICS 的暂停。图 47 示出根据本发明的一个修改的 ICS。该图中的 ICS 新提供有 `Still/Pause_information` 字段。`Still/Pause_information` 字段表示是否使再现装置的操作进入“静止”或“中止”。这里所参考的再现装置的操作包括通过视频解码器 5、流图形处理器 14 和图形解码器 12 的解码操作，以及通过图形控制器 17 和控制器 20 的引导操作。这里使用的术语“静止”指的是暂停解码操作和引导操作，而术语“中止”指的是在继续引导操作时暂停解码操作。在静止的情况下，暂停引导操作，从而最后一个再现图像仍然作为静止图像而显示，并且按钮状态转换是不可能的。

另一方面，在中止的情况下，继续引导操作，从而允许用户改变按钮状态。通过在 ICS 中提供 `Still/Pause_information`，可以在授权时定义暂停所执行的控制。

(I) 在第二实施例中，`button_info` 定义用于每个按钮的卡嗒声。另外，可以由遥控器的每个键的 ICS 定义卡嗒声。图 48 所示为定义用于遥控器的每个键的卡嗒声的 ICS 的图。

`upper_audio` 字段定义在按下上移键时参考的音频指定信息和再现控制信息。

`lower_audio` 字段定义在按下下移键时参考的音频指定信息和再现控制信息。

`left_audio` 字段定义在按下左移键时参考的音频指定信息和再现控制信息。

`right_audio` 字段定义在按下右移键时参考的音频指定信息和再现控制信息。

`activated_audio` 字段定义在按下激活键时参考的音频指定信息和再现控制信息。在按下遥控器 400 的一个键时，预载存储器 21 和混合器 22 根据与所按键有关的音频指定信息和再现控制信息进行操

作，从而再现相应的卡嗒声。

(J) 在上述实施例中所述的图形对象是行程长度编码的光栅数据。行程长度编码用于图形对象的压缩/编码，因为行程长度编码适用于子标题的压缩和解压缩。子标题的特性在于同一个像素值的连续长度在水平方向上是相对长的。因此，通过使用行程长度编码执行压缩，可以获得高的压缩率。另外，行程长度编码减小用于解压缩的负载，因此适用于通过软件实现解码。然而，对图形对象使用行程长度编码对于本发明来说不是必须的。例如，图形对象可以是 PNG 数据。还有，图形对象可以是矢量数据而不是光栅数据。此外，图形对象可以是透明图案。

## 工业实用性

根据本发明的记录介质和再现装置实现对电影的交互控制，由此提高了电影的价值。以这种附加值分销的电影使电影市场和消费电器市场受到鼓舞。因此，根据本发明的记录介质和再现装置非常适合于电影和消费电器工业。



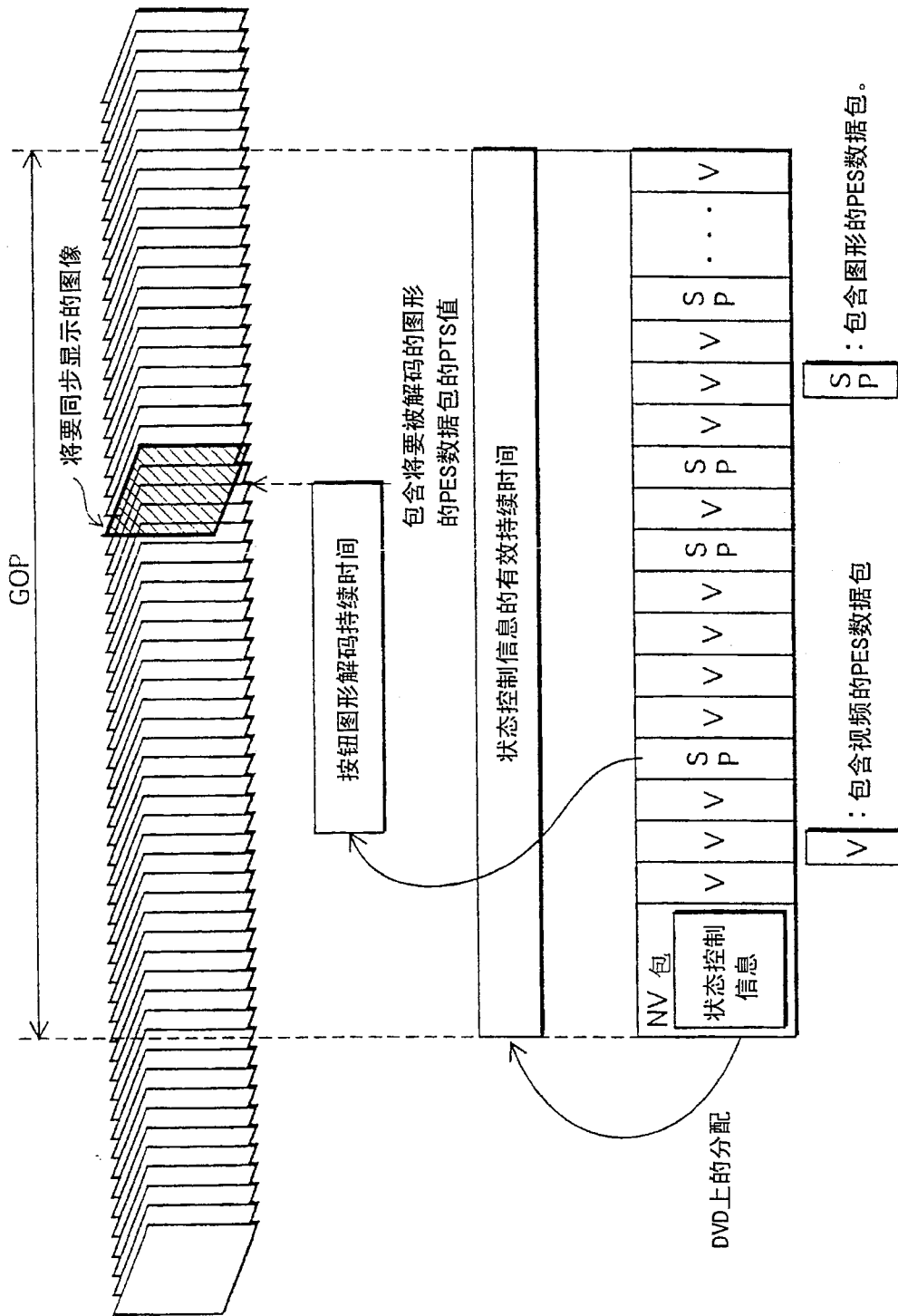


图1

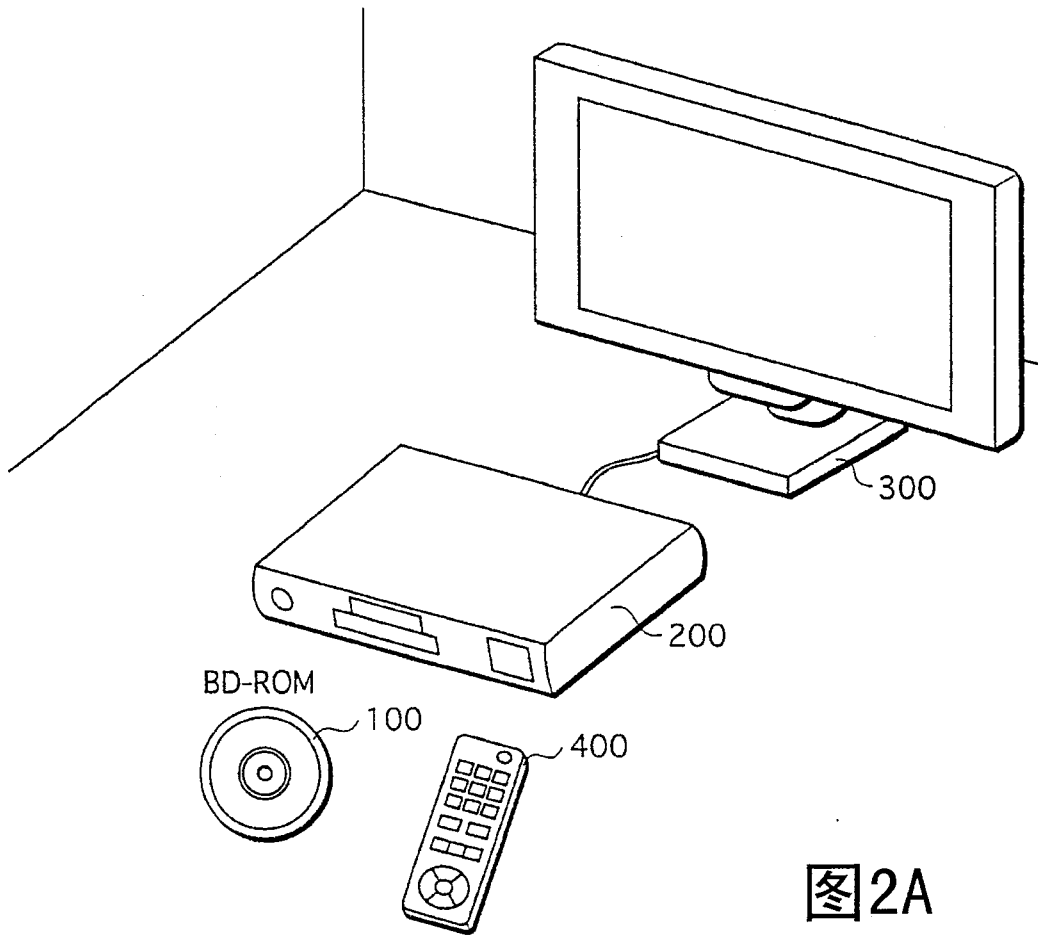


图2A

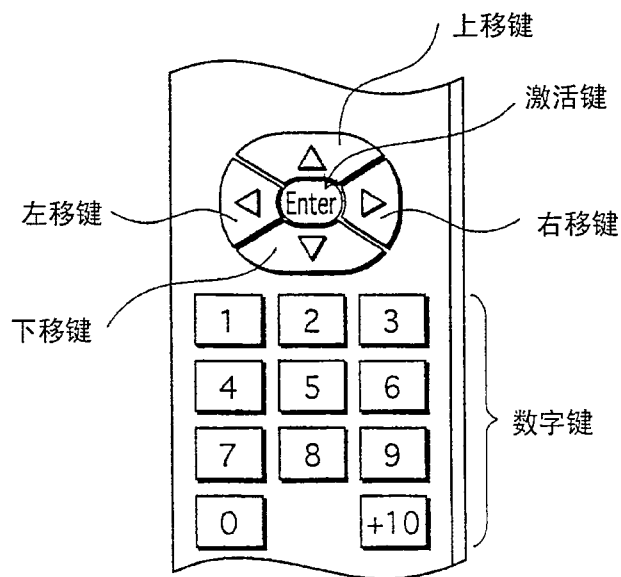
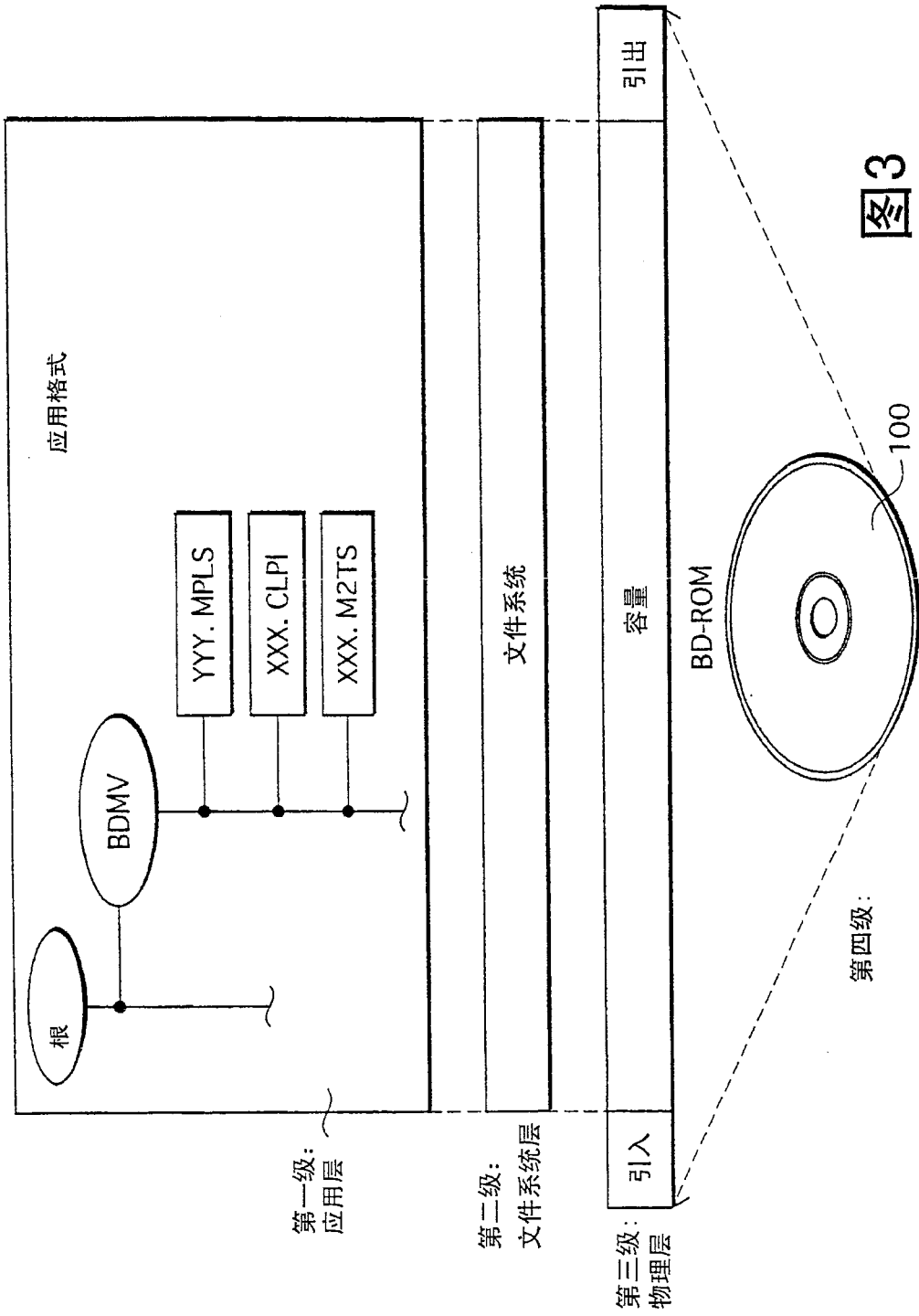


