

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04N 5/00 (2006.01)

G06F 9/445 (2006.01)



[12] 发明专利说明书

专利号 ZL 200710038510.8

[45] 授权公告日 2009年7月15日

[11] 授权公告号 CN 100515029C

[22] 申请日 2007.3.27

[21] 申请号 200710038510.8

[73] 专利权人 天数宽频科技(上海)有限公司

地址 200233 上海市闵行区宜山路 2016 号 9 楼 901-907 室

[72] 发明人 吕品 韩金鹏 徐征

[56] 参考文献

CN1512318A 2004.7.14

CN1642266A 2005.7.20

CN1848837A 2006.10.18

CN1655606A 2005.8.17

CN1856008A 2006.11.1

US2004/0205498A1 2004.10.14

CN1984232A 2007.6.20

CN1360256A 2002.7.24

CN1909458A 2007.2.7

审查员 吴琼

[74] 专利代理机构 上海专利商标事务所有限公司

代理人 陈亮

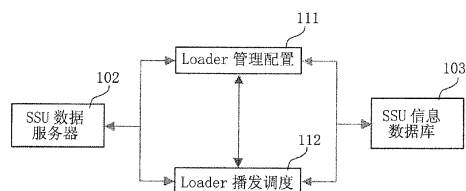
权利要求书 4 页 说明书 12 页 附图 1 页

[54] 发明名称

对机顶盒下载器文件进行统一升级管理的系统

[57] 摘要

本发明公开了一种对机顶盒 loader 文件进行统一升级管理的系统，其技术方案为：该系统包括一用于接收、存储 loader 文件并根据接收到的播发指令将 loader 文件升级广播的服务器、一用于管理调度 loader 文件的客户端和一存储 loader 记录的数据库，该客户端还包括配置管理模块，将 loader 文件传输到服务器中存储，同时根据该 loader 文件设置用于升级广播的 loader 记录并将其传输到数据库中存储；以及播发调度模块，从数据库用于升级广播的所有 loader 记录中选取所需升级广播的 loader 文件，设置 loader 文件升级广播的参数，并向所述服务器发送播发指令，同时根据参数导出当前升级的 loader 文件的描述符。本发明应用于机顶盒系统软件升级的领域。



1 一种对机顶盒下载器文件进行统一升级管理的系统，机顶盒厂商向运营商提供各自的下载器文件，由所述运营商使用所述系统对所有的机顶盒下载器文件进行统一的管理、调度和广播，其特征在于，所述系统包括一用于接收、存储下载器文件并根据接收到的播发指令将所述下载器文件升级广播的服务器、一用于管理调度下载器文件的客户端和一存储下载器记录的数据库，所述客户端包括：

配置管理模块，将所述下载器文件传输到所述服务器中存储，同时根据该下载器文件设置用于升级广播的下载器记录并将其传输到所述数据库中存储；

播发调度模块，从所述数据库用于升级广播的所有下载器记录中选取所需升级广播的下载器文件，设置所述下载器文件升级广播的参数，并向所述服务器发送播发指令，同时根据所述参数导出当前升级的下载器文件的描述符；

所述系统中定义了通用的链接下载描述符，与机顶盒约定了与升级范围相关的参数，这些参数使运营商可任意选定部分机顶盒进行升级。

2 根据权利要求1所述的统一升级管理的系统，其特征在于，所述下载器记录中包含厂商信息、硬件版本、软件版本、下载器文件格式、下载器文件在服务器中的存储位置。

3 根据权利要求1所述的统一升级管理的系统，其特征在于，所述下载器文件升级广播的参数包含升级的厂商名、升级的起止时间、升级的范围选择、下载器文件升级使用的包标识符和带宽。

4 根据权利要求3所述的统一升级管理的系统，其特征在于，所述升级的范围是机顶盒序号、机顶盒批次号或者硬件软件版本。

5 根据权利要求1所述的统一升级管理的系统，其特征在于，所述系统还包括一业务信息/节目特定信息的信息播出模块，接收经所述播发调度模块导出的当前升级的下载器文件的描述符并予以播发。

