



# [12] 发明专利申请公开说明书

[21] 申请号 200410009566.7

[43] 公开日 2005年3月23日

[11] 公开号 CN 1598767A

[22] 申请日 2004.9.15

[21] 申请号 200410009566.7

[71] 申请人 萧学文

地址 100083 北京市海淀区泉宗路2号万柳光大家园9号楼0938

[72] 发明人 黄小鹏 丁力 萧学文

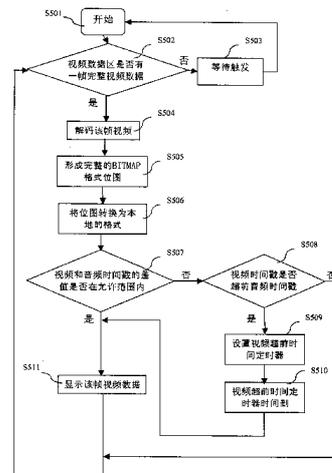
[74] 专利代理机构 北京律诚同业知识产权代理有限公司  
代理人 梁挥 祁建国

权利要求书3页 说明书9页 附图4页

[54] 发明名称 在 BREW 平台上同步播放音频视频的方法及系统

### [57] 摘要

本发明涉及一种在 BREW 平台上实现同步播放音频视频的方法，包括：设置一数据接收模块，用于与网络建立连接，设置数据接收触发事件，接收来自网络的数据，并存储到数据区；设置一音频播放模块，用于检查音频数据区的数据，播放音频数据，并且在播放音频数据的过程中，利用周期性事件触发定时更新音频播放的时间戳；设置一视频播放模块，用于检查视频数据区的数据、解码视频数据、将视频数据的格式转换为本地格式、调节视频和音频的同步和显示视频数据，还可以进一步调节视频帧间隔，从而实现在 BREW 环境中实现同步播放音频视频数据，并保证其流畅性和及时响应性。



ISSN 1008-4274

1. 一种在 BREW 平台上实现同步播放音频视频的方法，其特征在于，所述方法包括：

5        设置一数据接收模块，用于与网络建立连接，设置数据接收触发事件，接收来自网络的数据，并存储到数据区；

      设置一音频播放模块，用于检查音频数据区的数据，播放音频数据，并且在播放音频数据的过程中，利用周期性事件触发定时更新音频播放的时间戳；

10       设置一视频播放模块，用于检查视频数据区的数据、解码视频数据、将视频数据的格式转换为本地格式、调节视频和音频的同步、显示视频数据。

2. 根据权利要求 1 所述的在 BREW 平台上实现同步播放音频视频的方法，其特征在于，所述设置一音频播放模块中，当检查音频区，发现音频数据不够播放时，系统转到等待触发。

15       3. 根据权利要求 2 所述的在 BREW 平台上实现同步播放音频视频的方法，其特征在于，所述等待触发的方式是事件触发或定时器触发。

4. 根据权利要求 3 所述的在 BREW 平台上实现同步播放音频视频的方法，其特征在于，所述事件触发中的事件是数据流接收模块中的数据接收触发事件。

5. 根据权利要求 3 所述的在 BREW 平台上实现同步播放音频视频的方法，其特征在于，所述定时器触发的步骤，包括：

20       设置音频等待定时器；  
      音频等待定时器时间到。

6. 根据权利要求 1 所述的方法，其特征在于，所述音频播放的步骤包括：

      步骤 61、检查音频数据是否足够播放；音频数据足够时，就转到步骤 63；

      步骤 62、设置等待触发的条件，在触发条件满足时，返回步骤 61；

25       步骤 63、播放音频数据，并且在播放音频数据时，周期性事件触发更新音频播放时间戳。

      步骤 64、音频播放完成返回步骤 61。

30       7. 根据权利要求 6 所述的方法，其特征在于，所述的音频播放步骤还包括，在音频播放部分的开始设置音频播放时间的周期性事件触发，在播放音频数据的过程中，周期性事件触发定时更新音频播放的时间戳。

8. 根据权利要求1所述的方法，其特征在于，所述视频播放步骤包括：
- 步骤801、检查是否接收到一帧完整视频数据；接收到一帧完整视频数据时，转到步骤803；
- 步骤802、设置等待触发的条件，条件满足时，转到步骤801；
- 5 步骤803、解码接收到的视频数据；
- 步骤804、形成完整的BITMAP格式位图；
- 步骤805、将位图转换成为手机本地的格式；
- 步骤806、比较视频时间戳和音频时间戳的差值，如果差值在允许范围内，转到步骤810；
- 10 步骤807、比较视频时间戳是否超前音频时间戳，如果是，转到步骤809；
- 步骤808、返回步骤801；
- 步骤809、设置视频超前定时器，定时器时间到转到步骤806；
- 步骤810、显示该帧数据，返回步骤801。
9. 根据权利要求8所述的方法，其特征在于，在步骤806视频时间戳和音频时间戳的差值在范围内，还可以进一步调节视频数据的帧间隔。
- 15 10. 根据权利要求9所述的方法，其特征在于，所述调节视频数据的帧间隔的步骤包括：
- 步骤101、比较视频准确播放时间和当前时间，视频准确播放时间在后，执行步骤102，视频准确播放时间在前，跳转到步骤104；
- 20 步骤102、设置差值定时器；
- 步骤103、差值定时器时间到；
- 步骤104、帧间隔调节完毕。
11. 根据权利要求1所述的方法，其特征在于，所述调节视频和音频的同步包括有：
- 25 视频时间戳和音频时间戳的差值在允许的范围之内，调节完成；
- 视频时间戳和音频时间戳的差值在允许的范围之内，调节视频数据的帧间隔，调节完成；
- 视频时间戳和音频时间戳的差值不在允许的范围之内，并且视频时间戳超前音频时间戳，设置视频超前定时器，定时器时间到，调节完成；
- 30 视频时间戳和音频时间戳的差值不在允许的范围之内，并且视频时间戳落后音频