图2B



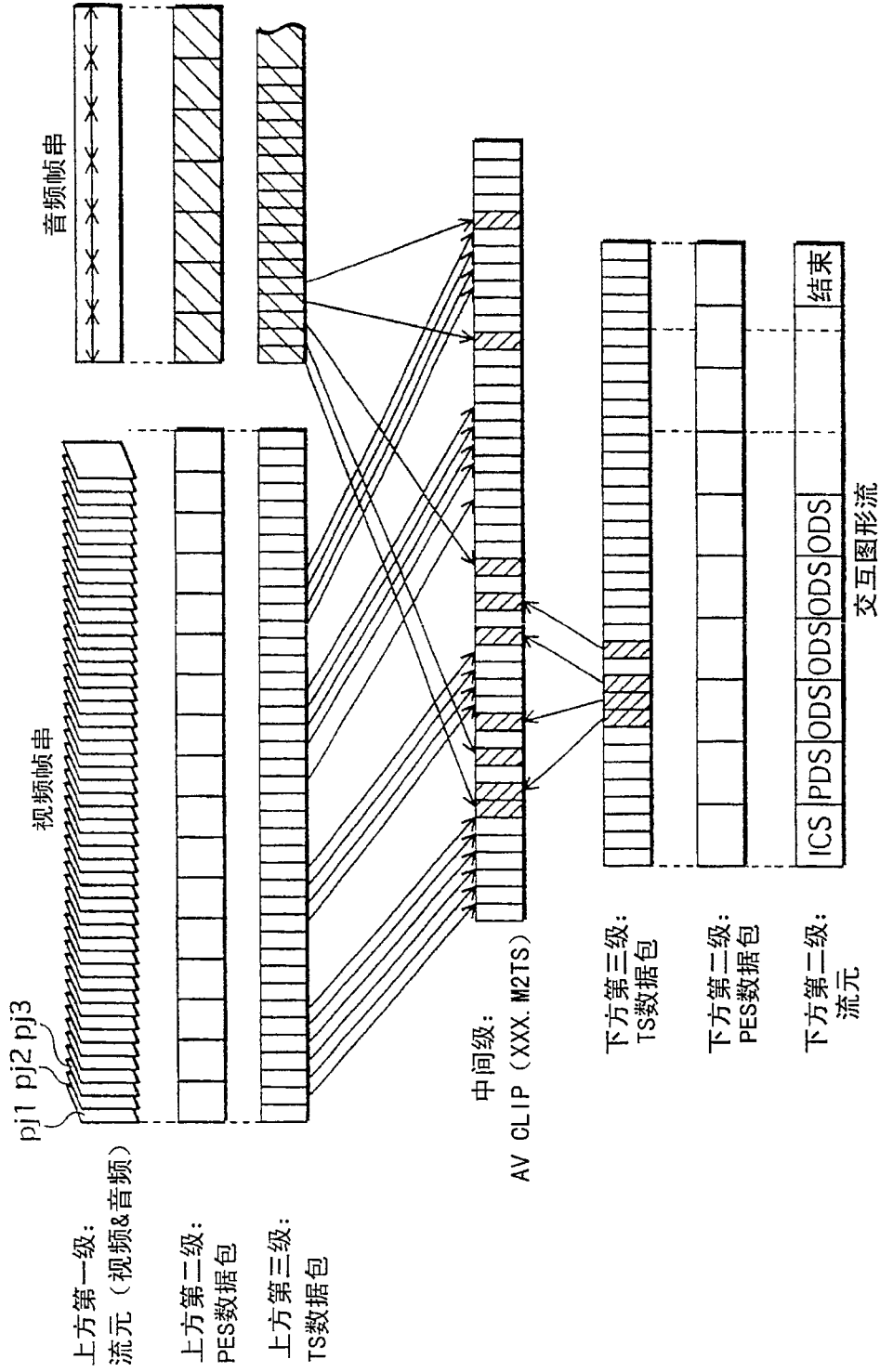


图4

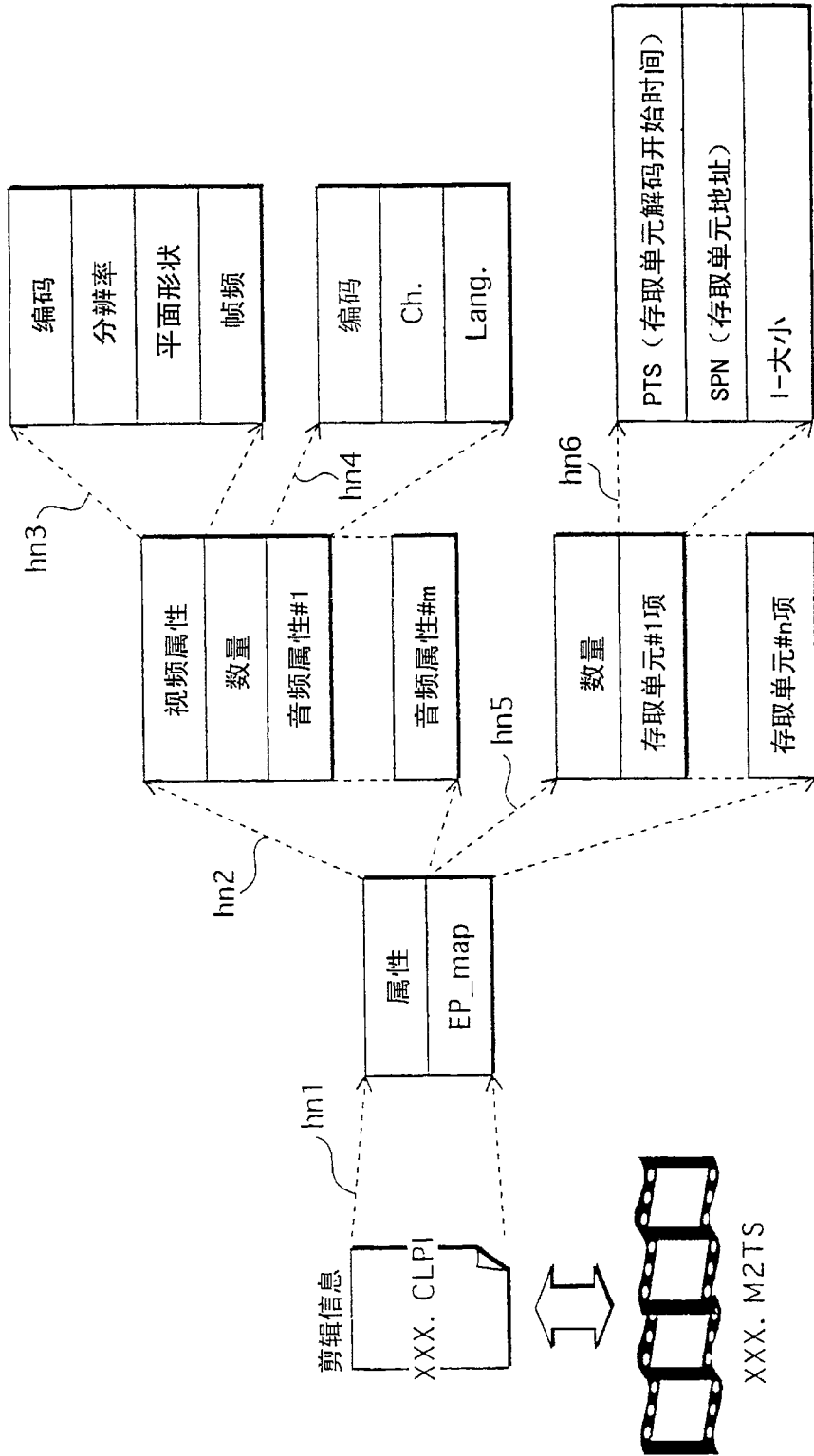


图5

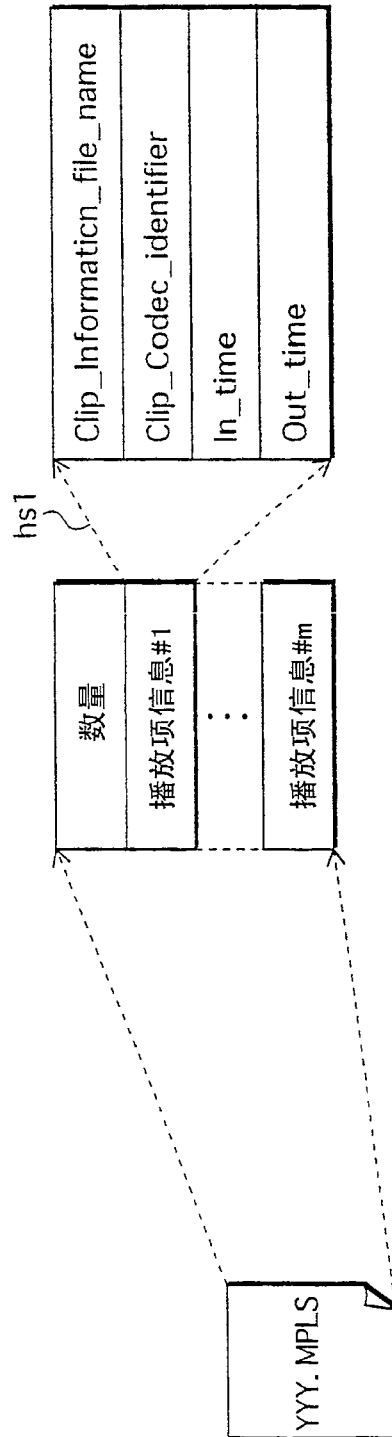


图6

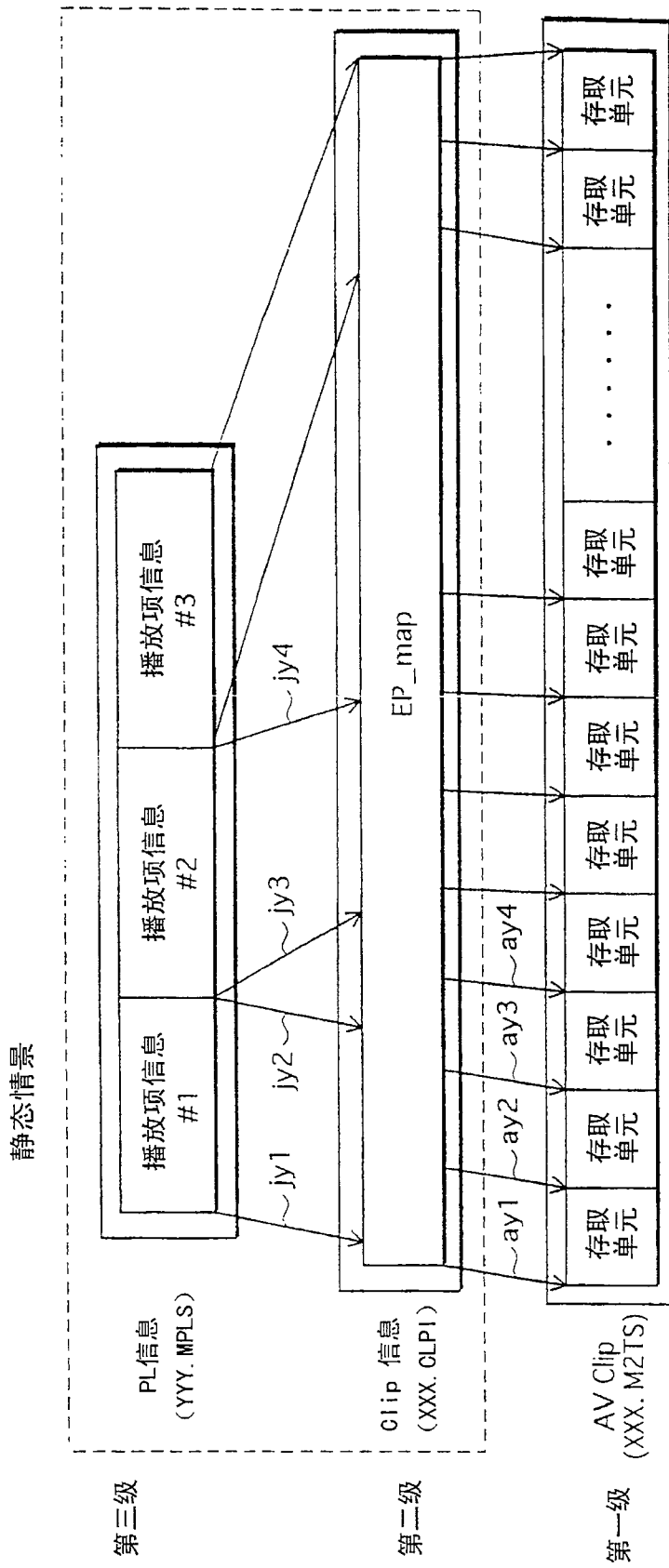


图7

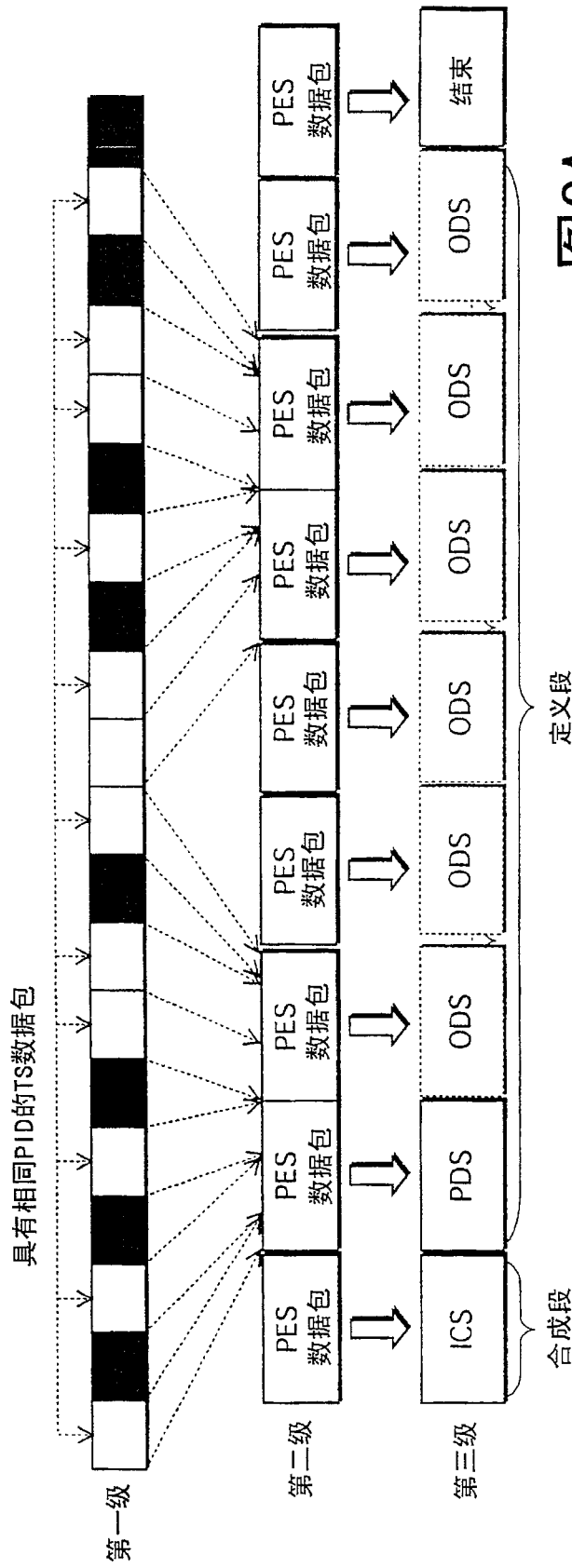


图8A

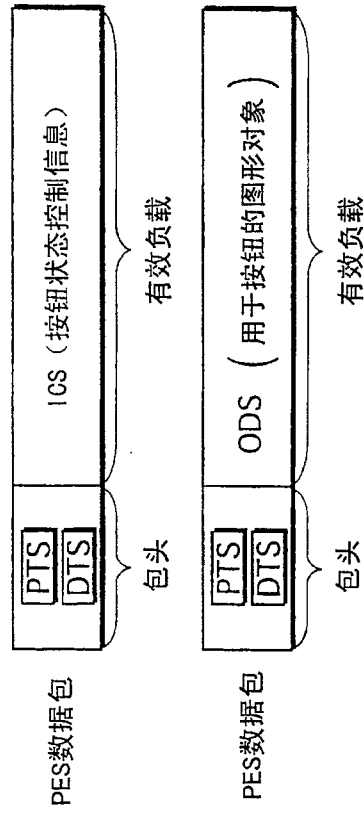


图8B