6 根据权利要求1所述的统一升级管理的系统，其特征在于，所述服务器中包含一码流封装模块，采用预设的统一系统技术应用规范将下载器文件封装打包。

7 根据权利要求1所述的统一升级管理的系统，其特征在于，所述机顶盒厂商还提供下载器文件对应的下载器描述文件，所述配置管理模块从所述下载器描述文件中自动提取用于升级广播的下载器记录信息。

8 根据权利要求1所述的统一升级管理的系统，其特征在于，所述配置管理模块包括下载器记录管理子模块和下载器文件升级历史管理子模块，所述下载器记录管理子模块查询、添加、修改、删除所述数据库中保存的下载器记录信息；所述下载器文件升级历史管理子模块查询所述播发调度模块记录在所述数据库中的升级历史信息。

9 根据权利要求1所述的统一升级管理的系统，其特征在于，所述播发调度模块通过和所述服务器通信来获得当前播发的下载器文件升级广播的参数。

10 根据权利要求1所述的统一升级管理的系统，其特征在于，所述系统还包括一信息交互模块，提供通信接口以实现所述系统与外部系统的通信。

11 根据权利要求1所述的统一升级管理的系统，其特征在于，所述与升级范围相关的参数包括以下信息：升级机顶盒号范围、升级区域、升级群体号、升级版本号、升级批次和升级厂商。

12 根据权利要求1所述的统一升级管理的系统，其特征在于，所述链接下载描述符包括以下参数：

描述符标签值，描述子表标识；

描述符长度值，表示链接下载描述符的循环中的字节总数；

传输流标识值,存储下载频点信息、QAM信息和符号率信息的NIT表的标识;
原始网络标识值,存储升级数据包下载所在频道的原始网络的唯一网络标识
值;

服务标识值,表示下载数据流的标识,为描述升级数据包的PMT所在频道的
节目号;

链接类型值,固定为0X09;

统一的设备厂商唯一编号,用于厂商识别;

循环长度值,表示OUI后的循环的字节长度值;

下载类型值,定义软件下载格式;

区域标志值,定义区域寻址控制标志;

群体标志值,定义群体寻址控制标志;

组件标签值,定义下载器文件升级使用的包标识符;

硬件主版本值,由各接收机厂商自行定义;

硬件次版本值,由各接收机厂商自行定义;

软件类型值,定义机顶盒的软件类型;

软件主版本值,由各厂商自行定义;

软件次版本值,由各厂商自行定义;

控制码,定义升级类型;

升级范围起始值,在按批次升级时,该值存储机顶盒升级批次范围的起始,
在按序列号升级时,该值存储机顶盒升级序列号范围的起始;

升级范围结束值,在按批次升级时,该值存储机顶盒升级批次范围的结束,
在按序列号升级时,该值存储机顶盒升级序列号范围的结束;

参数保留值,保留将来携带与升级有关的参数;

区域号码值,表示存储在机顶盒内的区域号码;

群体号码值,表示存储在机顶盒内的群体号码。

13 根据权利要求12所述的统一升级管理的系统,其特征在于,所述描述符
标签值固定为0x4A。

14 根据权利要求 12 所述的统一升级管理的系统，其特征在于，所述下载类型值参数进一步包括：

序列号来源，可表示为机顶盒号、智能卡卡号和配对时的配对号其中之一；

下载标识，可标识自动提示下载方式或手动下载方式；

传输流格式，可表示传输流是 DC 方式封装或私有格式。

15 根据权利要求 12 所述的统一升级管理的系统，其特征在于，所述软件类型值参数进一步包括：

完整软件类别，表示系统软件以及应用软件的完整整体；

驱动软件类别，表示最小下载器升级和硬件诊断；

测试用系统软件；以及

软件模块类别，为终端提供的附加外设设备的专用驱动程序；

16 根据权利要求 12 所述的统一升级管理的系统，其特征在于，所述控制码参数进一步包括以下信息：

强制升级，对软件版本不等于当前流中软件版本的用户升级；

基本升级，对所有低于当前播放软件版本的用户升级；

按批次升级，对在批次范围中并且软件版本低于当前播放软件版本的用户升级；

按序列号升级，对在序列号范围中并且软件版本低于当前播放软件版本的用户升级。

对机顶盒下载器文件进行统一升级管理的系统

技术领域

本发明涉及一种对软件的升级播发进行统一管理的系统，尤其涉及一种对机顶盒系统软件的升级播发进行统一管理的系统。

背景技术

随着数字电视的发展，网络内数字电视机顶盒的厂商和版本也逐渐增多，机顶盒软件版本升级也成为数字电视网络需要面临的重要问题。业务升级、开通新的增值应用以及解决机顶盒的软件缺陷都需要进行机顶盒的软件升级。