时间戳，调节完成；

12. 根据权利要求 8 所述的方法，其特征在于，在所述步骤 803、所述步骤 804、步骤 805、步骤 806 之后都可以增加短时定时器，在短时定时器时间到后再执行下一步骤。

5        13. 根据权利要求 8 所述的方法，其特征在于，步骤 802 所述等待触发的方式是事件触发或定时器触发。

14. 一种在 BREW 平台上实现同步播放音频视频的系统，其特征在于，所述系统包括数据接收模块、音频播放模块和视频播放模块；其中：

10        数据接收模块，用于实现与网络建立连接，设置数据接收触发事件，接收来自网络的数据，并存储到数据区；

      音频播放模块，用于实现检查音频数据区的数据，播放音频数据，在播放音频数据的过程中，利用周期性事件触发定时更新音频播放的时间戳；

      视频播放模块，用于实现检查视频数据区的数据、解码视频数据、转换视频数据的格式、调节视频和音频的同步和显示视频数据。

15        15. 根据权利要求 14 所述的在 BREW 平台上实现同步播放音频视频的系统，其特征在于，所述的触发条件包括事件触发或者定时器触发。

## 在 BREW 平台上同步播放音频视频的方法及系统

### 5 技术领域

本发明涉及多媒体通讯技术领域，尤其涉及在 BREW 平台手机上实现同步播放音频视频的方法。

### 背景技术

10 BREW 是 Binary Runtime Environment for Wireless 的缩写，是为无线数据应用程序开发和执行提供的通用平台。

BREW 运行时环境是一个较薄的软件接口层，位于芯片系统软件 and 应用程序间的无线设备上，使最终用户可以无线下载程序并在启用 BREW 的设备上运行。BREW 具有较高的闪存和 RAM 使用效率（即，它所占闪存较少），而且在应用程序运行时动态分配 RAM。BREW 提供一个高效、低成本、可扩展和熟悉的应用程序执行环境（AEE），一套高度集成的芯片集功能、系统软件以及用于实现高级因特网功能和多媒体功能的插件，着重开发可无缝植入任何实际手持设备的应用程序。BREW 提供的功能环境就如同 PC 机上的操作系统一样，可以通过服务提供商下载指定类型的应用程序或游戏来使用。同时，通过 BREW 接口功能，供应商可以提供成套完整的资讯、商务、娱乐功能。BREW 主要应用在移动通信领域，BREW 类似一个开放免费的 PC 操作系统，厂商可以在这个平台上设计各项应用。

20 由于 BREW 环境不支持多线程技术和本身平台的特殊性，在 windows 或 Linux 环境下实现多媒体音频视频数据同步播放的方式无法在 BREW 环境中实现，为了实现多媒体的同步播放，必须使用事件技术。

25

### 发明内容

本发明所要解决的技术问题在于提供一种在 BREW 平台上同步播放音频视频的方法及系统，实现在 BREW 环境中实现同步播放音频视频数据，并保证其流畅性和及时响应性。

30 为了实现上述目的，本发明提供了一种在 BREW 平台上实现同步播放音频视频的

方法，包括：

设置一数据接收模块，用于与网络建立连接，设置数据接收触发事件，接收来自网络的数据，并存储到数据区；

5 设置一音频播放模块，用于检查音频数据区的数据，播放音频数据，并且在播放音频数据的过程中，利用周期性事件触发定时更新音频播放的时间戳；

设置一视频播放模块，用于检查视频数据区的数据、解码视频数据、将视频数据的格式转换为本地格式、调节视频和音频的同步、显示视频数据。

所述设置一音频播放模块中，当检查音频区，发现音频数据不够播放时，系统转到等待触发。

10 所述等待触发的方式是事件触发或定时器触发。

所述事件触发中的事件是数据流接收模块中的数据接收触发事件。

所述定时器触发的步骤，包括：

设置音频等待定时器；

音频等待定时器时间到。

15 所述音频播放的步骤包括：

步骤 61、检查音频数据是否足够播放；音频数据足够时，就转到步骤 63；

步骤 61、设置等待触发的条件，在触发条件满足时，返回步骤 61；

步骤 63、播放音频数据，并且在播放音频数据时，周期性事件触发更新音频播放时间戳。

20 步骤 64、音频播放完成返回步骤 61。

所述的音频播放步骤还包括，在音频播放部分的开始设置音频播放时间的周期性事件触发，在播放音频数据的过程中，周期性事件触发定时更新音频播放的时间戳。

所述视频播放步骤包括：

25 步骤 801、检查是否接收到一帧完整视频数据；接收到一帧完整视频数据时，转到步骤 803；

步骤 802、设置等待触发的条件，条件满足时，转到步骤 801；

步骤 803、解码接收到的视频数据；

步骤 804、形成完整的 BITMAP 格式位图；

步骤 805、将位图转换成为手机本地的格式；

30 步骤 806、比较视频时间戳和音频时间戳的差值，如果差值在允许范围内，转到

步骤 810;