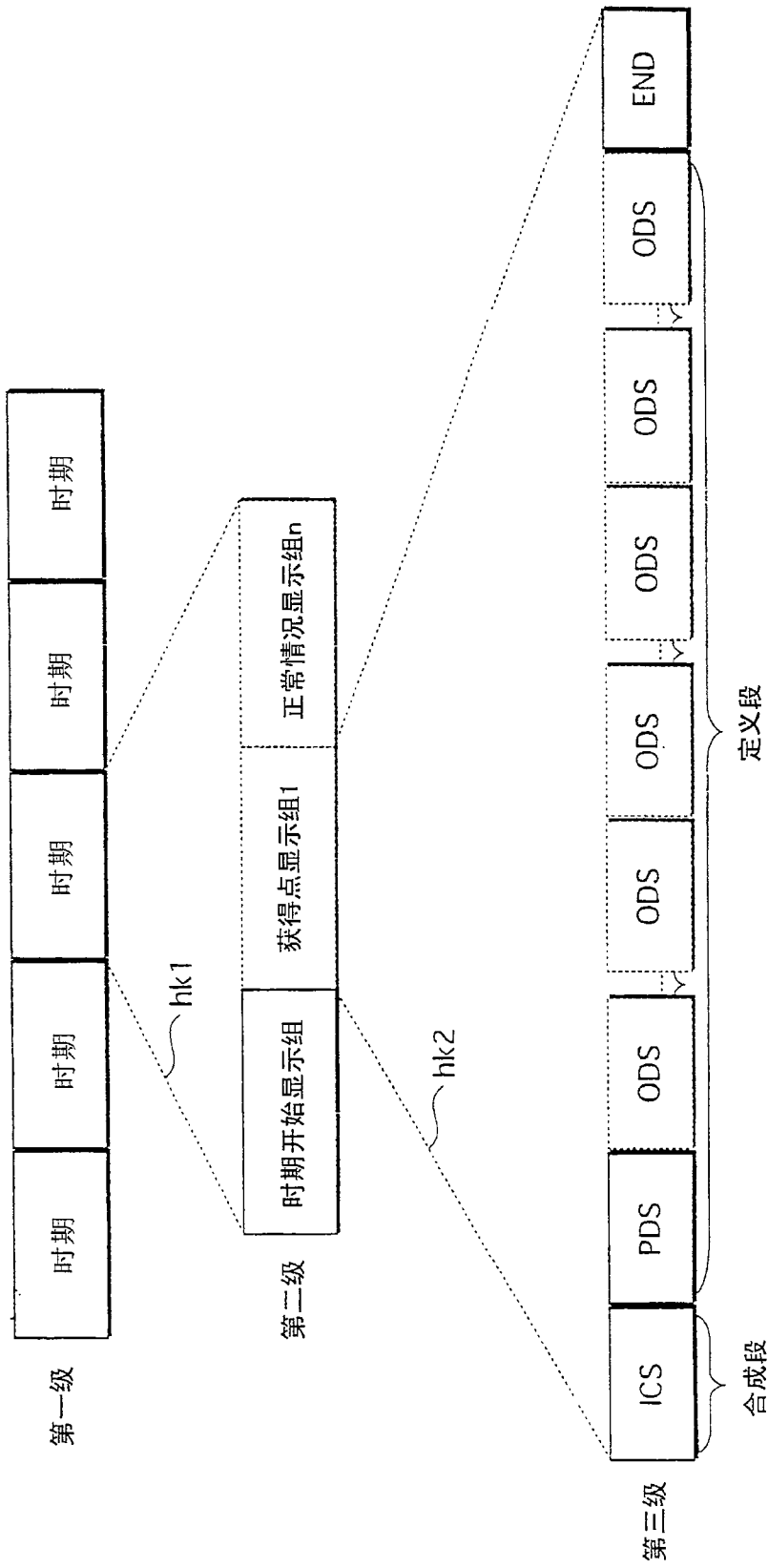


图9

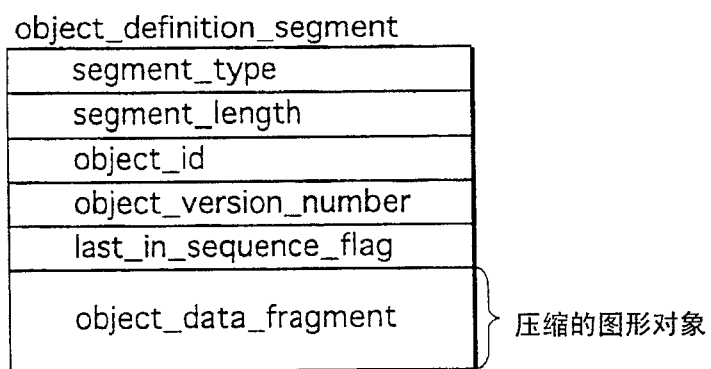


图10A

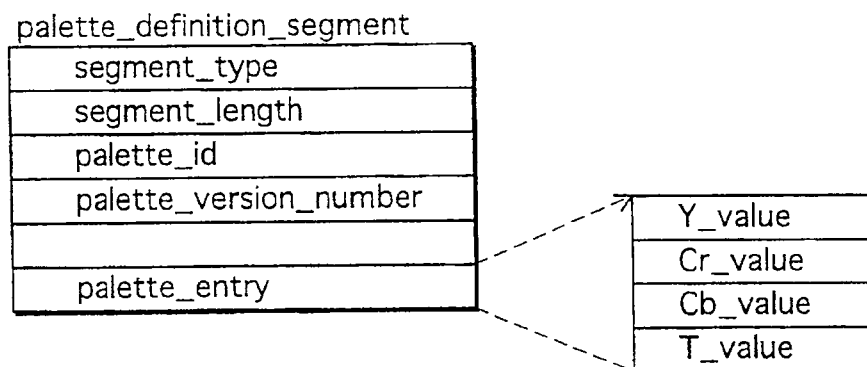


图10B

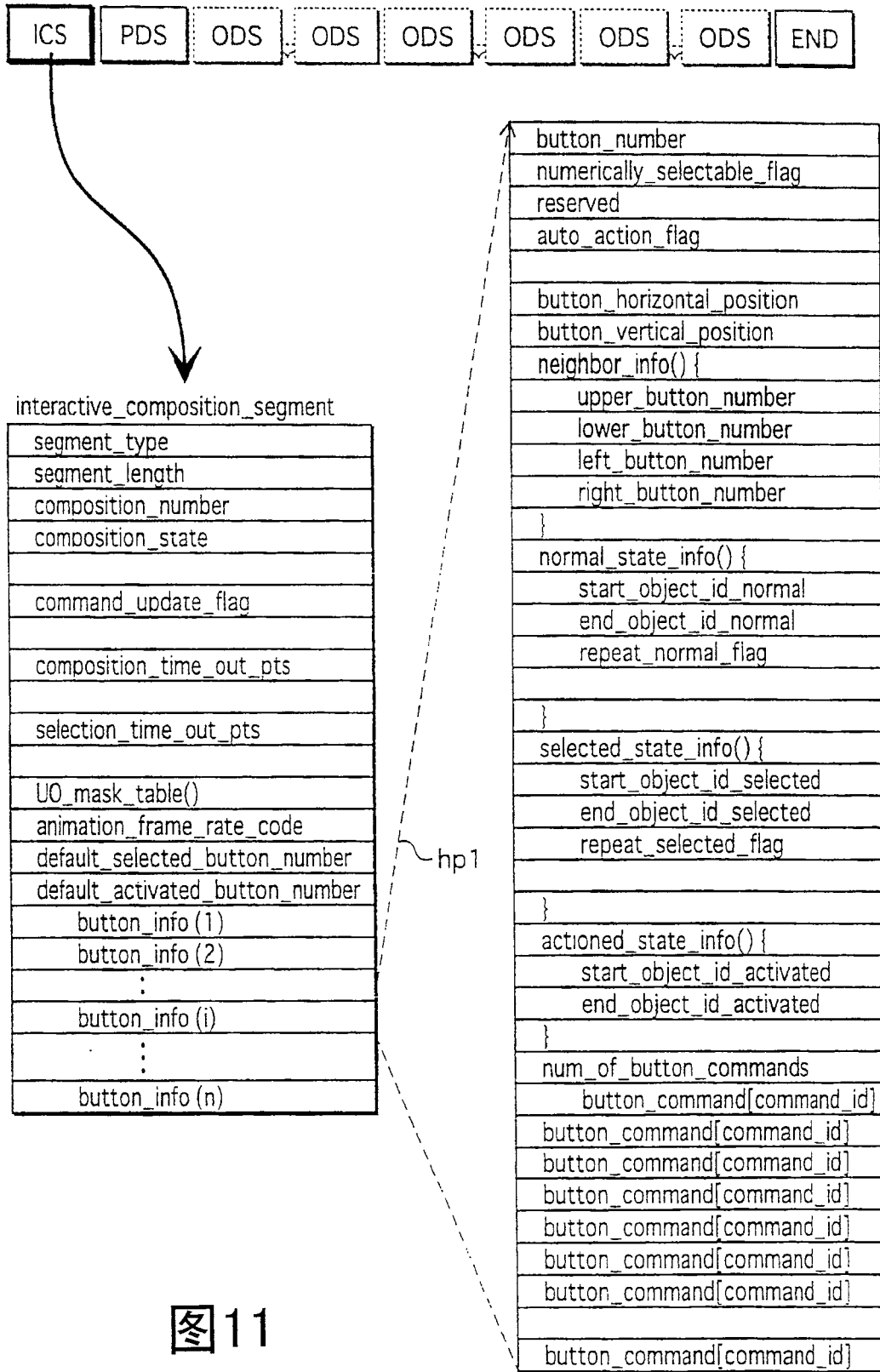


图 11

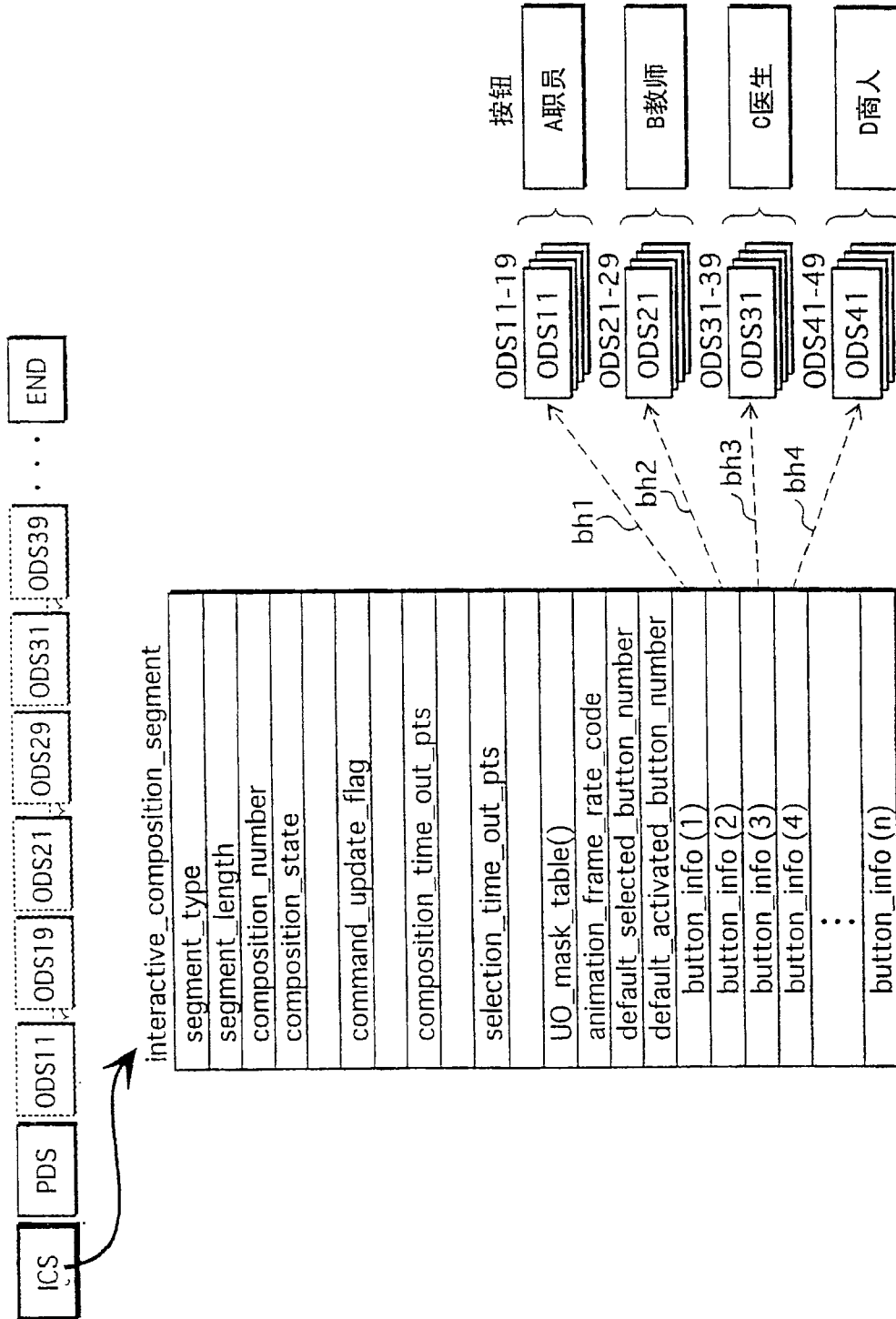


图12

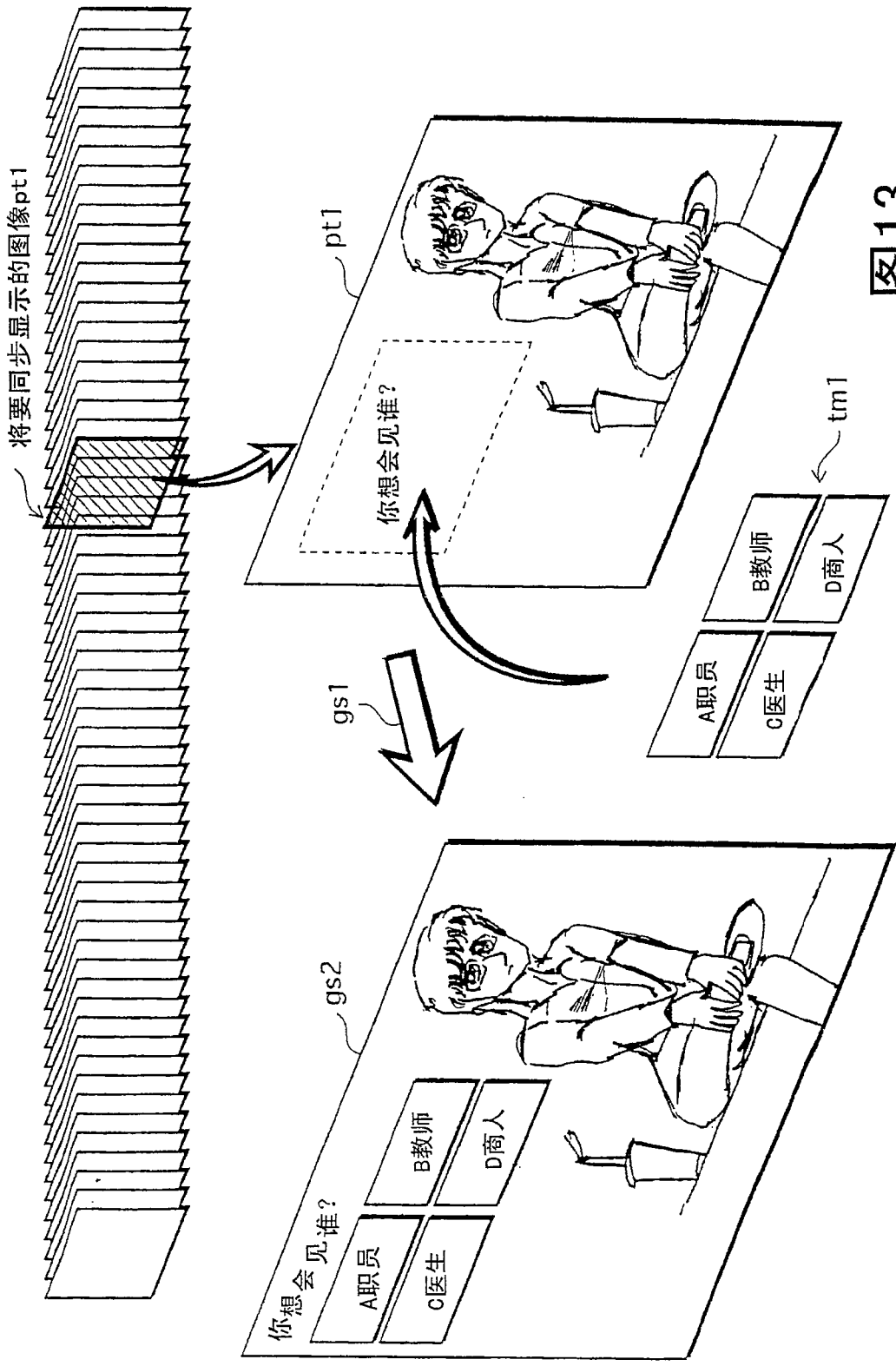


图13

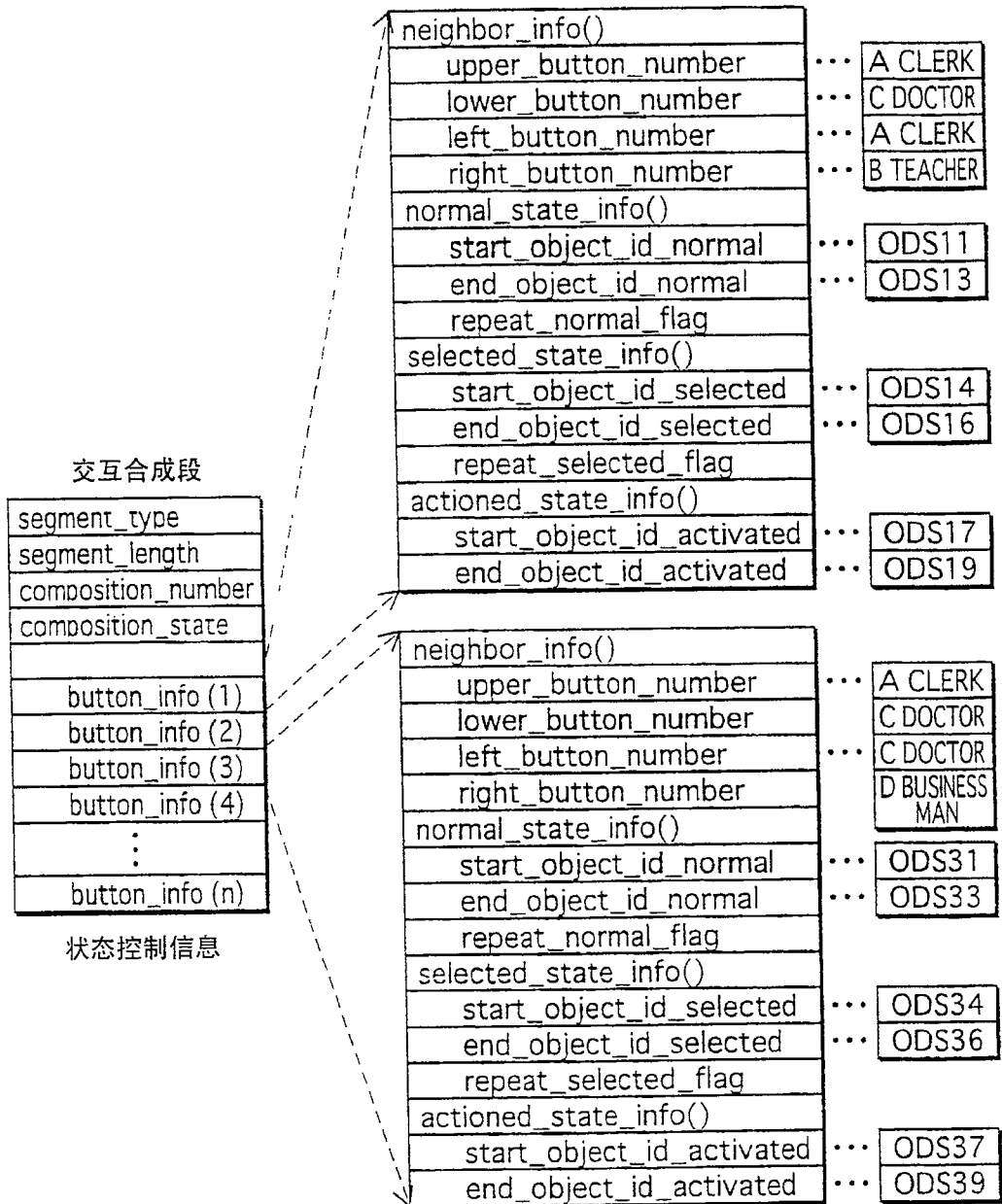