现有的对机顶盒的软件升级主要采用以下三种方式。第一种方式是每个厂商都使用自己的下载器(loader)广播系统，这种情况下运营商若需要对一个新的机顶盒生产厂商进行软件升级，则需要单独再行购置该厂商的loader广播系统，导致投资的增加。第二种方式是每个厂商仅提供码流，但需要在网络信息表NIT(Network Information Table)中插入自己定义的描述符，各个厂商之间的描述符不能通用，故而给EPG系统增加了负担。第三方式是每个厂商通过自己的CA系统进行广播，这样就很难根据运营商的要求增加根据机顶盒批次号选择性升级的功能。

发明内容

本发明的目的在于解决上述问题，提供了一种对机顶盒loader文件进行统一升级管理的系统，它确定了统一的loader技术规范，更好地解决了网络中不同厂商的机顶盒的软件版本管理问题，实现不同厂商升级码流同时进行播发的功能。

本发明的技术方案为：一种对机顶盒下载器文件进行统一升级管理的系统，机顶盒厂商向运营商提供各自的下载器文件，由所述运营商使用所述系统对所有的机顶盒下载器文件进行统一的管理、调度和广播，其中，所述系统包括一用于接收、存储下载器文件并根据接收到的播发指令将所述下载器文件升级广播的服务器、一用于管理调度下载器文件的客户端和一存储下载器记录的数据库，所述客户端包

括:

配置管理模块,将所述下载器文件传输到所述服务器中存储,同时根据该下载器文件设置用于升级广播的下载器记录并将其传输到所述数据库中存储;

播发调度模块,从所述数据库用于升级广播的所有下载器记录中选取所需升级广播的下载器文件,设置所述下载器文件升级广播的参数,并向所述服务器发送播发指令,同时根据所述参数导出当前升级的下载器文件的描述符;

所述系统中定义了通用的链接下载描述符,与机顶盒约定了与升级范围相关的参数,该些参数使运营商可任意选定部分机顶盒进行升级。

上述的统一升级管理的系统,其中,所述下载器记录中包含厂商信息、硬件版本、软件版本、下载器文件格式、下载器文件在服务器中的存储位置。

上述的统一升级管理的系统,其中,所述下载器文件升级广播的参数包含升级的厂商名、升级的起止时间、升级的范围选择、下载器文件升级使用的包标识符和带宽。

上述的统一升级管理的系统,其中,所述升级的范围是机顶盒序号、机顶盒批次号或者硬件软件版本。

上述的统一升级管理的系统,其中,所述系统还包括一业务信息/节目特定信息的信息播出模块,接收经所述播发调度模块导出的当前升级的下载器文件的描述符并予以播发。

上述的统一升级管理的系统,其中,所述服务器中包含一码流封装模块,采用预设的统一系统技术应用规范将下载器文件封装打包。

上述的统一升级管理的系统,其中,所述机顶盒厂商还提供下载器文件对应的下载器描述文件,所述配置管理模块从所述下载器描述文件中自动提取用于升级广播的下载器记录信息。

上述的统一升级管理的系统,其中,所述配置管理模块包括下载器记录管理子模块和下载器文件升级历史管理子模块,所述下载器记录管理子模块查询、添加、修改、删除所述数据库中保存的下载器记录信息;所述下载器文件升级历史管理子模块查询所述播发调度模块记录在所述数据库中的升级历史信息。