步骤807、比较视频时间戳是否超前音频时间戳，如果是，转到步骤809；

步骤 808、返回步骤 801；

步骤 809、设置视频超前定时器，定时器时间到转到步骤 806；

5 步骤 810、显示该帧数据，返回步骤 801。

在步骤 806 视频时间戳和音频时间戳的差值在范围内，还可以进一步调节视频数据的帧间隔。

所述调节视频数据的帧间隔的步骤包括：

步骤 101、比较视频准确播放时间和当前时间，视频准确播放时间在后，执行步

10 骤 102，视频准确播放时间在前，跳转到步骤 104；

步骤 102、设置差值定时器；

步骤 103、差值定时器时间到；

步骤 104、帧间隔调节完毕。

所述调节视频和音频的同步包括有：

15 视频时间戳和音频时间戳的差值在允许的范围之内，调节完成；

视频时间戳和音频时间戳的差值在允许的范围之内，调节视频数据的帧间隔，调节完成；

视频时间戳和音频时间戳的差值不在允许的范围之内，并且视频时间戳超前音频时间戳，设置视频超前定时器，定时器时间到，调节完成；

20 视频时间戳和音频时间戳的差值不在允许的范围之内，并且视频时间戳落后音频时间戳，调节完成；

在所述步骤 803、所述步骤 804、步骤 805、步骤 806 之后都可以增加短时定时器，在短时定时器时间到后再执行下一步骤。

步骤 802 所述等待触发的方式是事件触发或定时器触发。

25 本发明还公开了一种在 BREW 平台上实现同步播放音频视频的系统，所述系统包括数据接收模块、音频播放模块和视频播放模块；其中：

数据接收模块，用于实现与网络建立连接，设置数据接收触发事件，接收来自网络的数据，并存储到数据区；

30 音频播放模块，用于实现检查音频数据区的数据，播放音频数据，在播放音频数据的过程中，利用周期性事件触发定时更新音频播放的时间戳；

视频播放模块,用于实现检查视频数据区的数据、解码视频数据、转换视频数据的格式、调节视频和音频的同步和显示视频数据。

所述的触发条件包括事件触发或者定时器触发。

本发明实现在 BREW 环境中实现同步播放音频视频数据,并保证其流畅性和及时  
5 响应性。

### 附图说明

图 1 所示为设置网络数据接收触发事件的流程图;

图 2 所示为数据流接收的流程图;

10 图 3 所示为音频数据播放的流程图;

图 4 所示为音频定时器触发的流程图;

图 5 所示为视频数据播放的流程图;

图 6 所示为视频定时器触发的流程图;

图 7 所示为调节视频显示帧间隔的流程图。

15

### 具体实施方式

以下结合附图和具体实施例对本发明进行详细描述,但不作为对本发明的限定。

由于 BREW 平台不支持多线程技术,在 BREW 平台要实现音视频的同步性要使用基于事件和时间触发的技术,实现如下三大模块的功能:数据流接收模块、音频播放模  
20 块和视频播放模块。

数据流接收模块,主要负责接收来自网络的数据。其功能包括:设置网络数据接收触发事件;和在有数据到达后,系统产生数据接收触发事件,随后开始接收数据。

图 1 为数据流接收模块设置网络数据接收触发事件的流程图。

在步骤 S101 中,通过 BREW 平台与服务器相连接。

25 在步骤 S102 中,设置数据接收触发事件。这可以通过调用函数 `ISOCKET_Readable` 来设置。

图 2 为在有数据到达 BREW 平台后,接收数据流的流程图。

在步骤 S201 中,有数据到达 BREW 平台。

在步骤 S202 中,系统产生数据接收触发事件。

30 在步骤 S203 中,开始从网络接收流媒体数据。

在步骤 S204 中，将接收的数据流中的音频数据放到音频数据区中，视频数据放到视频数据区中。

步骤 S205 为音频播放和视频播放的开始点。

当有新的数据到达时，BREW 平台会自动产生新的数据接收触发事件，重复执行如图 2 所示的数据接收步骤。

音频播放模块，负责播放音频数据。

图 3 为音频数据播放的流程图。

在图 3 所示的流程图中步骤 S301，即音频数据播放流程的开始点，可以是图 2 所示数据接收流程图步骤 S205 的位置 A。

10 在步骤 S302 中，检查音频数据区的数据是否够播放。是否足够播放的条件是判断音频数据区的音频数据有多少，或音频数据区的数据能够播放多长时间。这个值用户可以自行设定，如够播放 3 秒钟。如果音频数据区的音频数据不够播放，那么跳转到步骤 S303；如果数据够播放跳转到步骤 S304。