图14

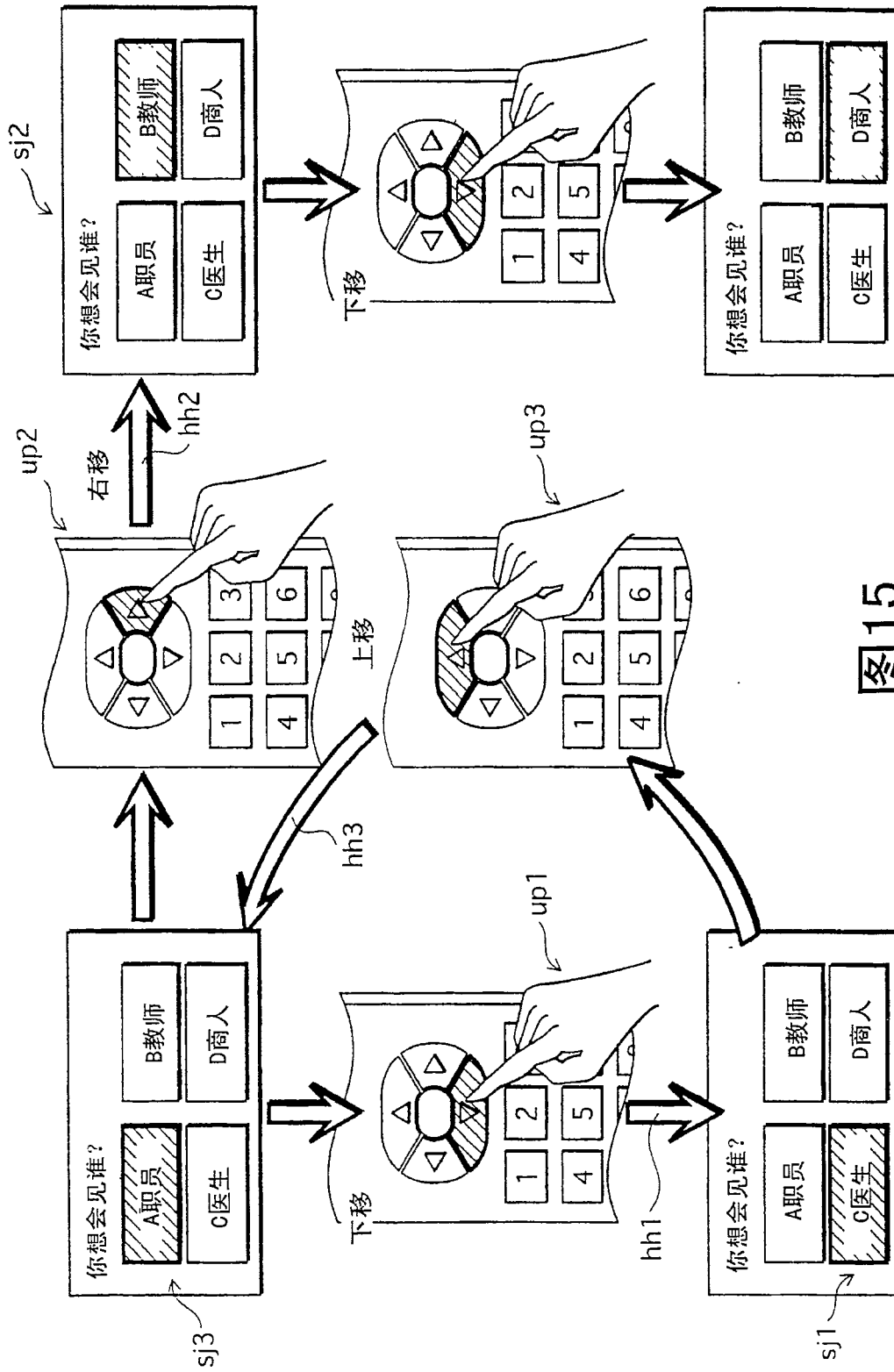


图15



图16



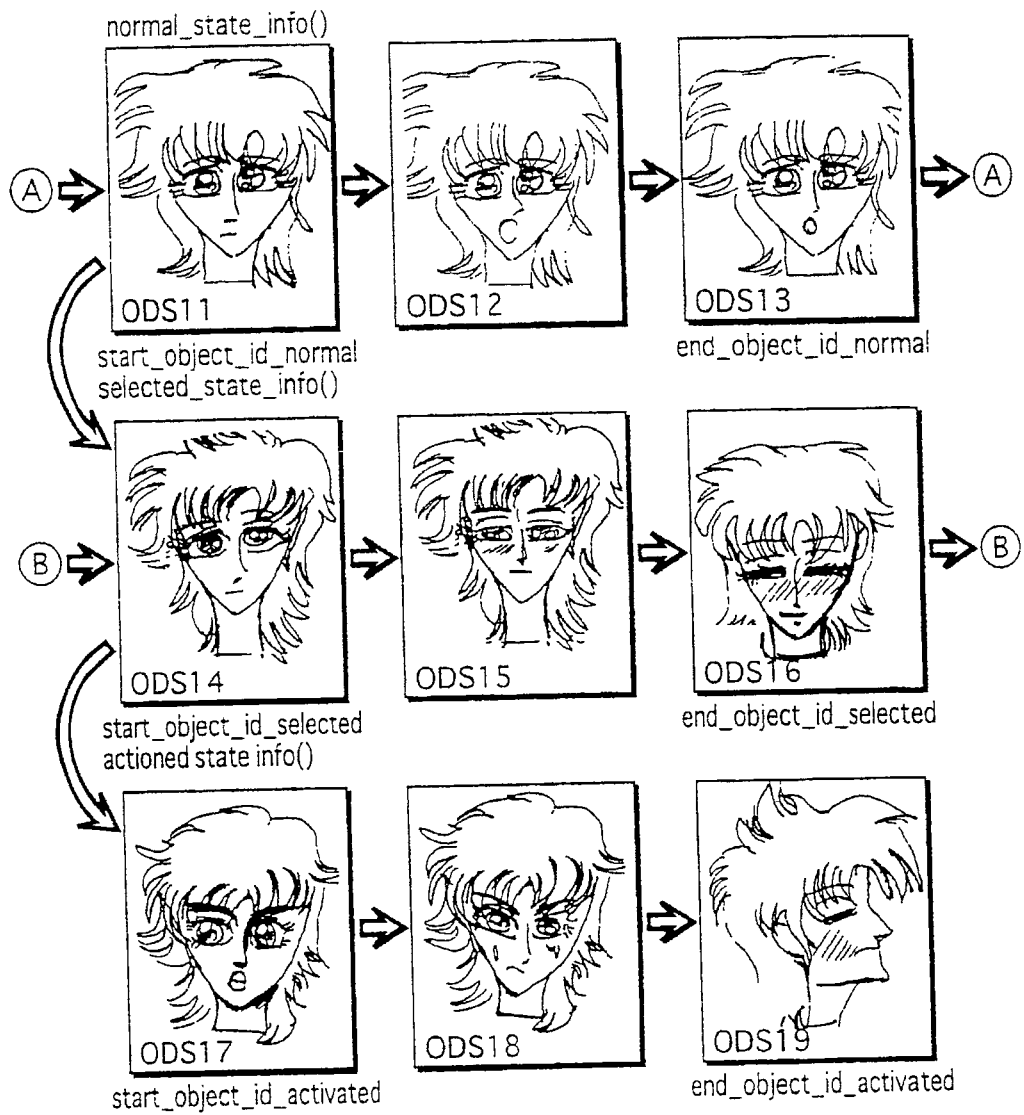


图17

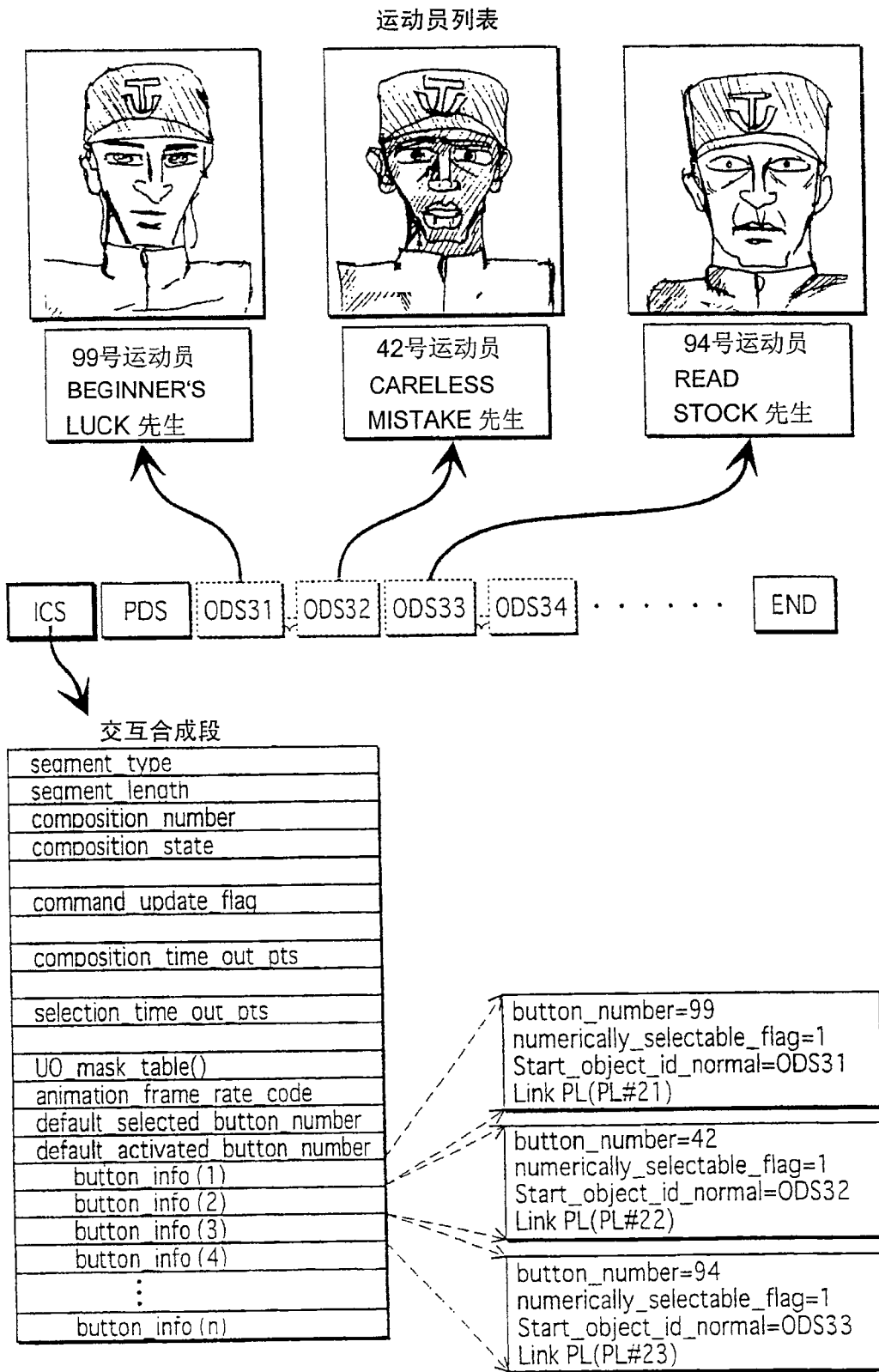


图18

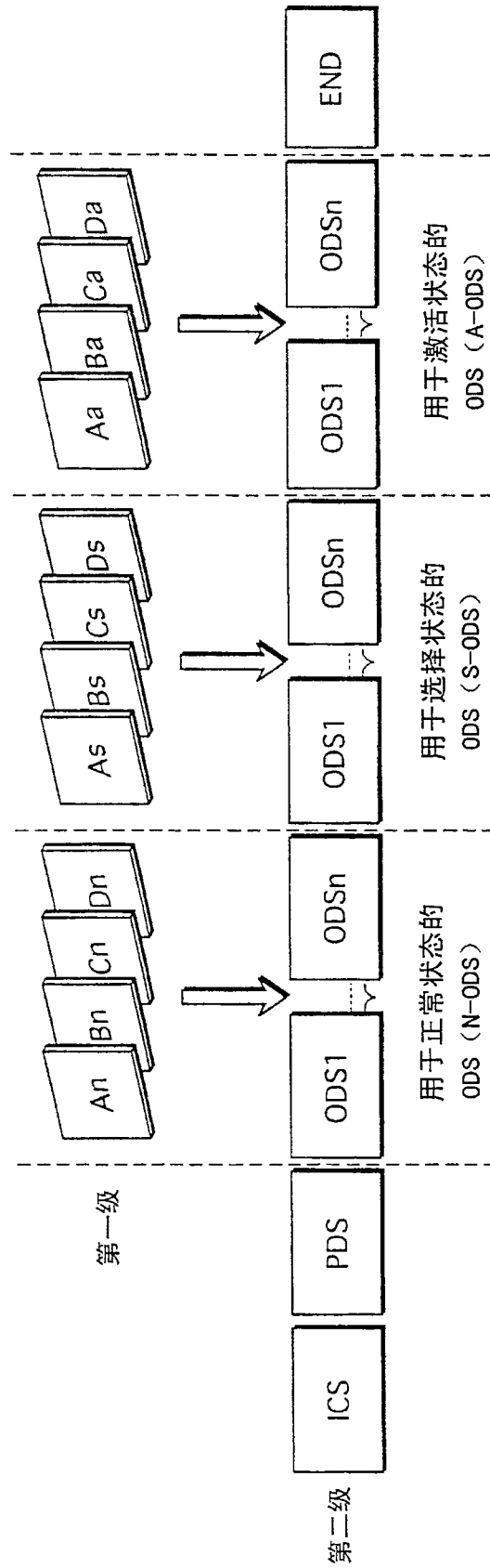


图19

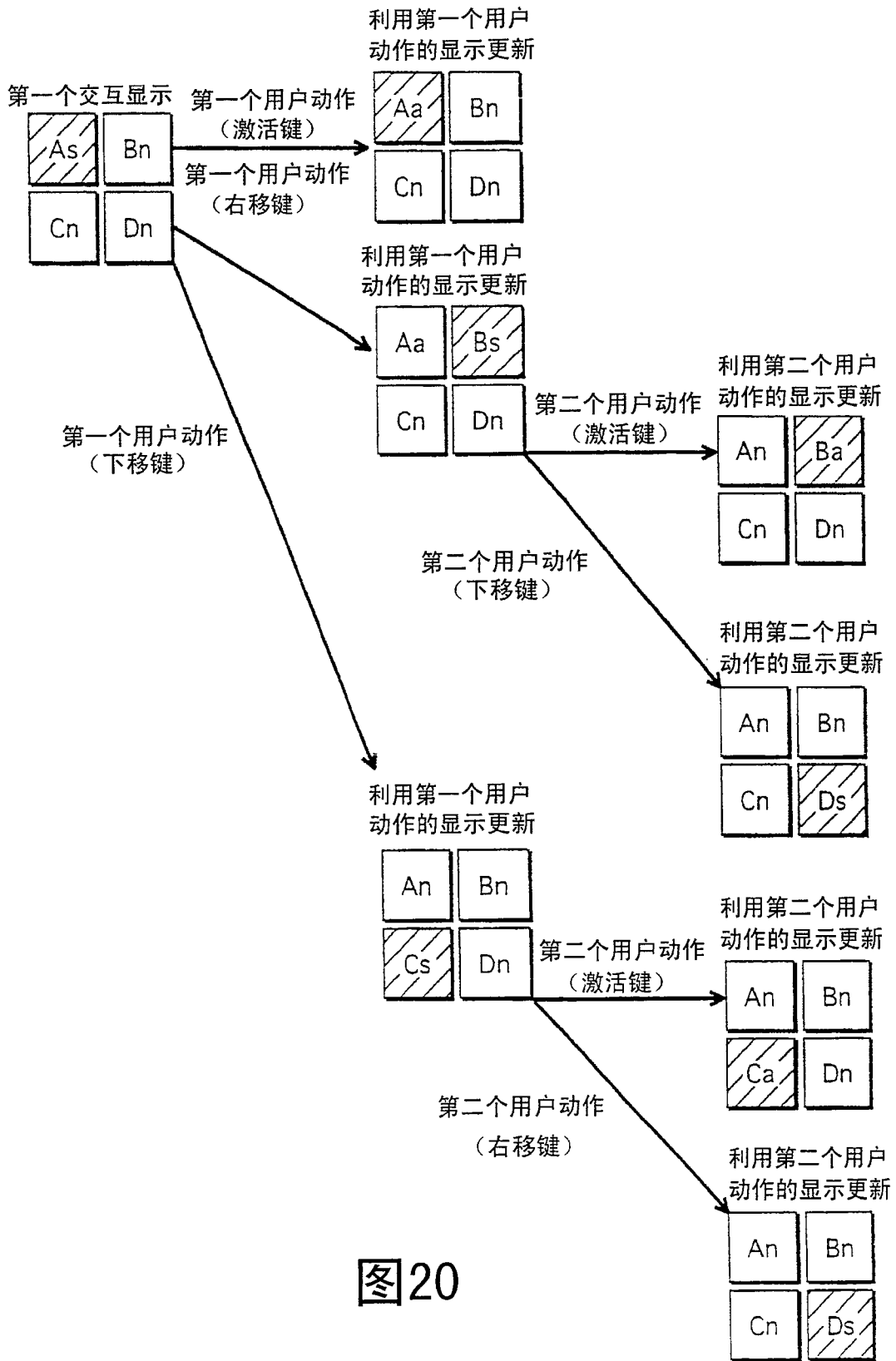


图20

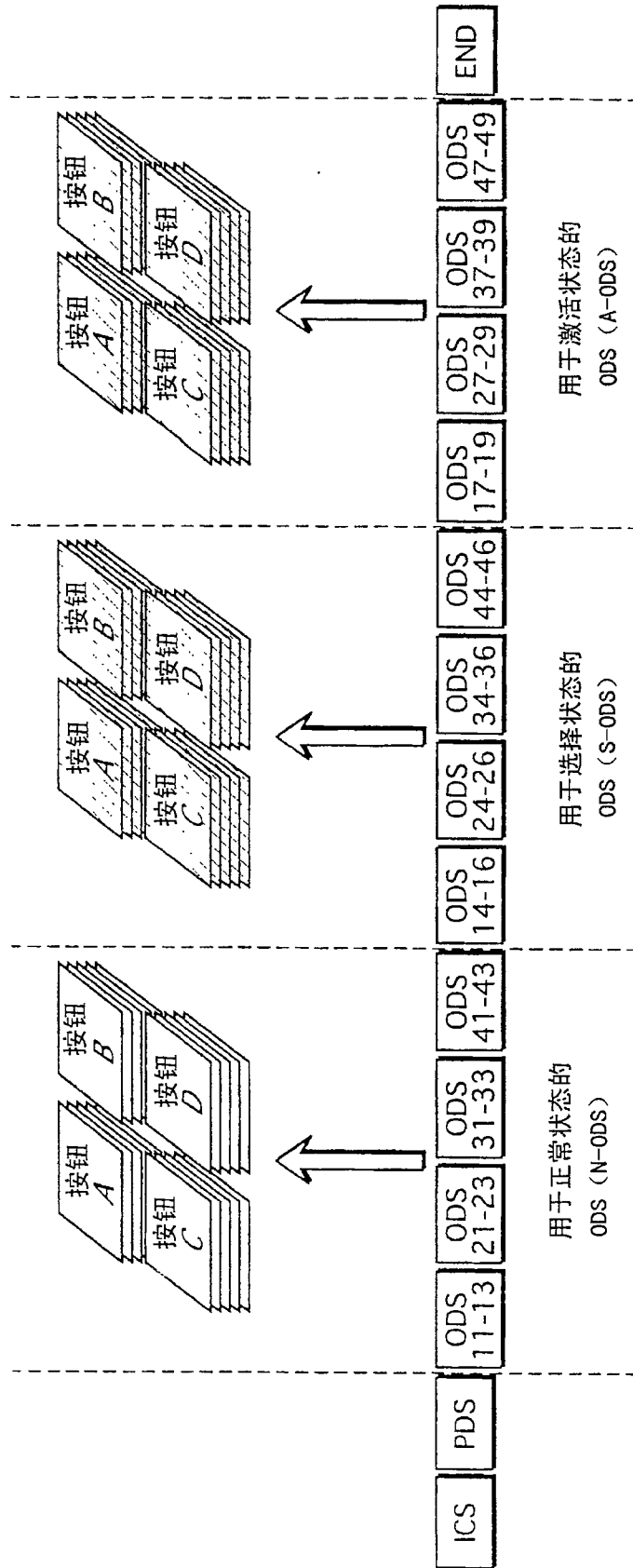


图21

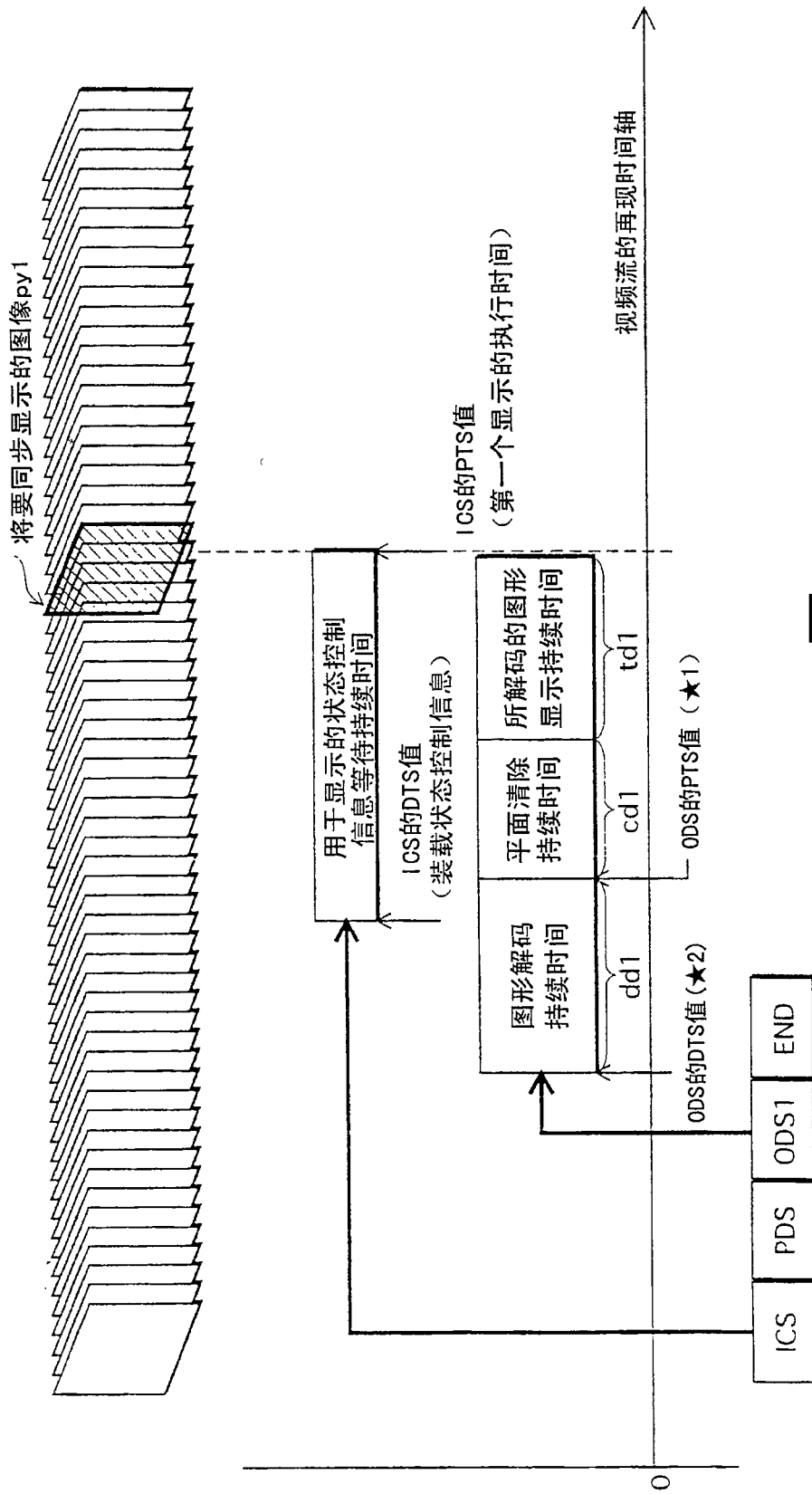


图22

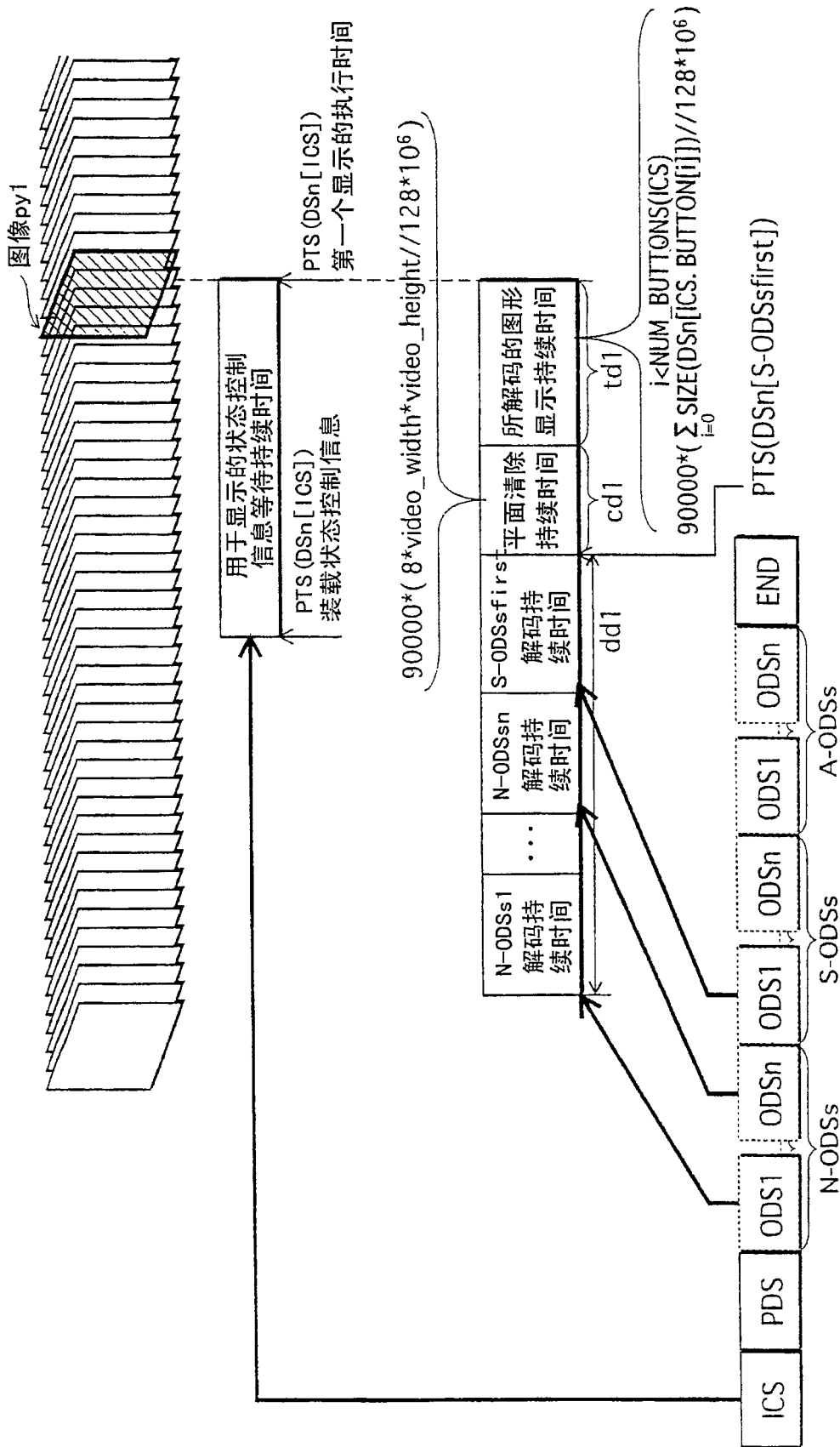


图23

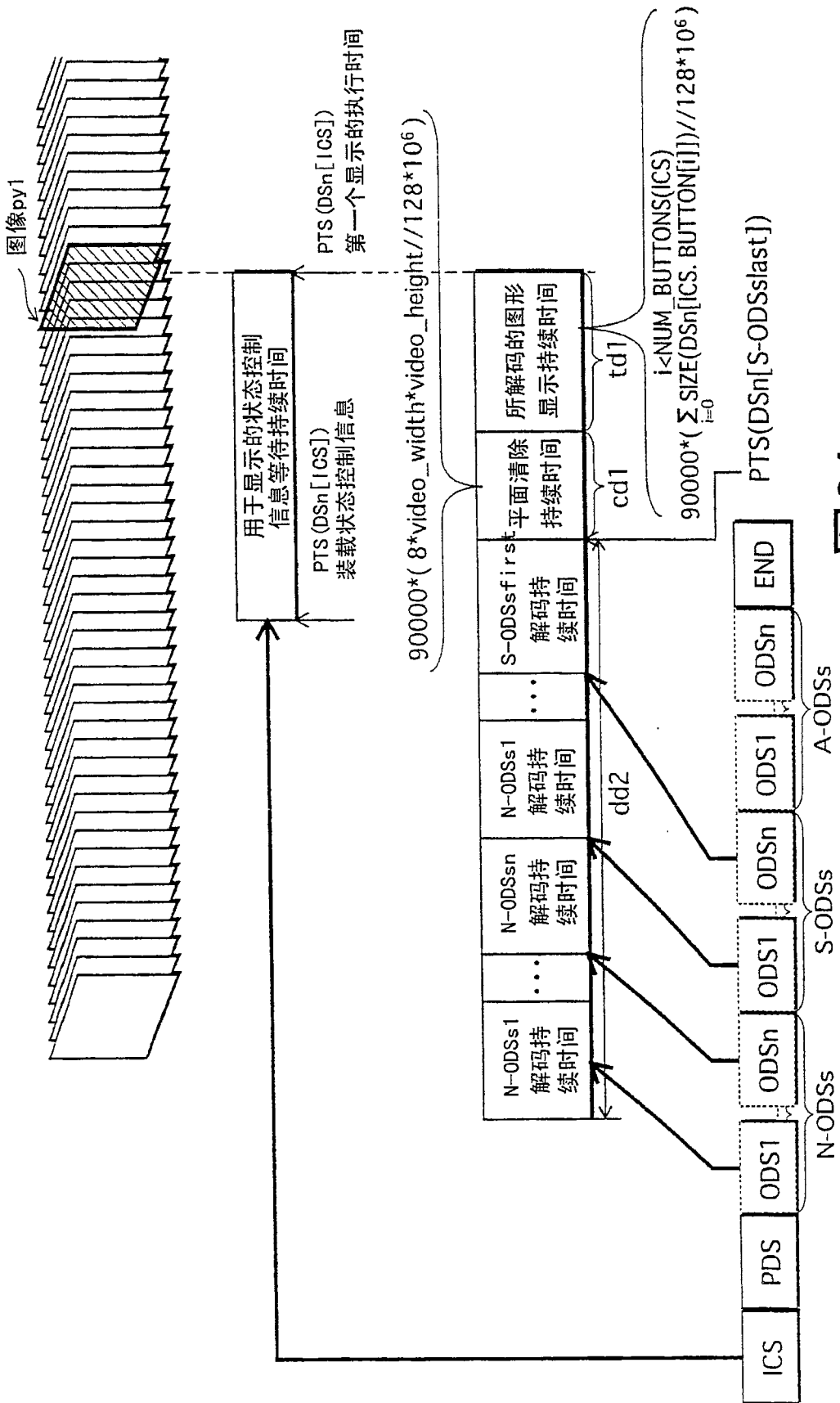


图24



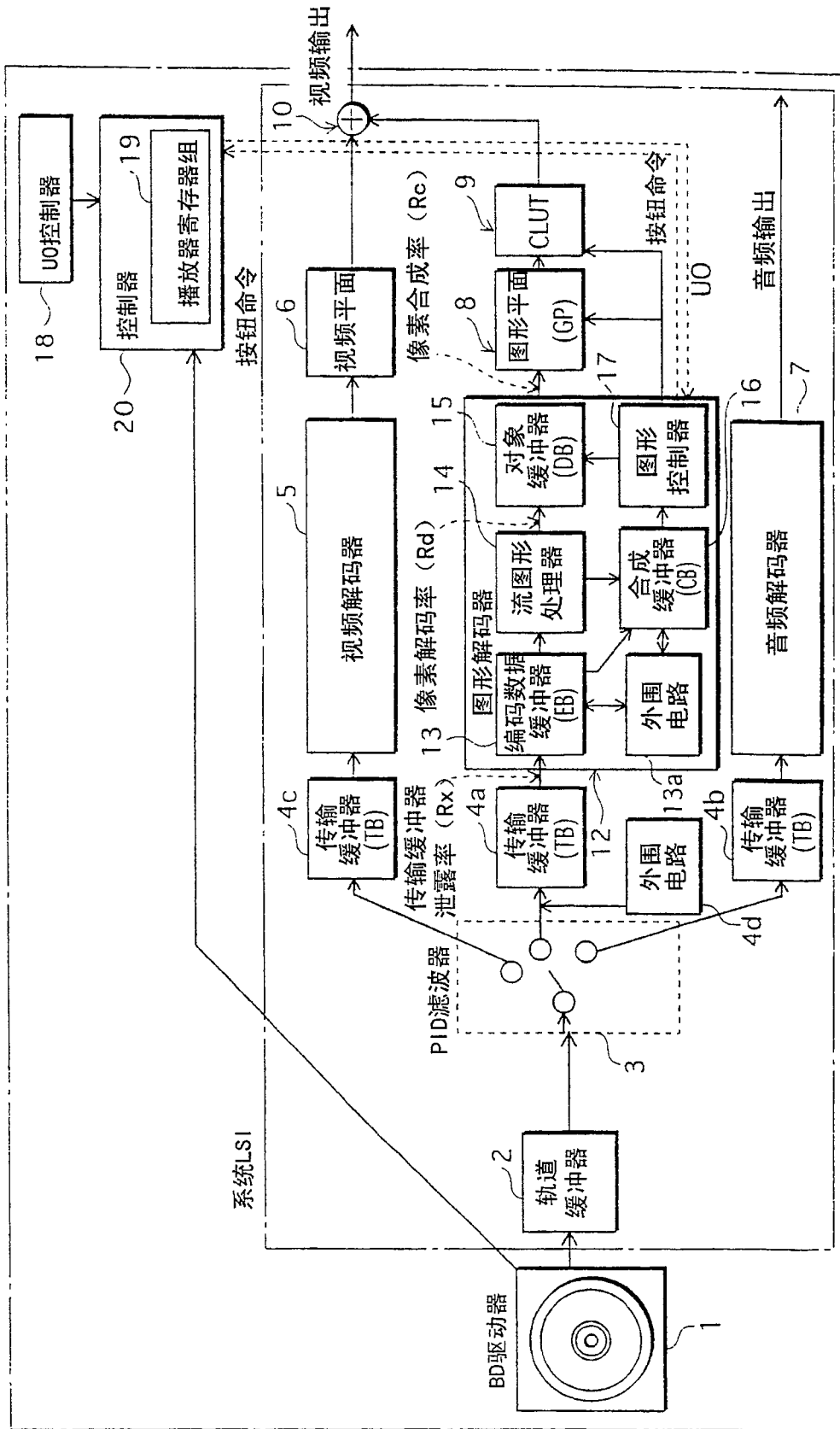


图25

指定的缺省选择按钮

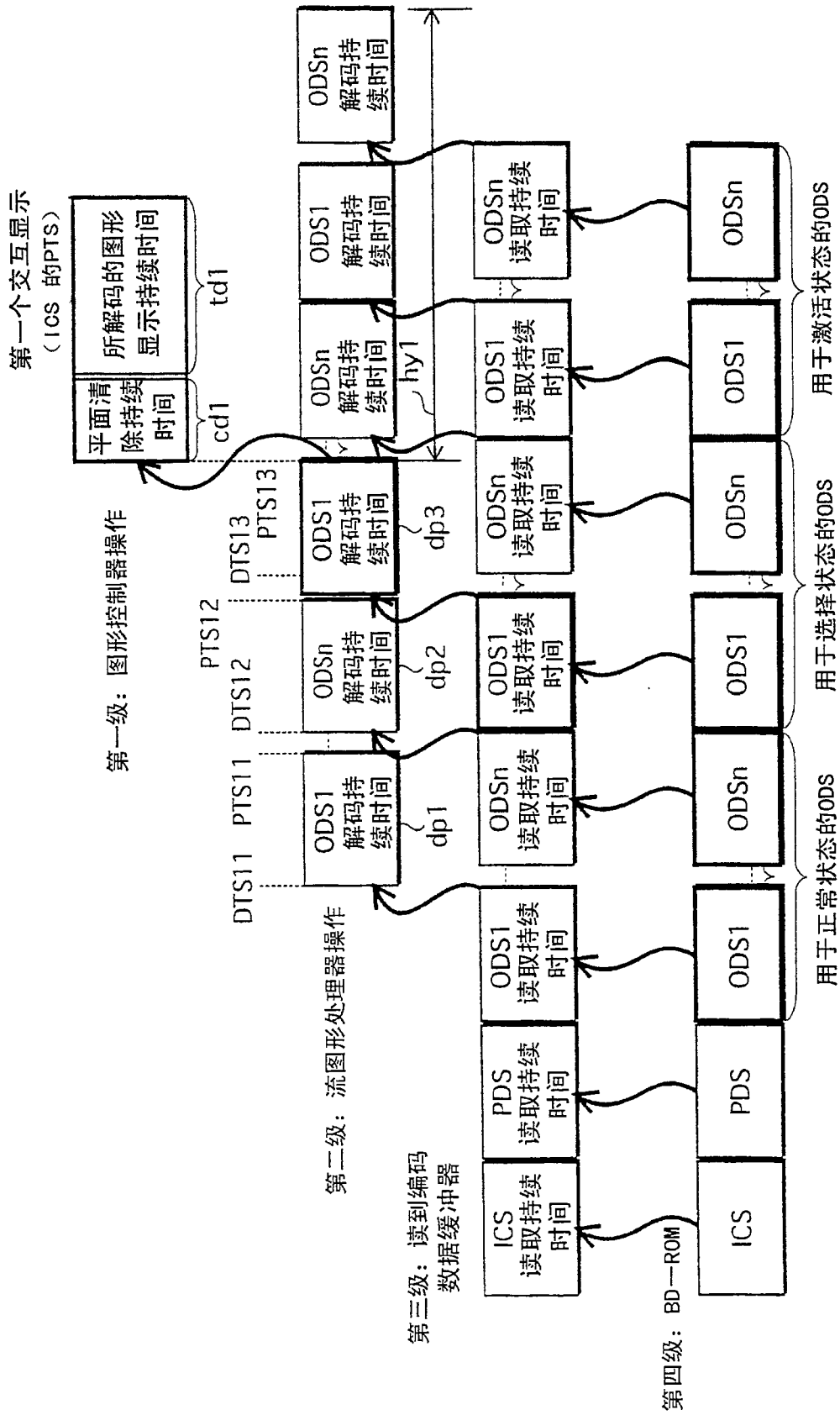
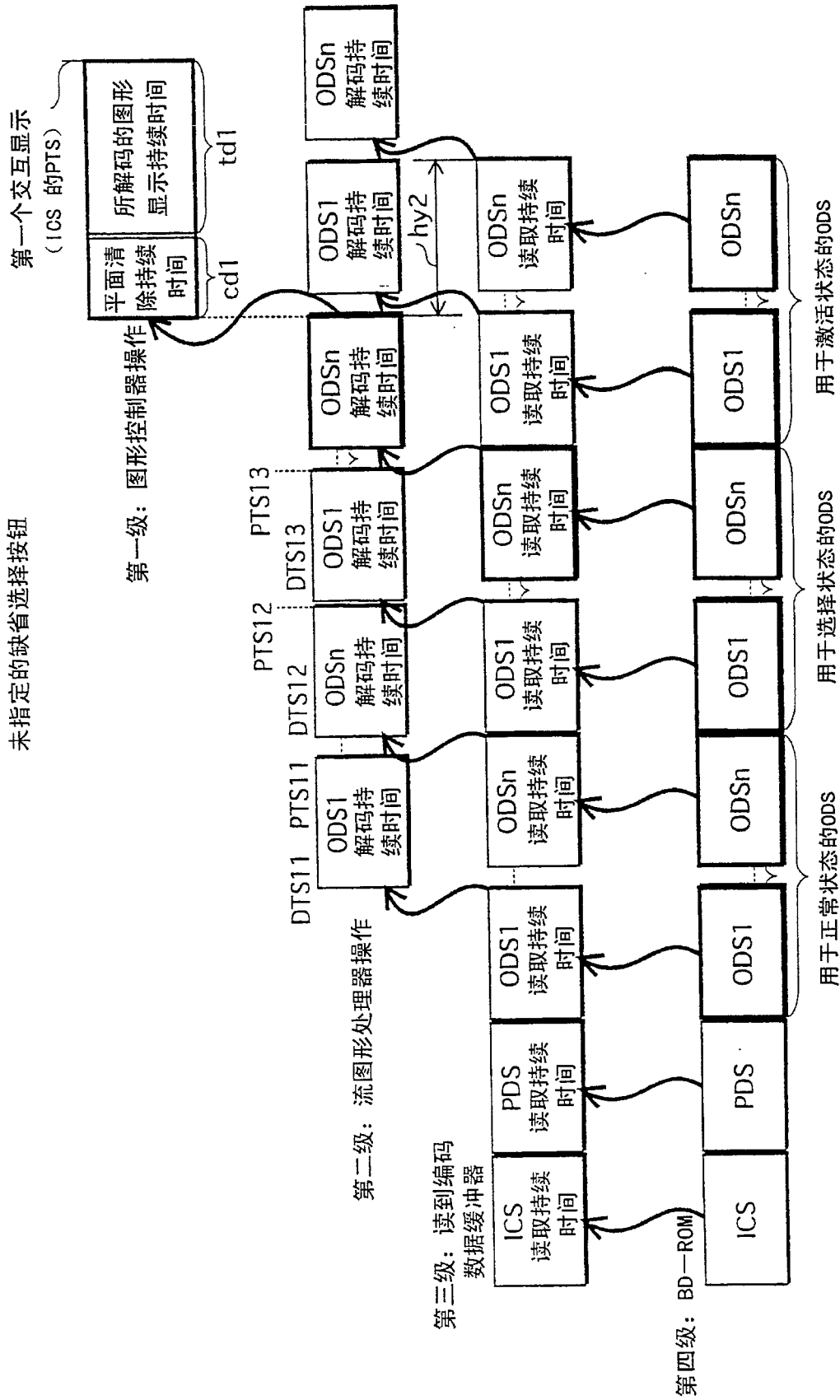