15 在步骤 S303 中，系统等待触发事件的到达。当触发条件满足后，跳转到步骤 S302，继续检查音频数据区的数据是否够播放。触发的条件可以是事件触发，如数据流接收模块中的数据接收触发事件，也可以是定时器触发。图 4 所示为音频定时器触发的流程图。

在步骤 S304 中，更新音频时间戳。

在步骤 S306 中，播放音频数据。

20 在步骤 S307 中，播放完音频数据后，跳转到步骤 S302 继续检查音频数据是否够播放。

在图 3 音频播放流程图的开始位置，可以设置音频播放时间的周期性事件触发，在播放音频数据的过程中，周期性触发事件在触发条件满足后 (S305)，更新音频播放的时间戳 (S304)。这个时间戳被视频播放模块用作播放时间的参考。例如可以设置  
25 音频播放中每播放 500 毫秒的音频数据就触发一个事件。如果没有设置音频播放时间的周期性事件触发，则系统缺省每播放 1000 毫秒的音频数据就触发一个事件。在音频播放的过程中，音频播放的时间戳由于周期性的事件触发被定期的更新。例如，当前时间戳为 1000 毫秒，过 500 毫秒的音频数据就触发一个事件，事件触发后时间戳更新为 1500 毫秒。

30 图 4 为音频定时器触发的流程图。在图 4 中，设置音频等待定时器 (S401)，在

音频等待定时器时间到 (S402) 后, 再去检查音频数据区中是否有足够的数 据 (S302)。

视频播放模块, 负责播放视频数据。

图 5 为视频数据播放的流程图。

在图 5 所示的流程图中步骤 S501, 即视频数据播放流程的开始点, 可以是图 2  
5 所示数据接收流程图步骤 S205 的位置 A。

在步骤 S502, 检查视频数据区的数据是否够一帧完整的视频数据, 如果视频区的  
数据不够一帧完整的视频数据, 跳转到步骤 S503 等待触发; 如果视频区的数据够一  
帧完整的视频数据, 执行步骤 S504。

在步骤 S503, 系统等待触发事件的到达。当触发条件满足后, 转到 S502 继续检  
10 查视频数据区的数据是否够一帧完整的视频数据。在步骤 S503 中触发的条件可以是  
事件触发, 如数据流接收模块中的数据接收触发事件, 也可以是定时器触发。如图 6  
所示为视频定时器触发的流程图。

在步骤 S504 中, 视频数据区的数据够一帧完整的视频数据, 解码这一帧视频数  
据。

15 在步骤 S505 中, 将解码出来的数据形成一幅完整的 BITMAP 格式位图。

在步骤 S506 中, 将步骤 S505 中形成的 BITMAP 格式位图转换为手机本地的格式。

在步骤 S507 中, 计算视频时间戳和音频时间戳的差值, 判断音频当前播放的时  
间戳和该视频帧的时间戳的差值 (也称上述差值为同步误差) 是否在允许的范围内。  
例如视频时间戳和音频时间戳差值的允许范围可以设置为正负 100 毫秒之间。

20 当视频时间戳和音频时间戳的差值不在允许的范围内时, 转到步骤 S508, 判断视  
频时间戳是否超前音频时间戳。视频时间戳超前音频时间戳时, 根据视频时间戳和音  
频时间戳的差值设置视频超前定时器 (S509), 视频超前定时器时间到时 (S510),  
从而保证同步误差在允许的范围内; 视频时间戳落后音频时间戳时, 显示该帧视频数  
据已经没有意义, 丢弃该视频数据, 然后转到 S502 继续检查视频数据区是否有一帧  
25 完整的视频数据。

当视频时间戳和音频时间戳的差值在允许的范围内时, 显示该帧视频数据 (S511)  
然后转到步骤 S502 继续检查视频区是否有一帧完整的视频数据。

在如图 6 中, 首先设置视频等待定时器 (S601), 在视频等待定时器时间到 (S602)  
后, 检查视频数据区中是否有一帧完整的视频数据 (S502)。

30 在步骤 S504、步骤 S505、步骤 S506、步骤 S507 之后可以增加短时定时器, 在短

时定时器时间到后再执行下面的步骤。例如在步骤 S504，即解码该帧视频后，设置 10 毫秒的短时定时器，在 10 毫秒时间到后再执行步骤 S505 形成完整的 BITMAP 格式位图。这样可以使手机在播放视频音频流的时候依然可以及时相应用户的按键请求等。

- 5 为了使视频播放更加流畅，不仅要求音频数据和视频数据的同步误差在允许的范围内，还要求任何两帧视频之间的时间间隔基本一致。为了达到上述目的，当同步误差在允许范围内时，还可以进一步调节帧间隔显示该帧视频数据。