图 26



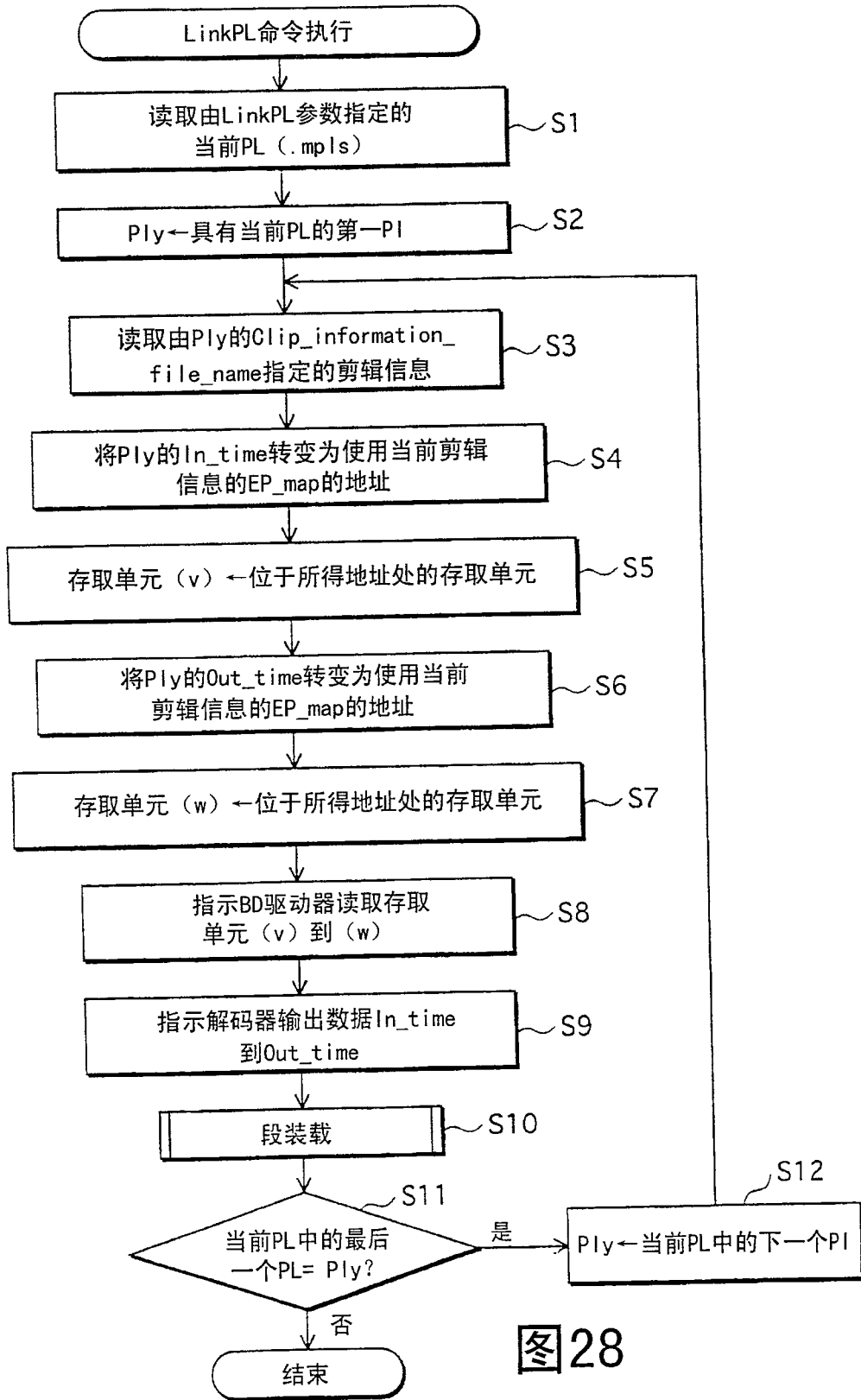


图28

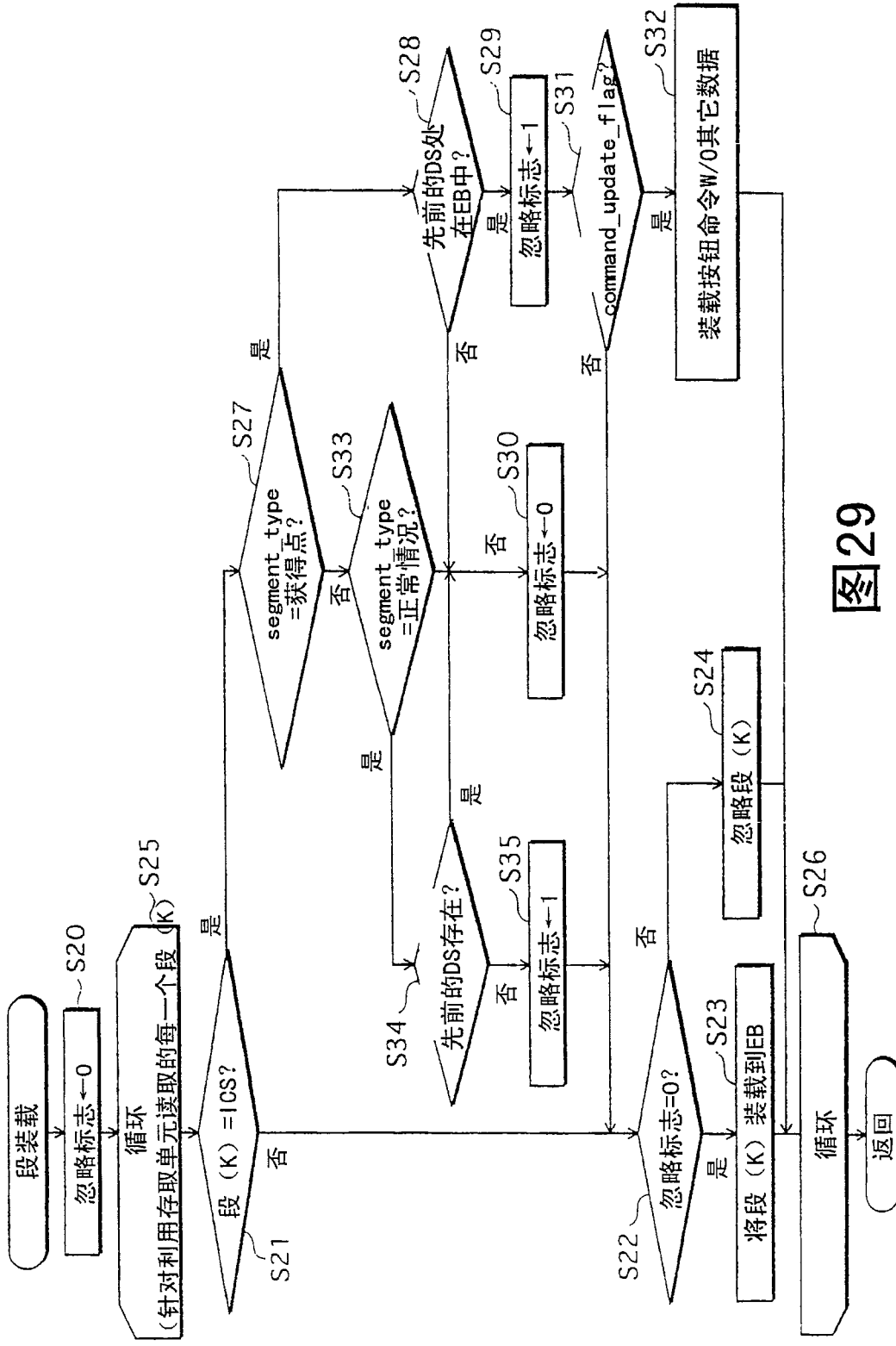


图29

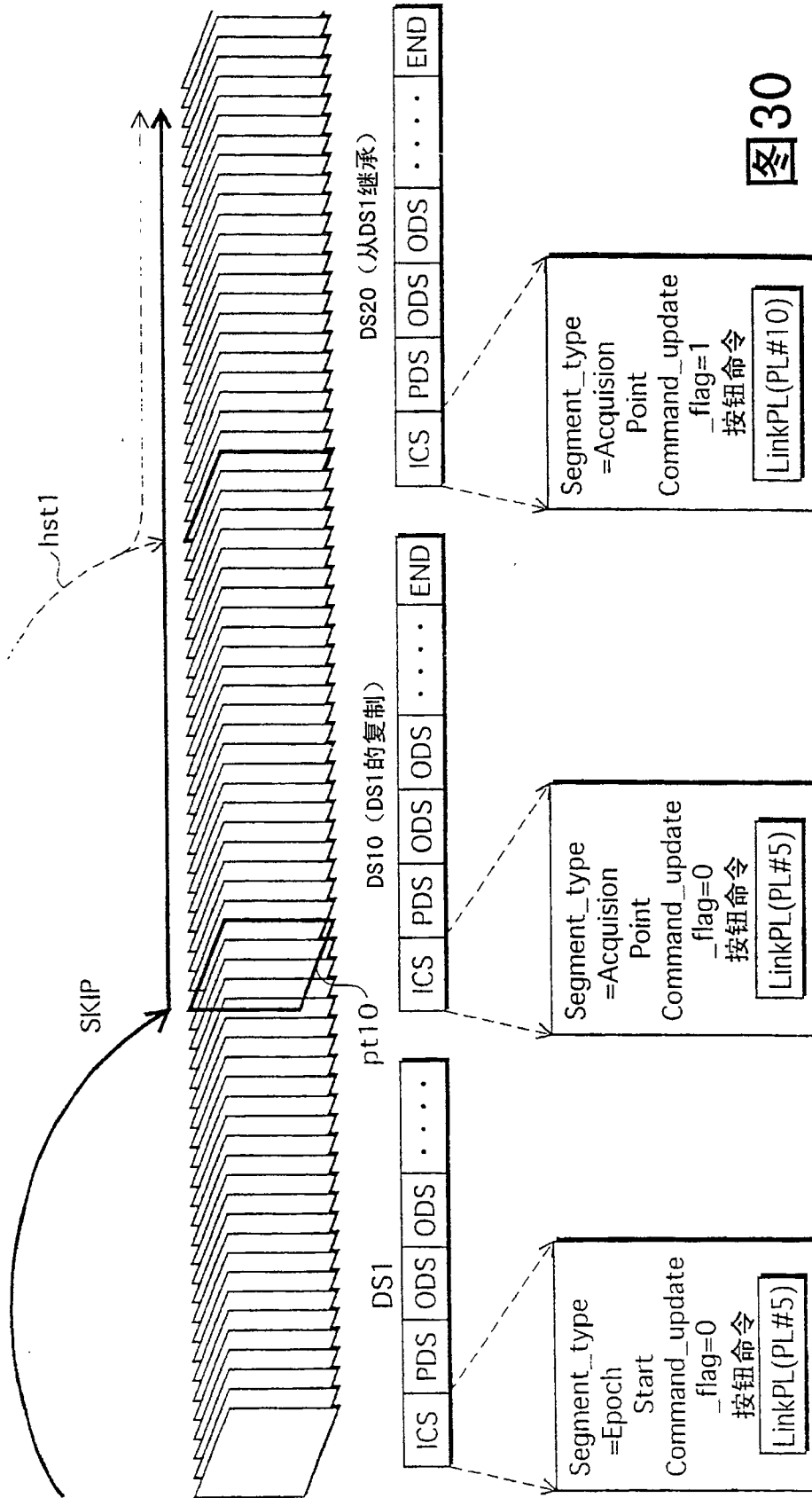


图30

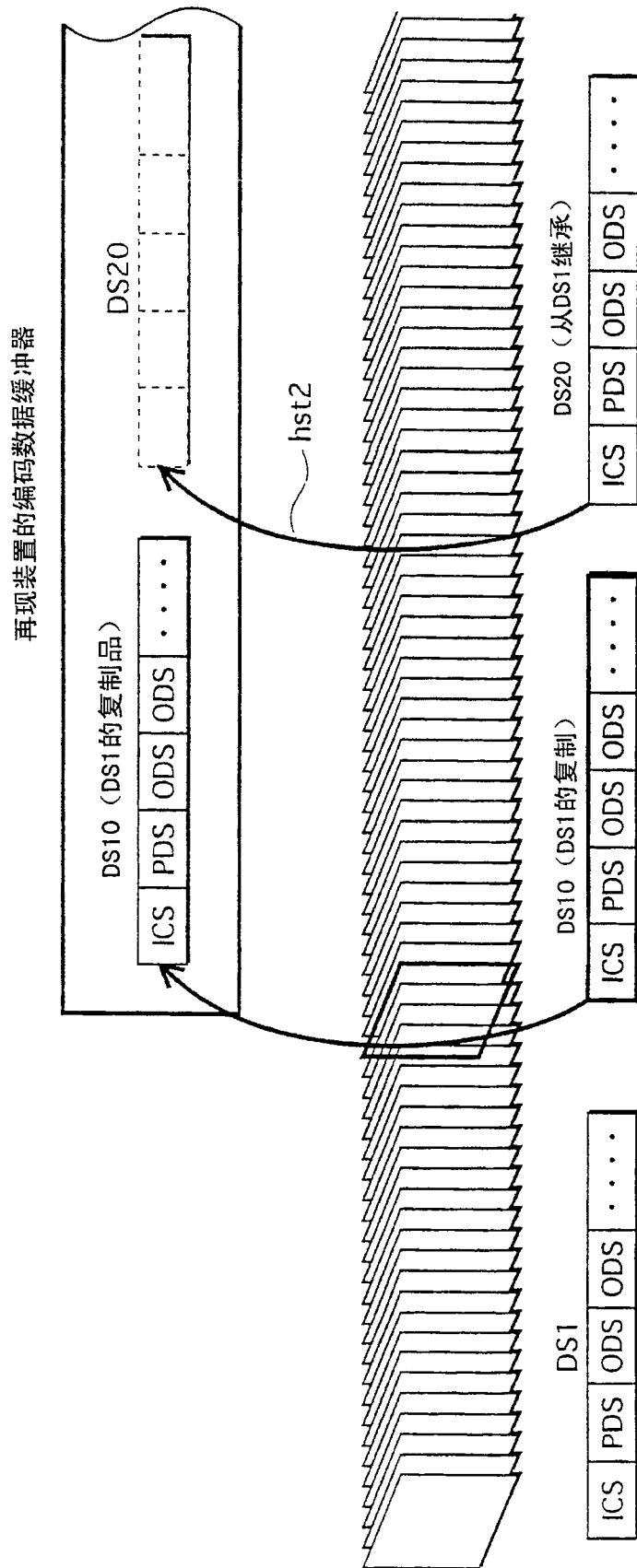


图31

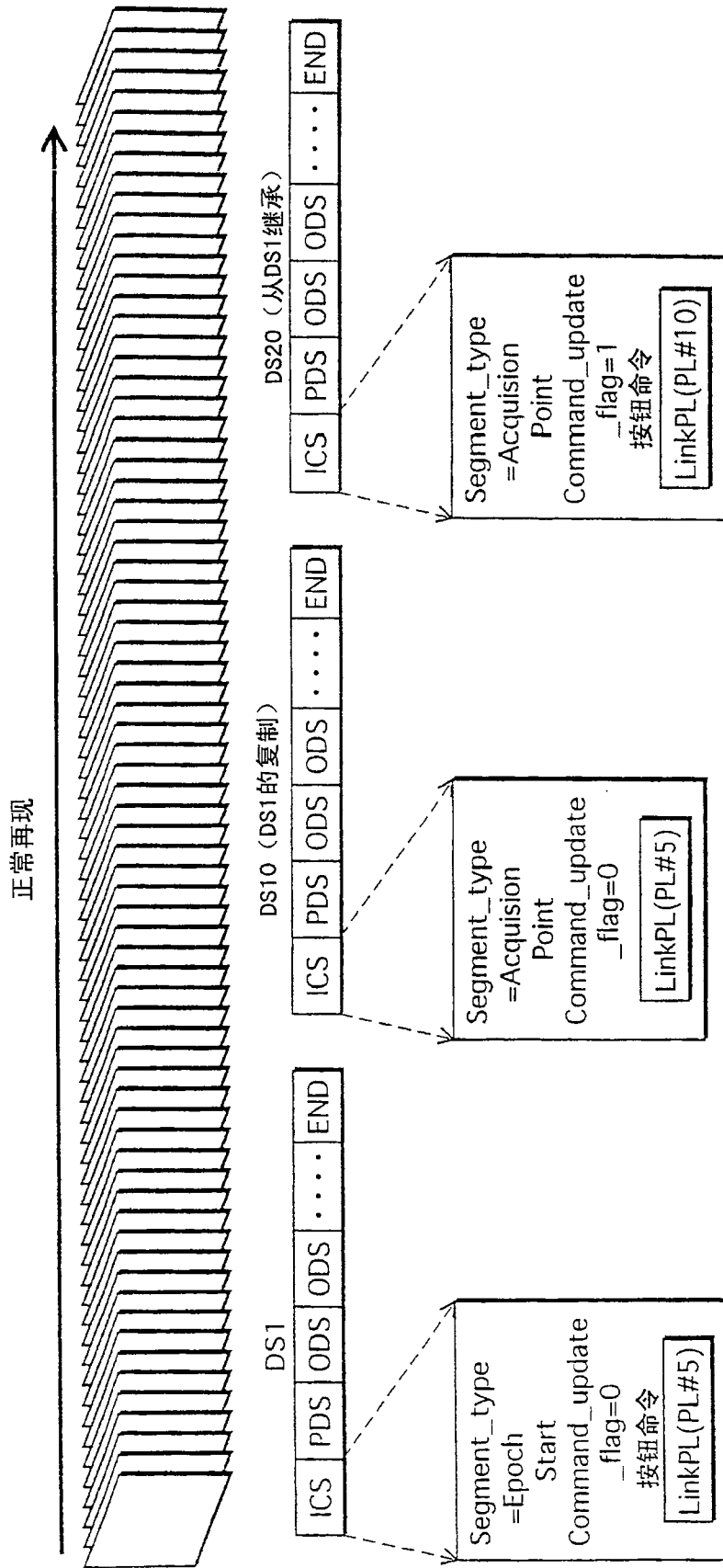
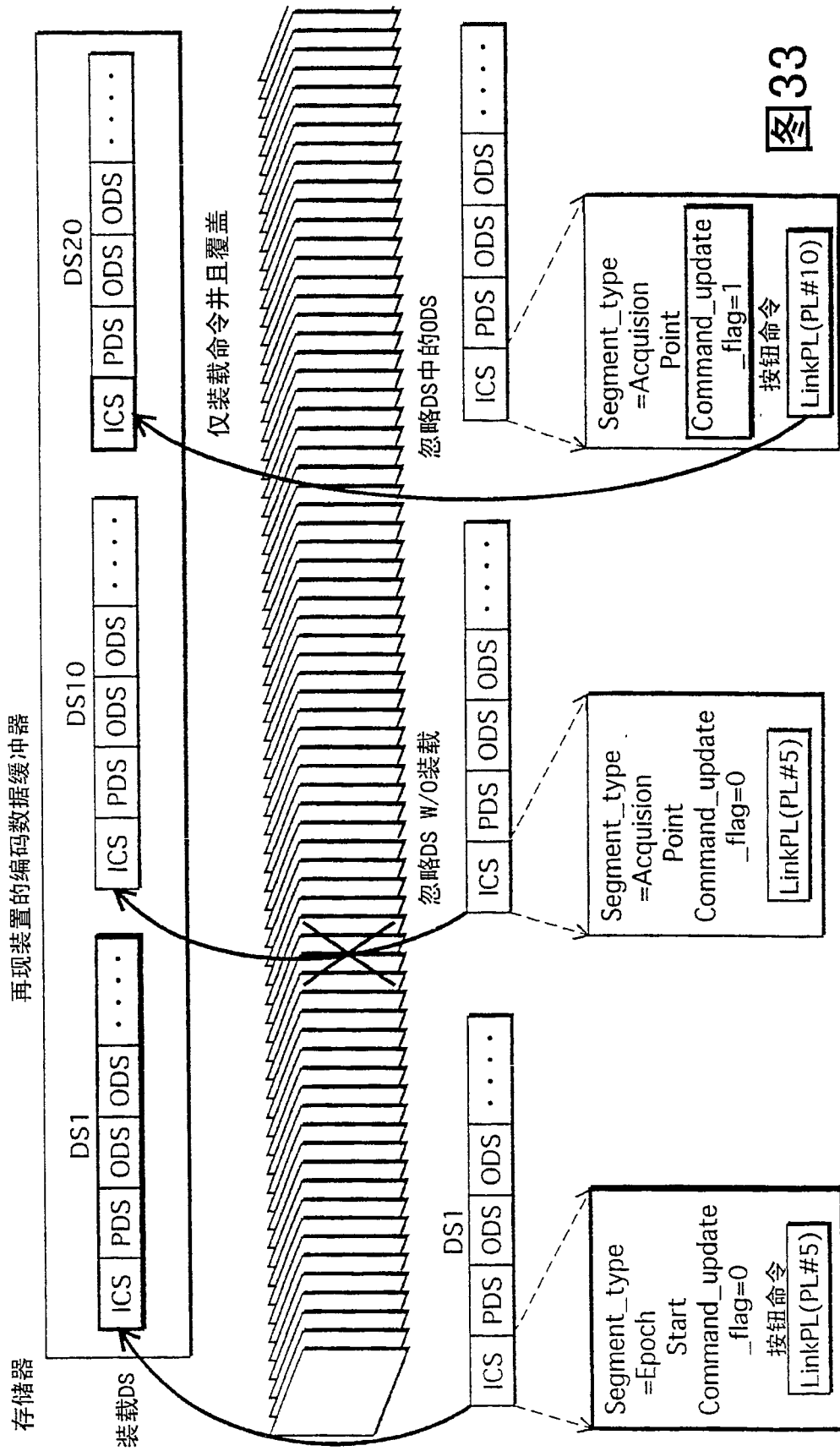