在音频视频同步的情况下，视频准确显示的时间等于上一帧显示的时间加上帧间隔时间。帧间隔时间等于帧率的倒数。例如每秒显示 5 帧视频数据，帧间隔时间是 200 10 毫秒。在实际播放中，很能难保证两帧显示之间的时间差绝对准确地等于帧间隔时间，因此，可以设定视频帧间隔时间误差（称为帧间隔误差）允许范围。当两帧之间的帧间隔误差在允许的范围内时，认为两帧视频之间的时间间隔基本一致，即两帧视频的播放是流畅的。例如设置帧间隔误差允许范围为  $\pm 10\%$ ，即当两帧显示的时间差在 200 毫秒时，允许的误差范围为正负 20 毫秒。

- 15 图 7 为调节视频显示帧间隔的流程图。

在图 7 中，首先视频时间戳和音频时间戳的差值在允许范围内（S701）；判断当前时间是否超前于视频准确播放时间（S702），当前时间落后于视频准确播放时间时，直接显示此帧视频数据（S705）；当前时间超前视频准确播放时间时，设置差值定时器（S703），差值定时器时间到（S704），显示此帧视频数据（S705）。显示此帧视 20 频数据（S705）后，记录播放时间戳（S706）。在图 7 中，步骤 S702、S703、S704、S705、S706 共同构成帧间隔调节模块 B。

在步骤 S703 中设置的差值定时器的差值满足如下两个条件：

条件一、同步误差在允许范围内；和

条件二、帧间隔误差在允许范围内。

- 25 条件一可以保证视频数据和音频数据同步；条件二可以保证视频数据帧间隔基本一致，从而得到流畅的视频效果。

在视频播放的过程中，若选择调节帧间隔，那么可以用图 7 中的帧间隔调节模块 B 取代图 5 中的步骤 S511，并且当视频时间戳落后音频时间戳时，转到步骤 S706 之前，记录视频播放时间戳，从而在 BREW 环境中实现同步播放音频视频数据。

- 30 下面是在以 BREW 平台为基础的手机上，应用本发明的方法播放音频和视频的另

一个实施例。

首先手机与服务器相连接，然后调用系统函数，设置网络数据接收触发事件。

当来自服务器的数据流到达手机的 BREW 终端后，系统会产生数据接收触发事件，随后开始接收数据。系统分配两个数据区，一个为音频数据区，一个为视频数据区。

- 5 对到达 BREW 终端的数据流进行分析，将视频的数据放到视频数据区，音频数据放到音频数据区。

例如设置足够播放的条件为音频数据区的数据能连续播放 5 秒钟。检查音频数据区的数据是否足够播放。检查结果发现音频数据区的数据仅仅能播放 3 秒钟，设置音频等待定时器为 2 秒钟。在 2 秒钟音频定时器时间到后，再次检查音频数据区的数据

- 10 是否足够。检查结果音频数据区的数据能够播放 5 秒。

播放音频数据区的数据。播放完成，则继续检查音频数据区是否有足够的音频数据。

在音频播放的开始设置音频播放时间的周期性事件触发，例如每播放 500 毫秒的音频数据就触发一个事件。在播放音频数据的过程中，音频播放的时间戳由于周期性的事件触发被定期的更新，如现在的时间戳为 1000，经过 500 毫秒的音频数据就触发

- 15 一个事件，触发后，时间戳更新为 1500，再过 500 毫秒，时间戳更新为 2000。

视频播放时，先检查视频数据区的数据是否够一帧完整的视频数据，如果不够，等待触发。触发条件在此选用定时器触发。例如设置视频等待定时器 1 秒钟。在 1 秒钟后，再去检查数据是否足够。

- 20 视频数据区的数据够一帧完整的视频数据后，就调用解码器解码这一帧视频。解码出来的数据通过色彩空间转换变为 RGB24 位的色彩空间，在内存区组织形成一幅完整的 BITMAP 格式位图内容，再将该位图转换为手机本地的格式。

随后要比较同步误差是否在允许的范围之内。例如同步误差的允许范围设置为正负 100 毫秒内。若音频的时间戳是 2000，视频的时间戳是 2050，那么就可以直接显示

- 25 该帧视频数据。

如果同步误差不在允许的范围之内，就要判断视频时间戳是否超前音频时间戳。当视频时间戳超前音频时间戳时，例如，音频的时间戳是 2000，视频的时间戳是 2150，即视频超前  $(2150 - 2000) = 150$  毫秒，此时设置 150 毫秒的定时器，在该定时器时间到后，再显示该帧视频。

- 30 当视频时间戳落后音频时间戳时，例如，音频的时间戳是 2000，视频的时间戳是

1850, 在这种情况下, 显示该帧视频数据已经没有意义, 因此丢弃该帧数据。

为了使视频播放更加地流畅, 需要进一步调节帧间隔。

例如, 在BREW终端上设置每秒5帧视频图像, 帧间隔时间是200毫秒, 设定帧间隔误差范围 $\pm 10\%$ , 即允许两帧显示的时间差在200毫秒正负20毫秒之间。