图 32





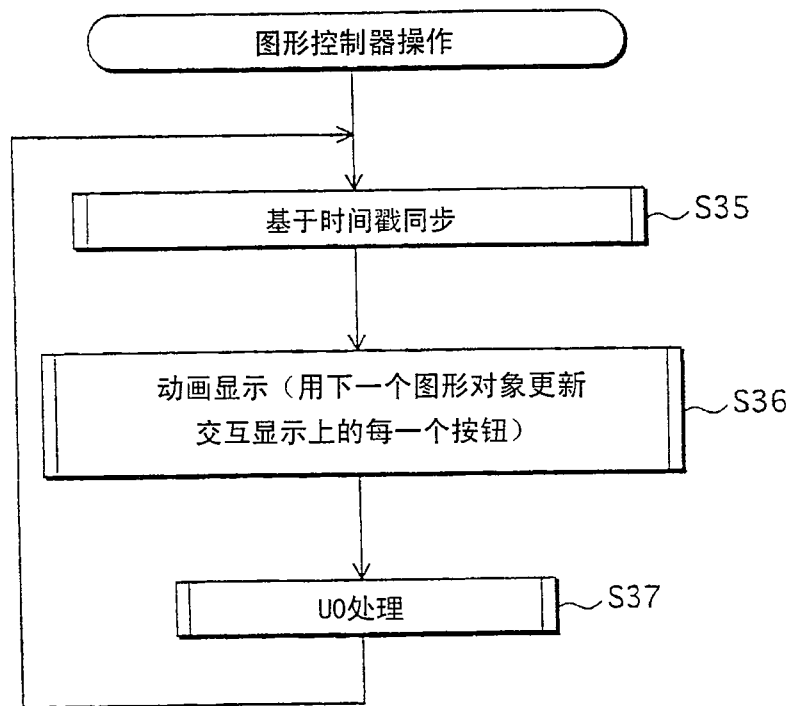


图34

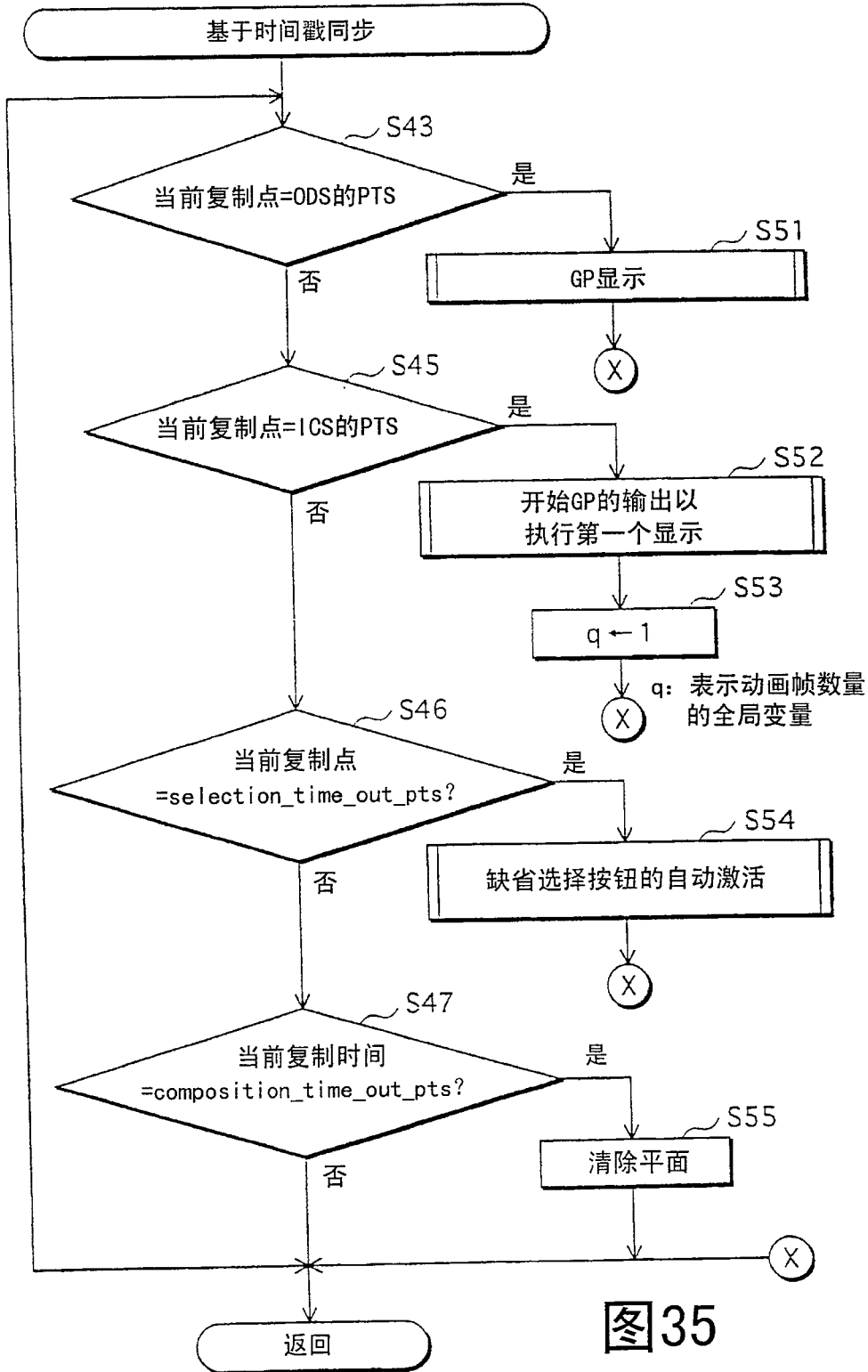


图35

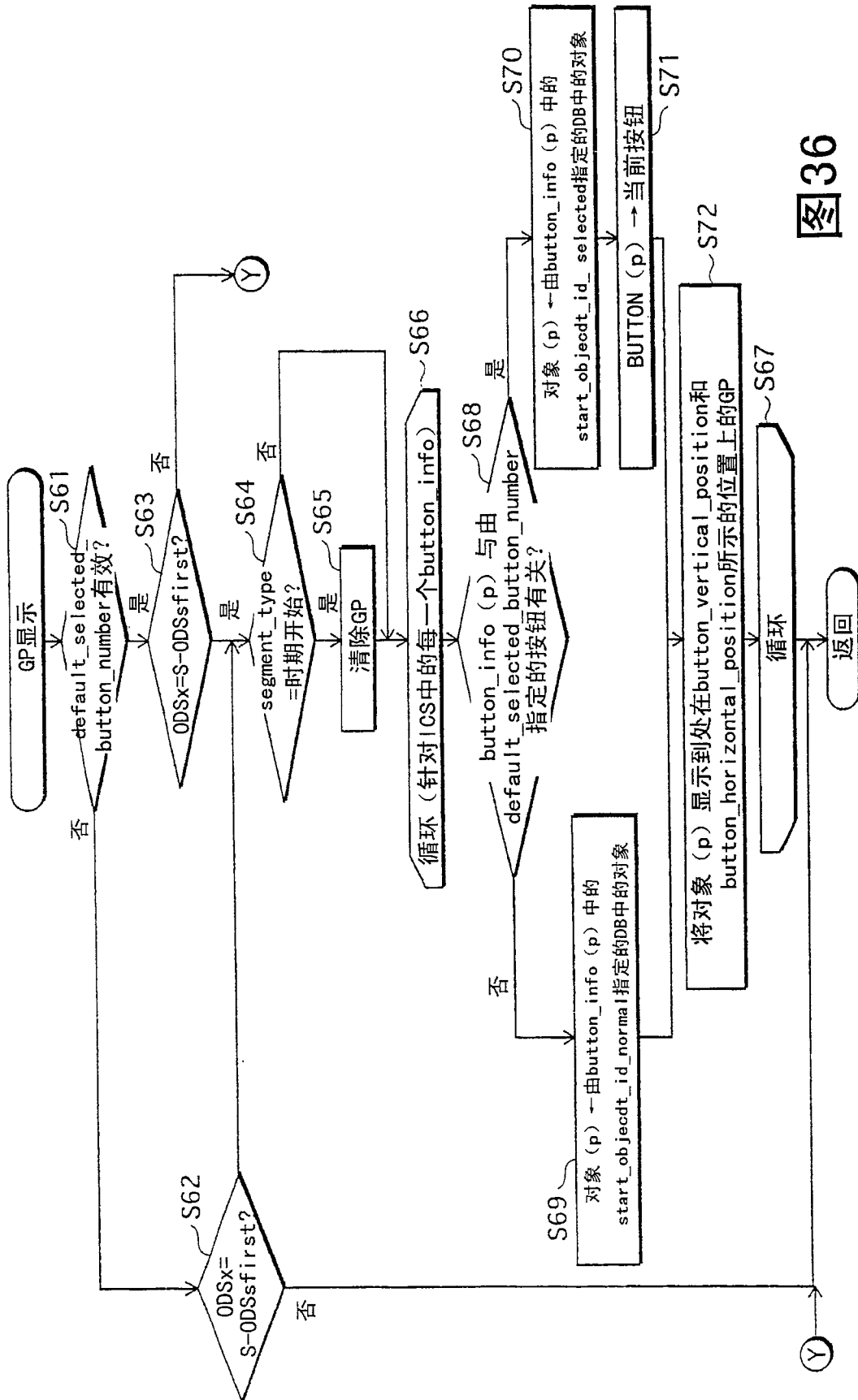


图 36

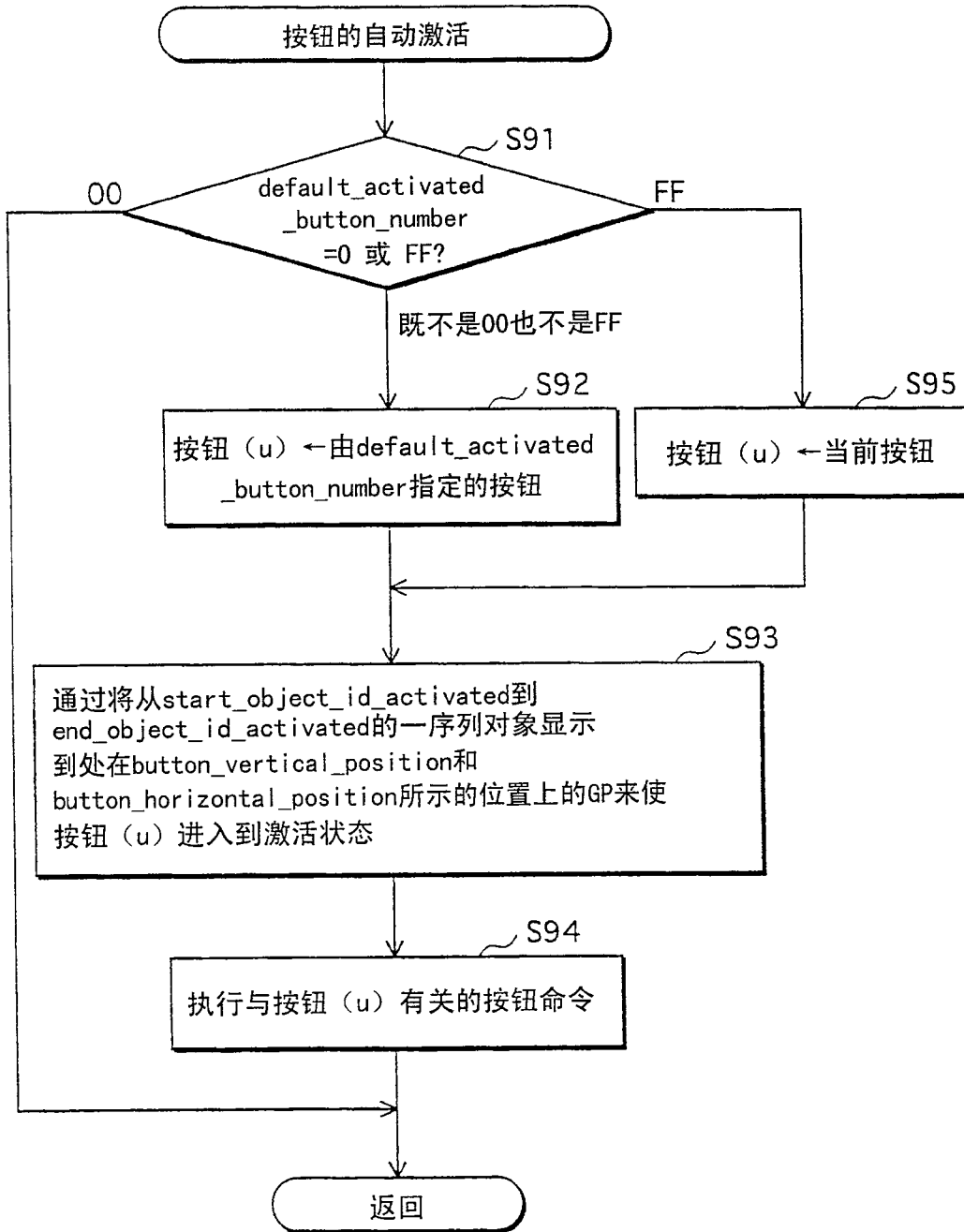


图37

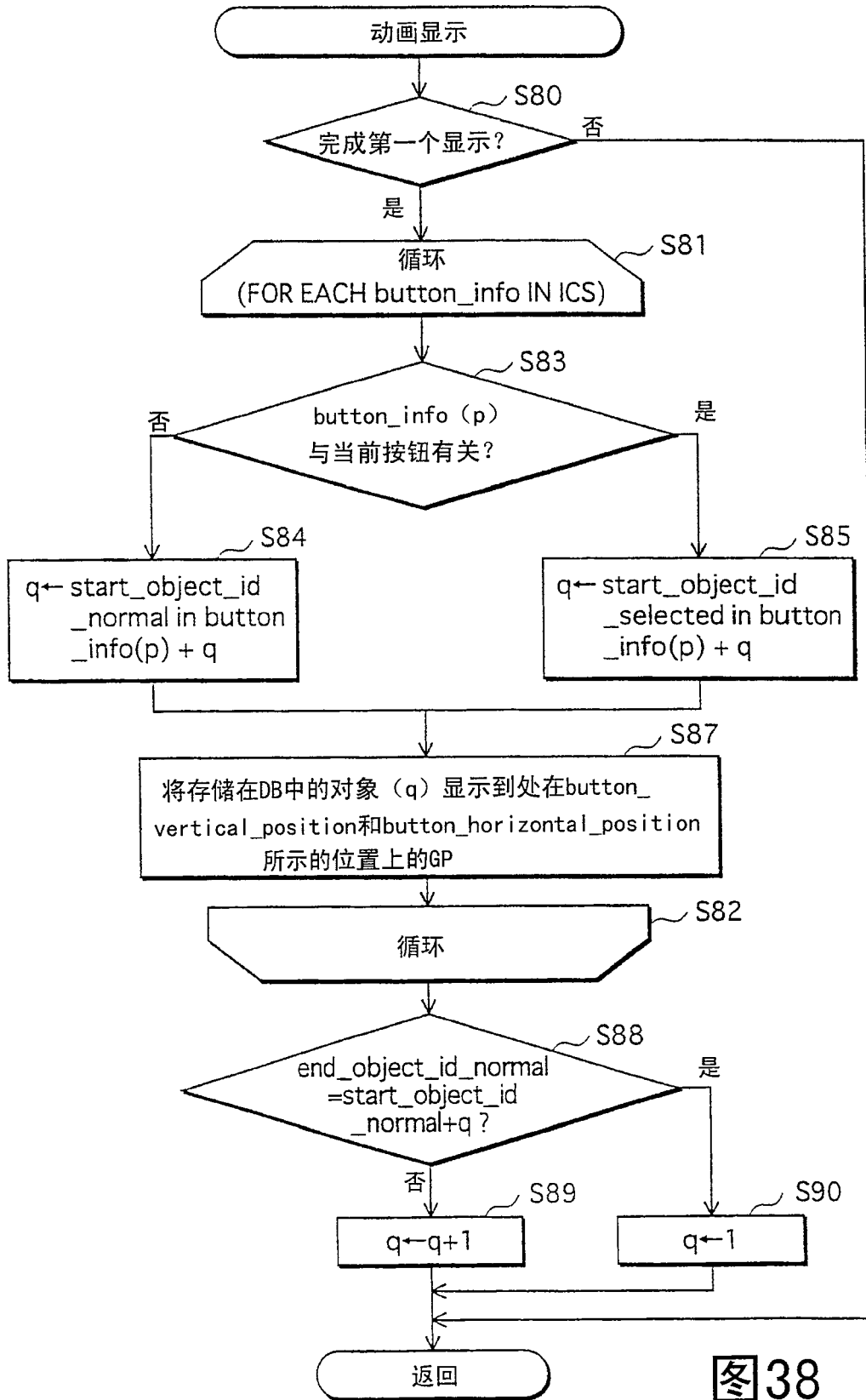


图38

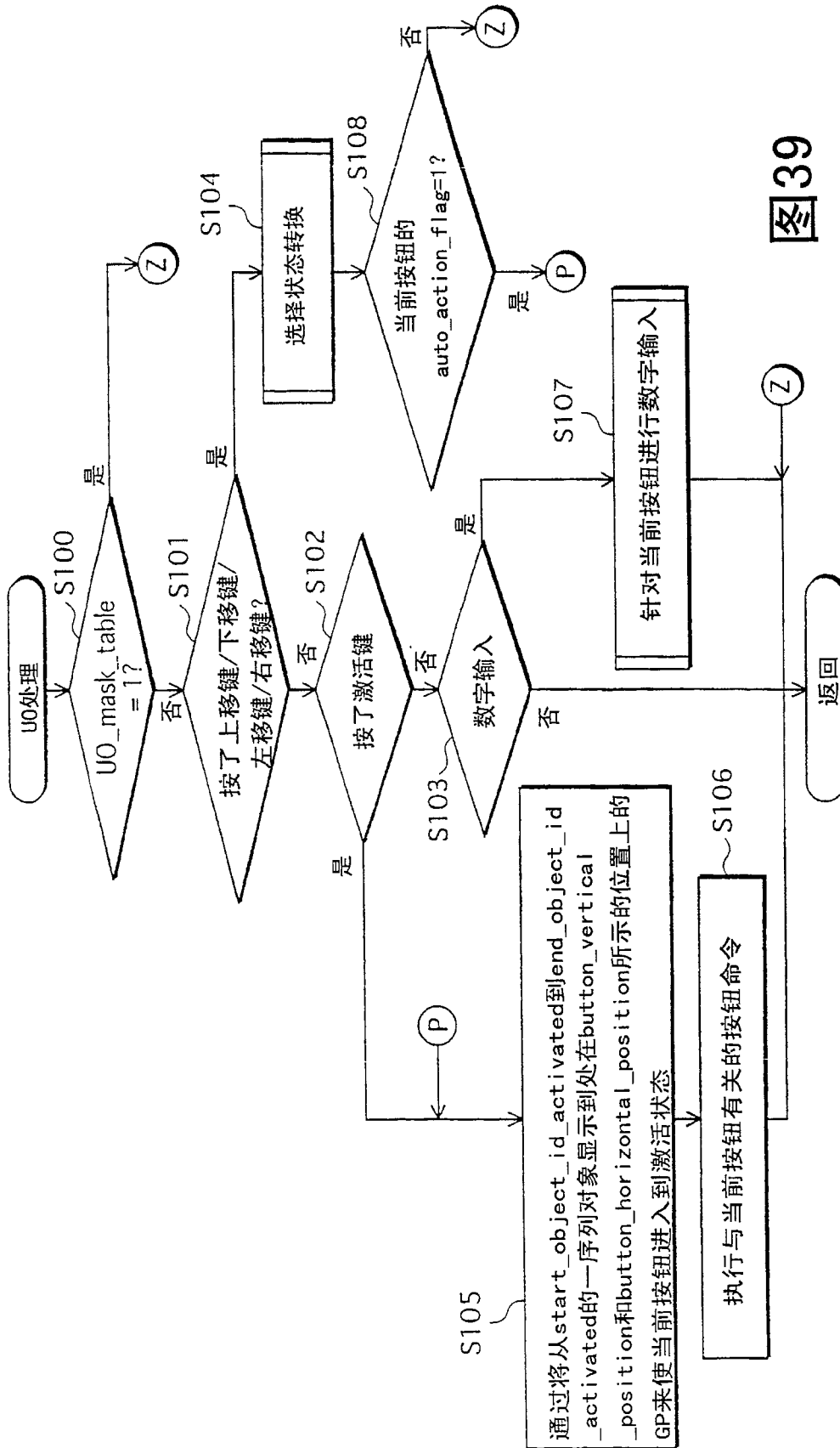


图39

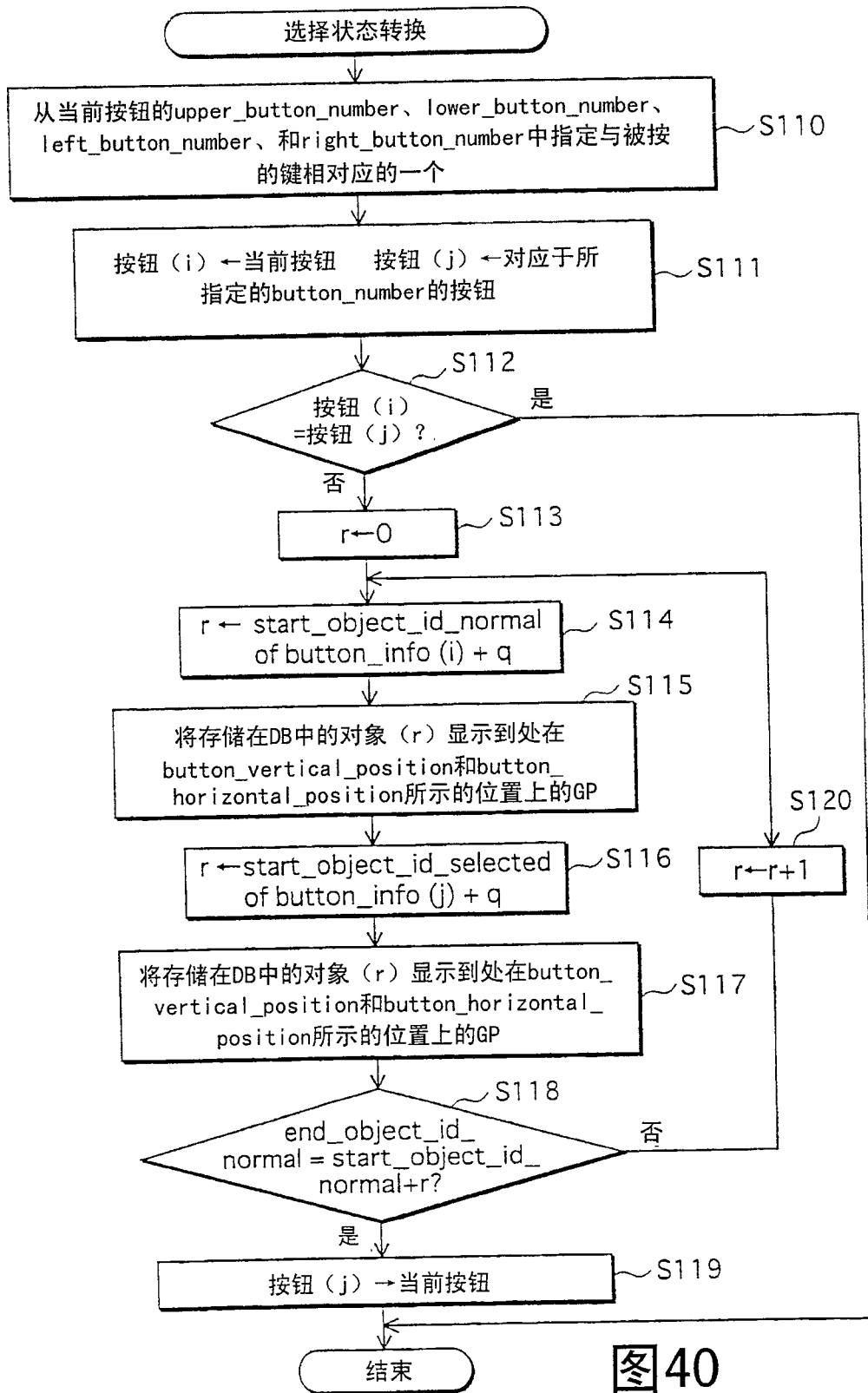


图40



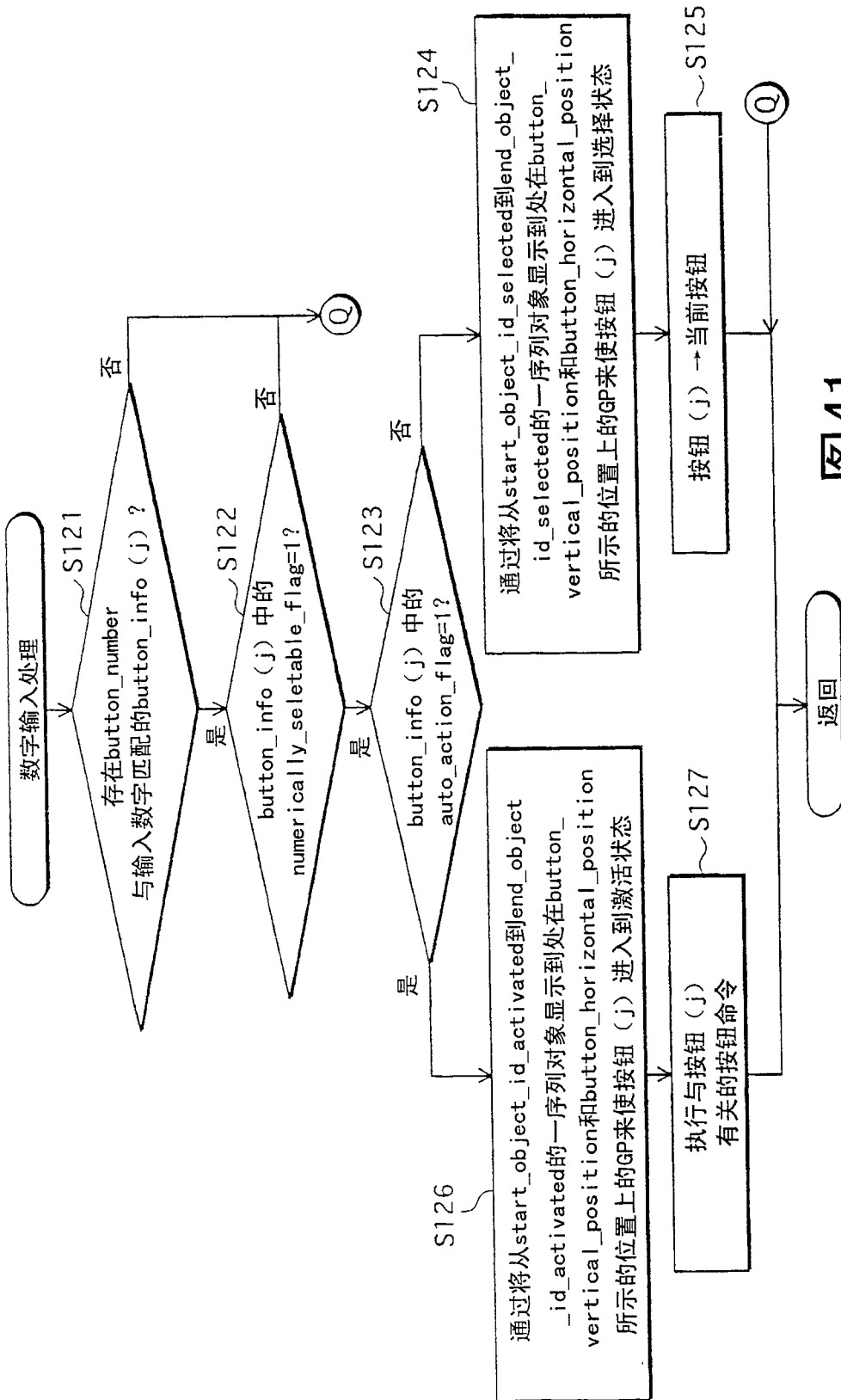


图41

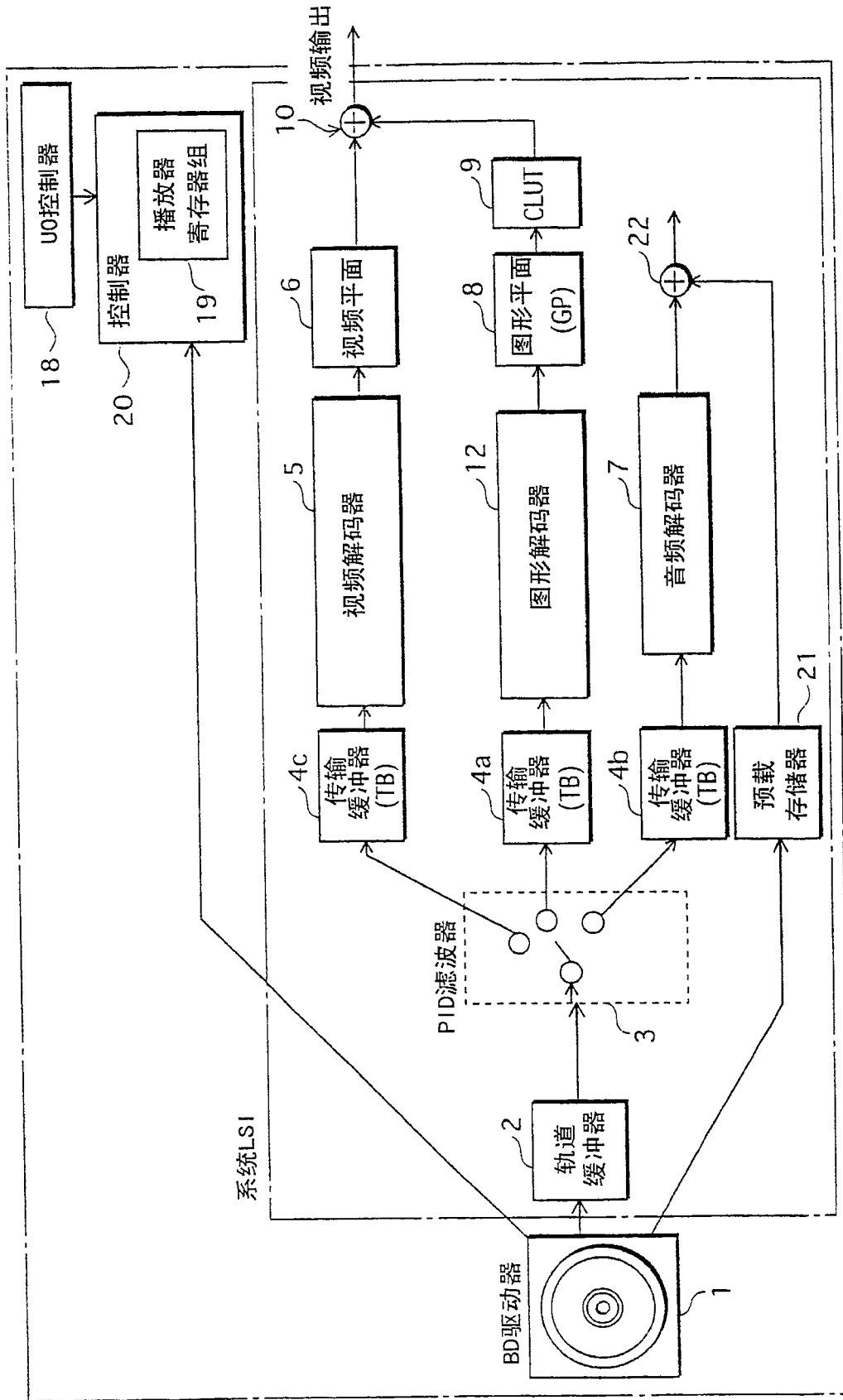


图42

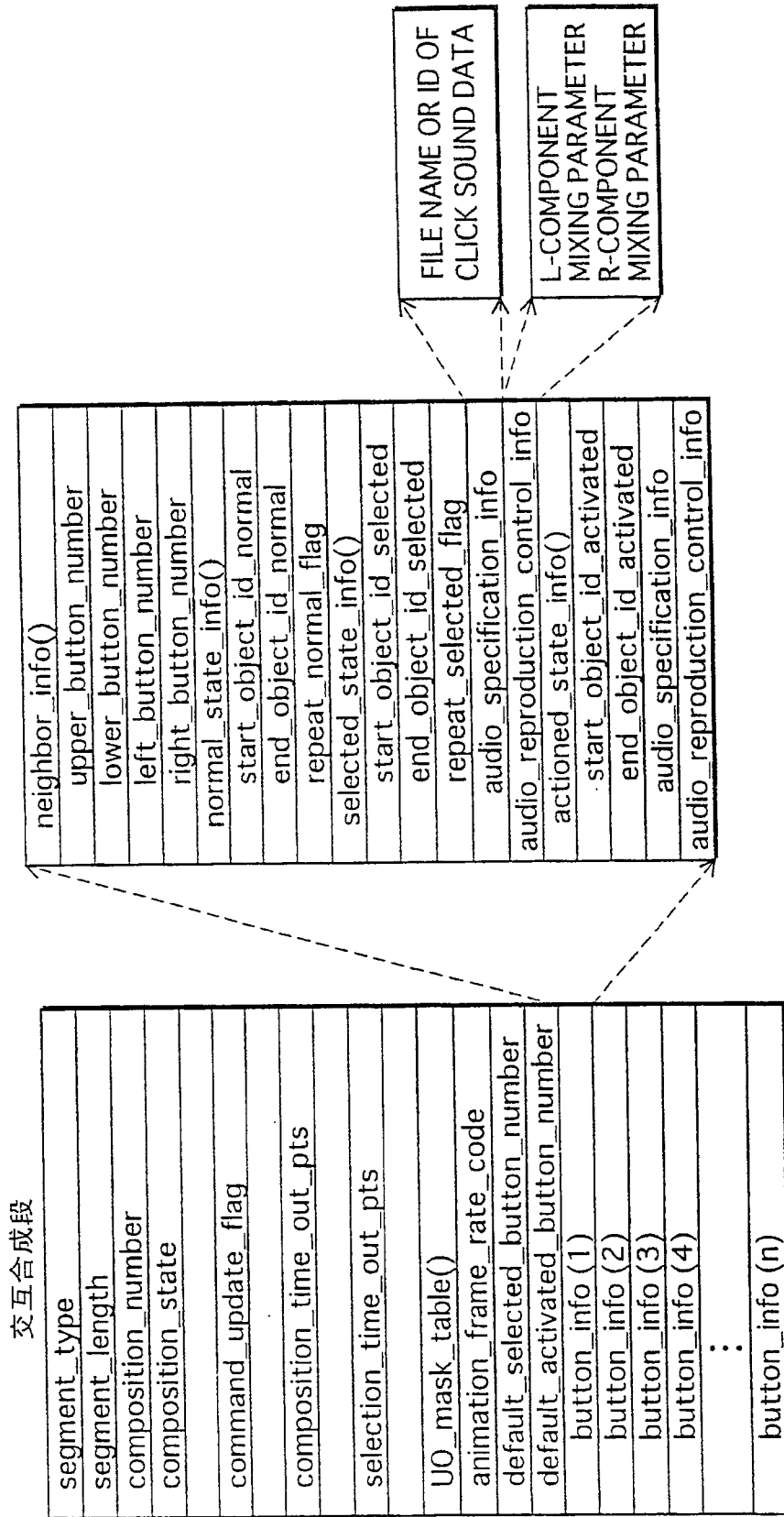


图 43

状态控制信息

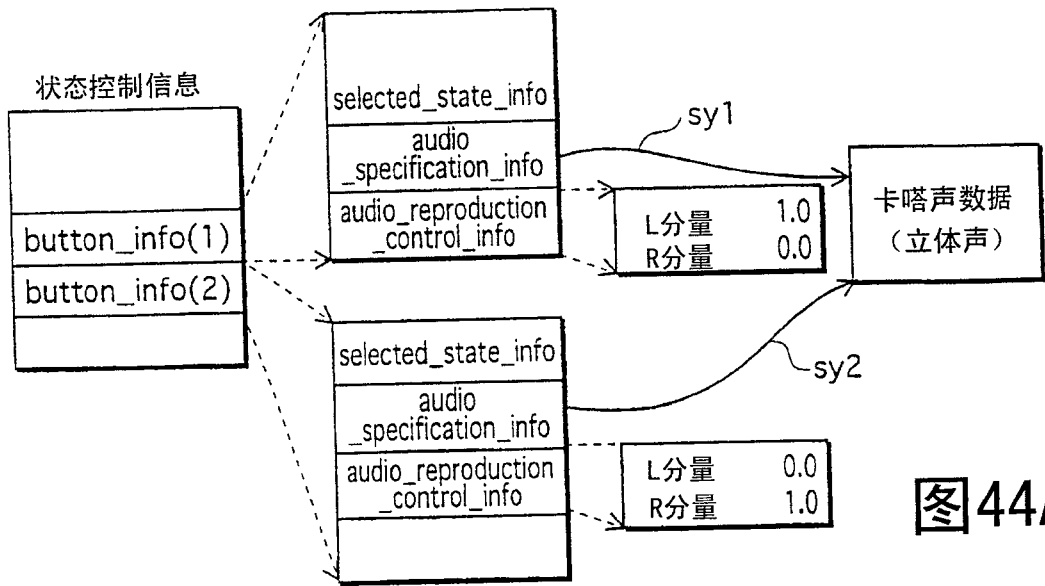


图44A

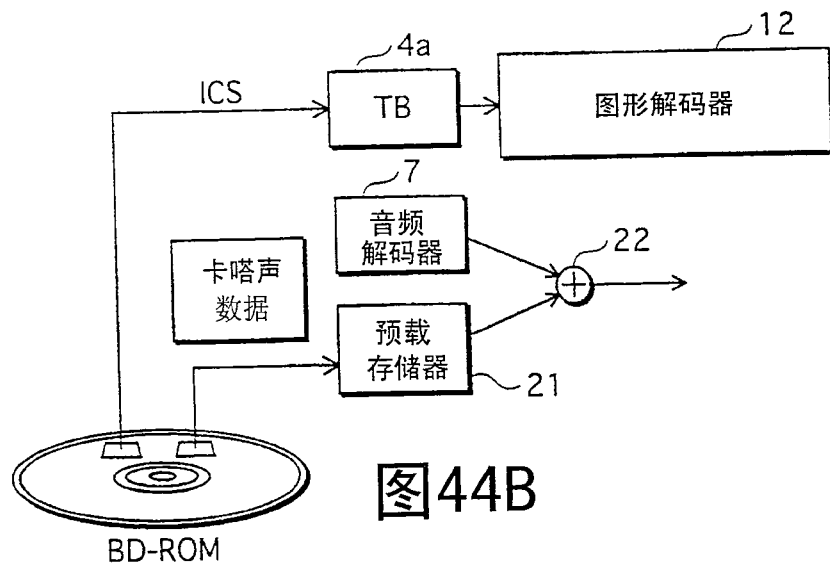


图44B

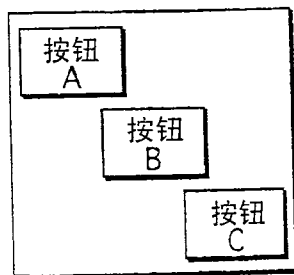


图44C

	按钮 A	按钮 B	按钮 C
L分量	1.0	0.1	
R分量		0.1	
中间分量		0.4	
左后分量		0.2	
右后分量		0.2	1.0