- 5        当第N帧视频播放的时间是2050, 则第(N+1)帧视频的准确播放时间应该是2250。若当前视频的时间戳是2230, 那么视频帧超前20毫秒, 帧间隔误差在误差允许的范围( $\pm 20$  毫秒)之内, 因此可以直接显示当前帧的视频数据。或当前视频的时间戳是2220, 那么视频帧超前30毫秒, 帧间隔误差超过了误差允许的范围, 则需要设置差值定时器的值为30毫秒, 以调节帧间隔, 从而保证播放的视频图像流畅。在差值定时器时间到后, 显示该帧视频。若当前视频的时间戳是2260, 表示已经落后于准确显示时间, 则直接显示该帧视频。显示完该帧视频数据后, 记录视频时间戳的值。
- 10

在解码该帧视频、形成完整的BITMAP格式位图、将位图转换为手机本地的格式和显示该帧视频这四个步骤之后增加10毫秒的定时器, 在10毫秒到后再继续执行下面的步骤。这样可以使手机在播放视频音频流的时候依然可以及时响应用户的按键请求

15 等。

在处理完一帧数据之后, 继续检查视频数据区的情况。

当然, 本发明还可有其他多种实施例, 在不背离本发明精神及其实质的情况下, 熟悉本领域的技术人员当可根据本发明作出各种相应的改变和变形, 但这些相应的改变和变形都应属于本发明所附的权利要求的保护范围。

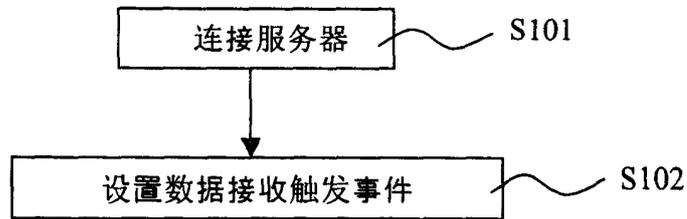


图 1

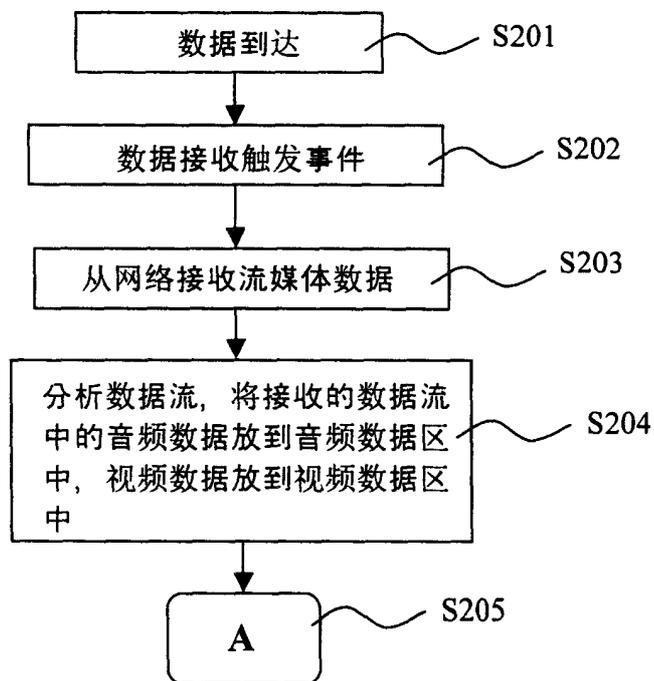


图 2

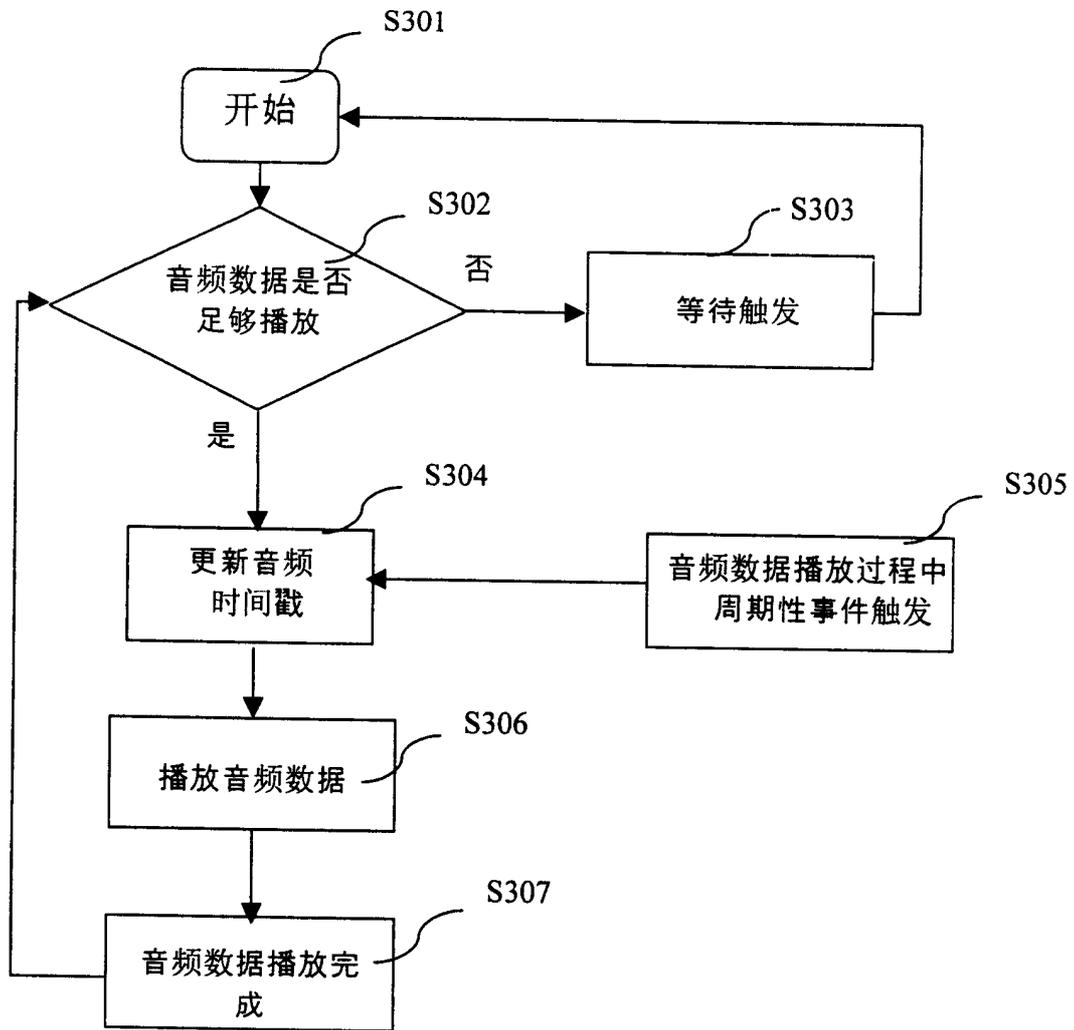


图 3

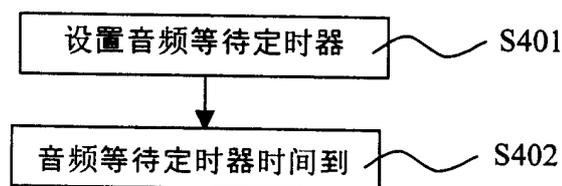