处于选择状态的按钮1

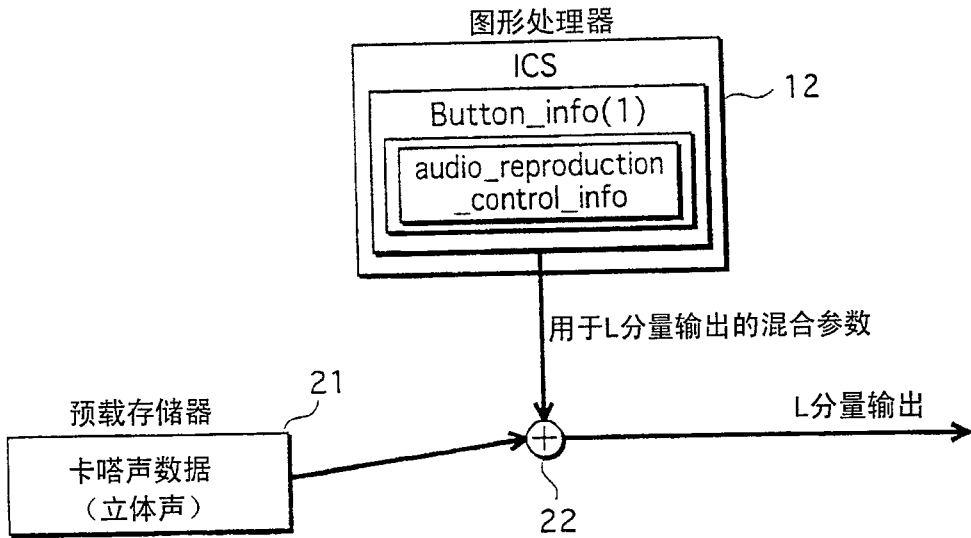


图45A

处于选择状态的按钮2

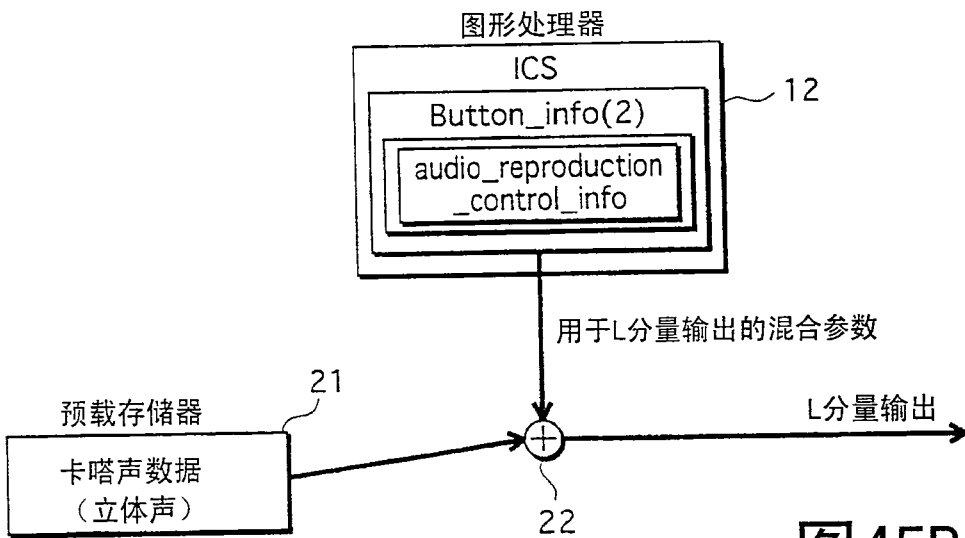


图45B

按钮 A	按钮 B	按钮 C		按钮 A	按钮 B	按钮 C
			L分量	1.0	0.5	0.0
			R分量	0.0	0.5	1.0

图45C

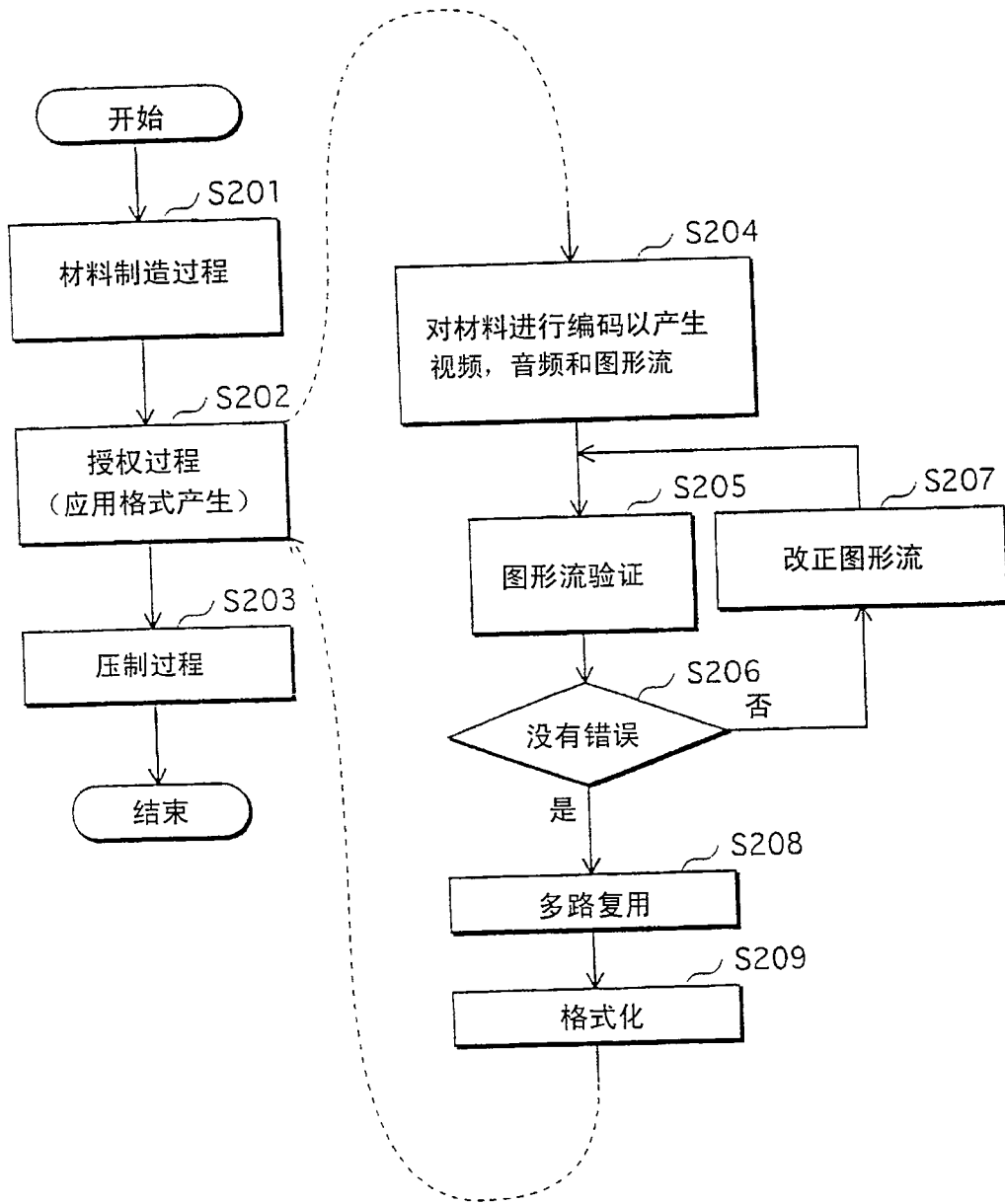


图46

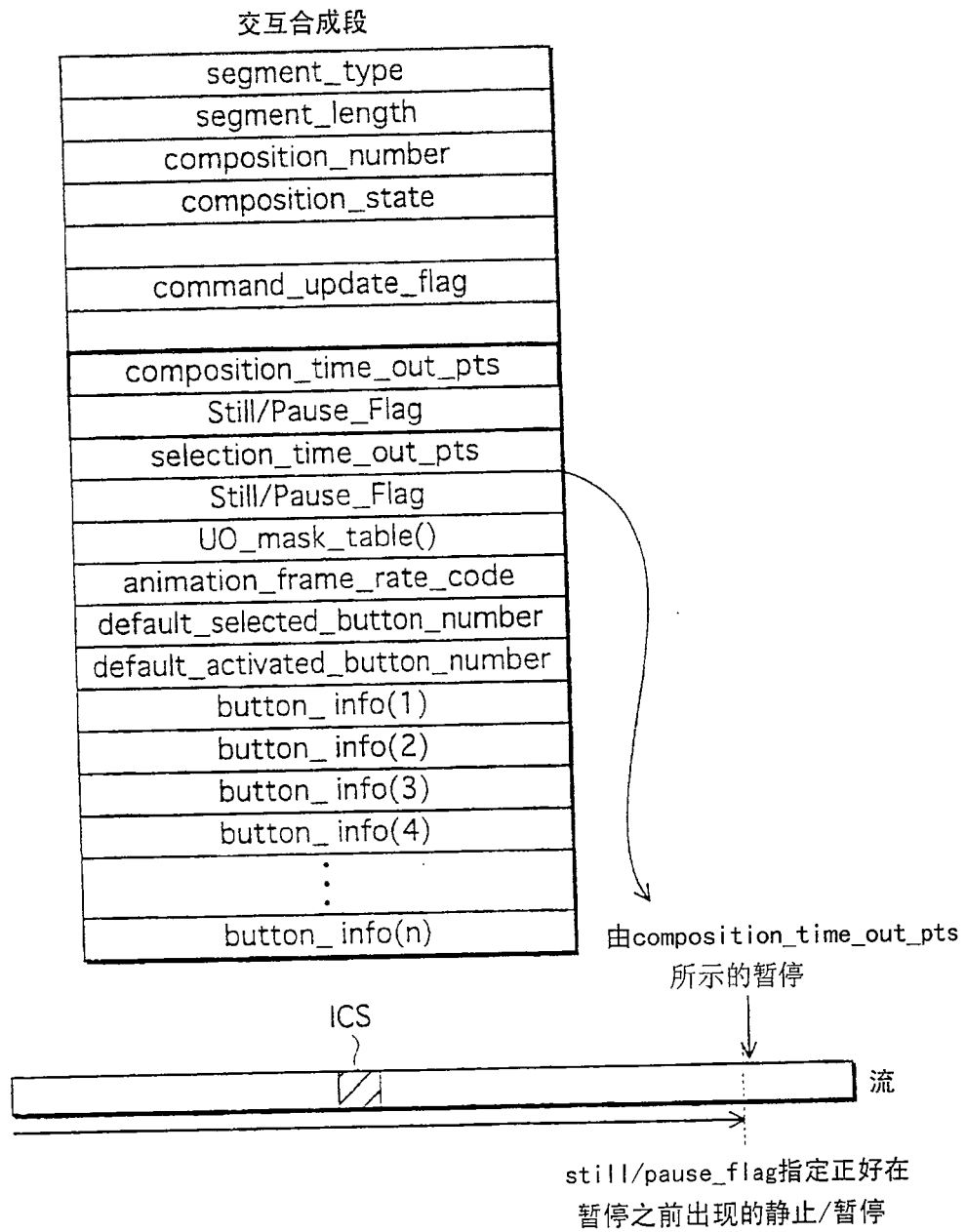


图47

交互成段

segment_type
segment_length
composition_number
composition_state
command_update_flag
composition_time_out_pts
selection_time_out_pts
upper_audio
lower_audio
left_audio
right_audio
activated_audio
UO_mask_table()
animation_frame_rate_code
default_selected_button_number
default_activated_button_number
button_info(1)
button_info(2)
button_info(3)
button_info(4)
::
button_info(n)

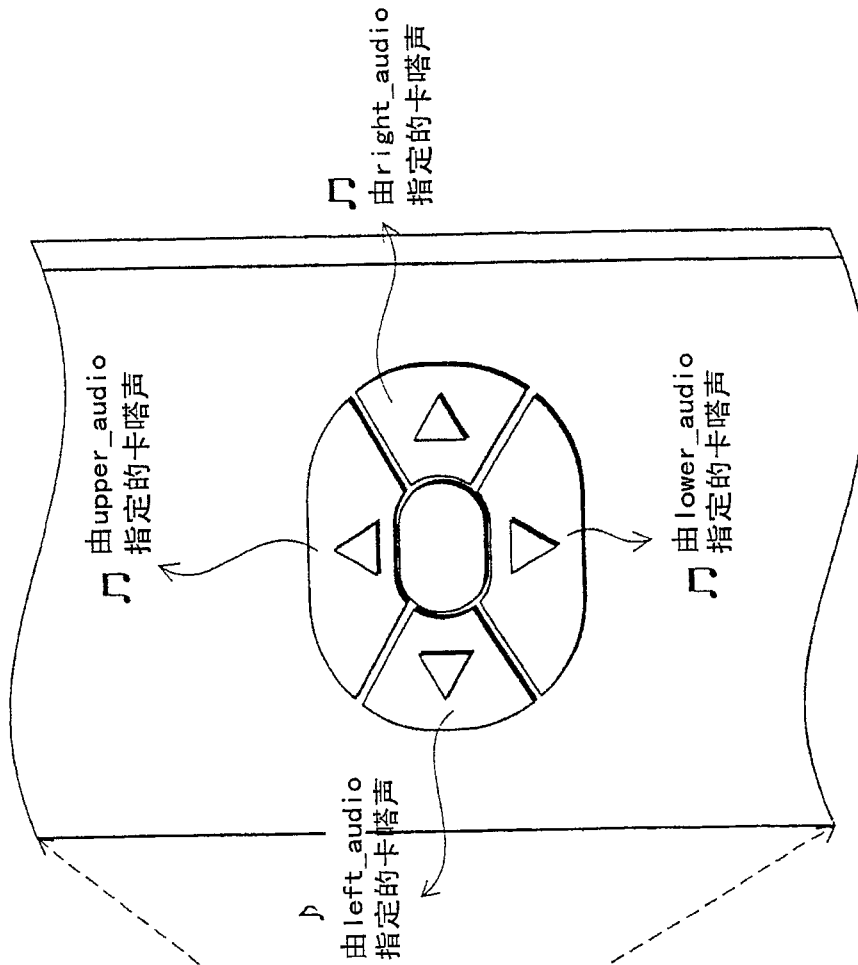


图48