图 4

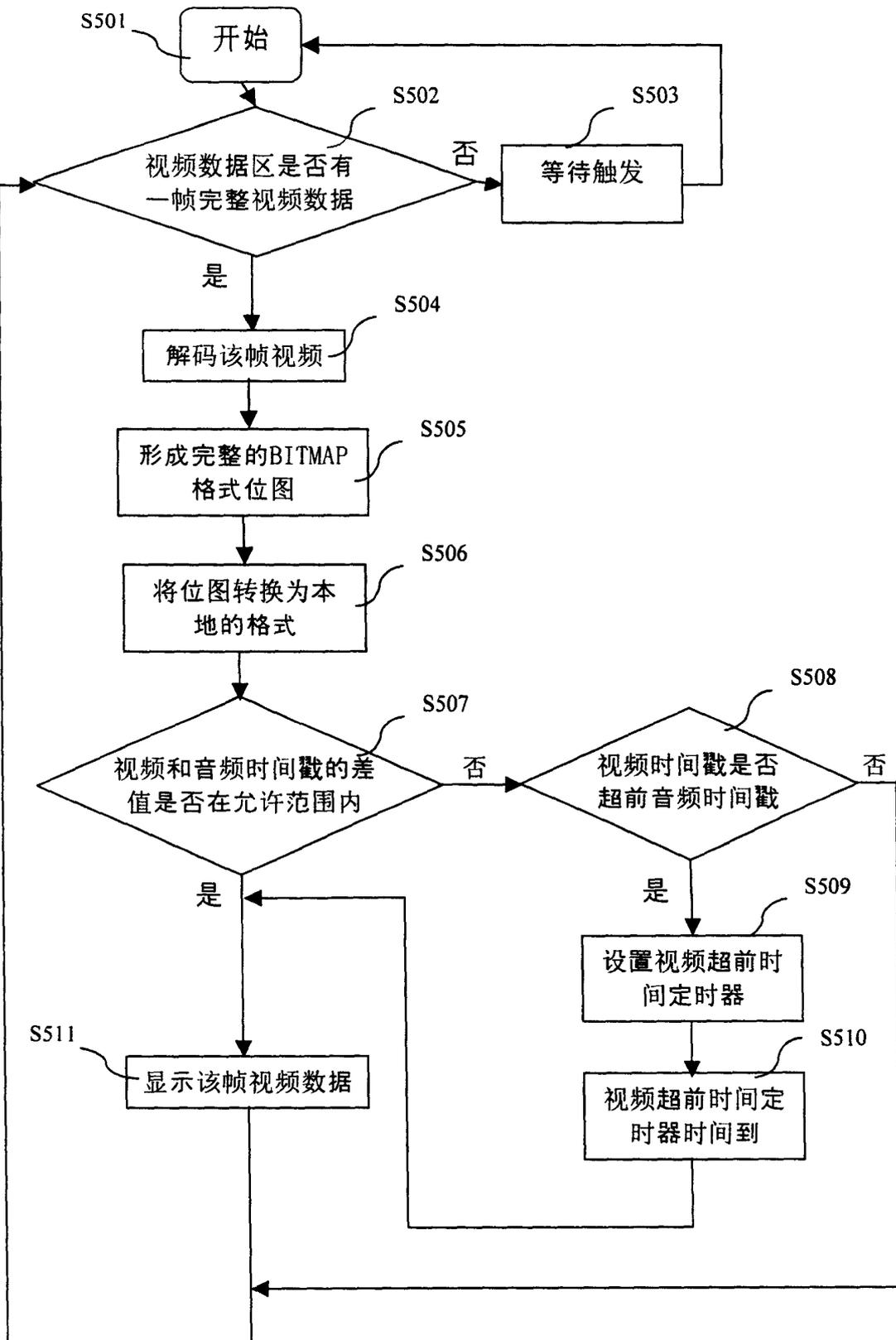


图 5

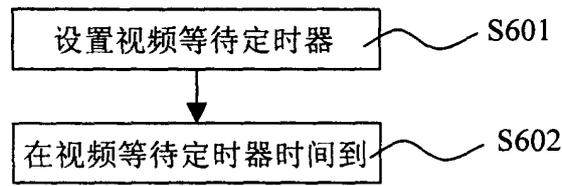


图 6

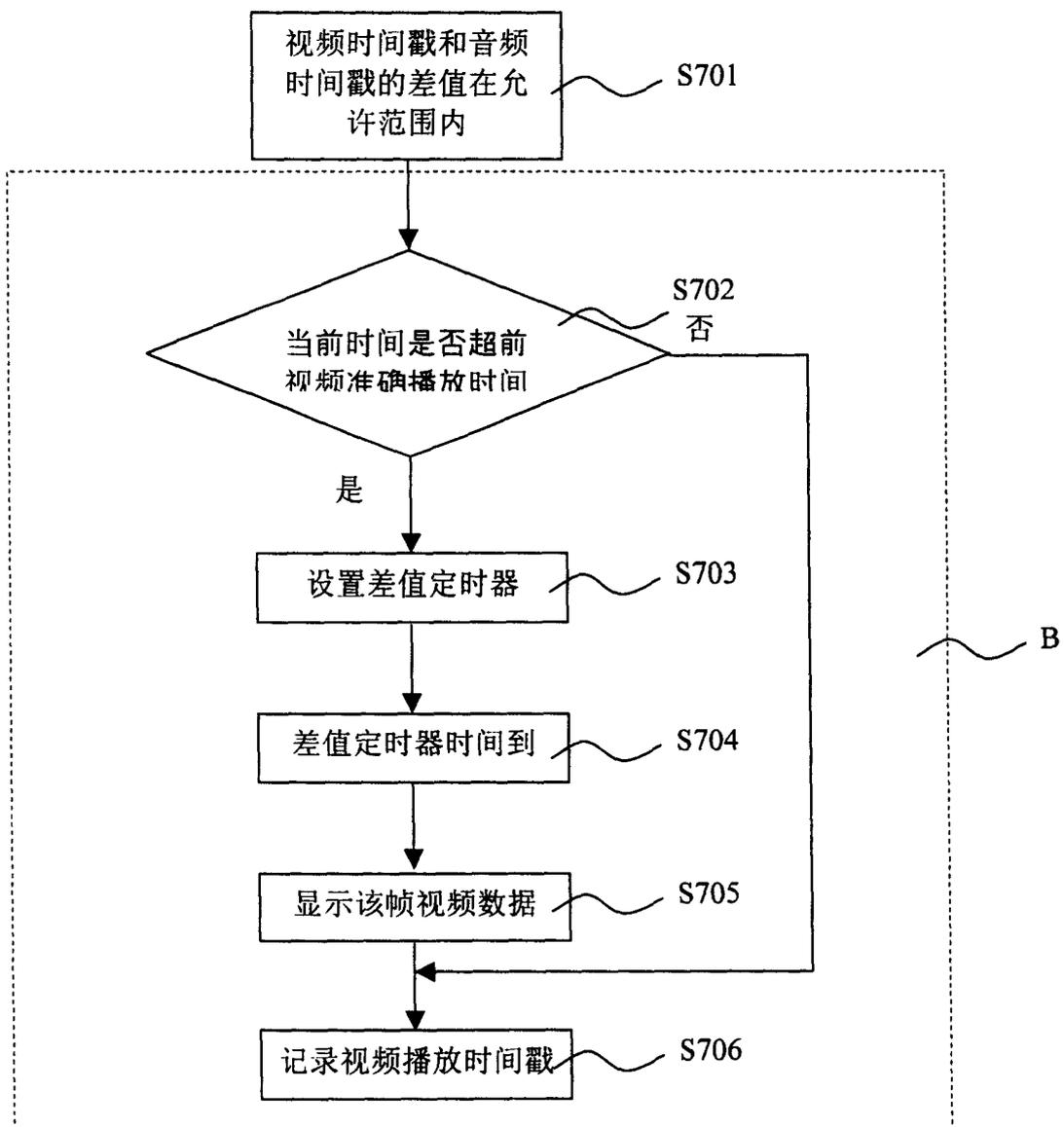


图 7