上述的统一升级管理的系统,其中,所述播发调度模块通过和所述服务器通信来获得当前播发的下载器文件升级广播的参数。

上述的统一升级管理的系统，其中，所述系统还包括一信息交互模块，提供通信接口以实现所述系统与外部系统的通信。

上述的统一升级管理的系统，其中，所述与升级范围相关的参数包括以下信息：升级机顶盒号范围、升级区域、升级群体号、升级版本号、升级批次和升级厂商。

上述的统一升级管理的系统，其中，所述链接下载描述符包括以下参数：

描述符标签值，描述子表标识；

描述符长度值，表示链接下载描述符的循环中的字节总数；

传输流标识值，存储下载频点信息、QAM 信息和符号率信息的 NIT 表的标识；

原始网络标识值，存储升级数据包下载所在频道的原始网络的全国唯一网络标识值；

服务标识值，表示下载数据流的标识，为描述升级数据包的 PMT 所在频道的节目号；

链接类型值，固定为 0X09；

OUI (Organization Unique Identifier)，即全国统一的设备厂商唯一编号，用于厂商识别；

循环长度值，表示 OUI 后的循环的字节长度值；

下载类型值，定义软件下载格式；

区域标志值，定义区域寻址控制标志；

群体标志值，定义群体寻址控制标志；

组件标签值，定义下载器文件升级使用的包标识符；

硬件主版本值，由各接收机厂商自行定义；

硬件次版本值，由各接收机厂商自行定义；

软件类型值，定义机顶盒的软件类型；

软件主版本值，由各厂商自行定义；

软件次版本值，由各厂商自行定义；

控制码，定义升级类型；

升级范围起始值，在按批次升级时，该值存储机顶盒升级批次范围的起始，在按序列号升级时，该值存储机顶盒升级序列号范围的起始；

升级范围结束值，在按批次升级时，该值存储机顶盒升级批次范围的结束，在按序列号升级时，该值存储机顶盒升级序列号范围的结束；

参数保留值，保留将来携带与升级有关的参数；

区域号码值，表示存储在机顶盒内的区域号码；

群体号码值，表示存储在机顶盒内的群体号码。

上述的统一升级管理的系统，其中，所述描述符标签值固定为 0x4A。

上述的统一升级管理的系统，其中，所述下载类型值参数进一步包括：

序列号来源，可表示为机顶盒号、智能卡卡号和配对时的配对号其中之一；

下载标识，可标识自动提示下载方式或手动下载方式；

传输流格式，可表示传输流是 DC 方式封装或私有格式。

上述的统一升级管理的系统，其中，所述软件类型值参数进一步包括：

完整软件类别，表示系统软件以及应用软件的完整整体；

驱动程序类别，表示最小下载器升级和硬件诊断；

测试用系统软件；以及

软件模块类别，为终端提供的附加外设设备的专用驱动程序；

上述的统一升级管理的系统，其中，所述控制码参数进一步包括以下信息：

强制升级，对软件版本不等于当前流中软件版本的用户升级；

基本升级，对所有低于当前播放软件版本的用户升级；

按批次升级，对在批次范围中并且软件版本低于当前播放软件版本的用户升级；

按序列号升级，对在序列号范围中并且软件版本低于当前播放软件版本的用户升级。

本发明对比现有技术有如下的有益效果：本发明中的客户端配置管理模块用于管理系统升级软件需要设置的各种网络信息以及系统版本信息，服务器作为数字视频广播 DVB (Digital Video Broadcasting) 前端发送系统软件升级码流的服务器，通过标准通信方法接收来自客户端播发调度模块的命令，根据控制命令将存储在待播库中的系统升级软件（即 loader 文件）打包发送，并具有一定的容错性、报警提示和日志功能。本发明方便了网络中不同机顶盒软件升级版本的管理问题，实现不同厂商升级码流同时进行播发，而且在一定程度上减轻了机顶盒大面积升级

带来的负面影响。

附图说明

图 1 是本发明一个实施例的网络结构图。

图 2 是本发明一个实施例的系统框架图。

具体实施方式

下面结合附图和实施例对本发明作进一步的描述。

图 1 示出了本发明一个实施例的网络结构。如图 1 所示，在前端的网络中，由客户端 101、SSU(系统软件升级, System Software Upgrade)数据服务器 102、SSU 信息数据库 103 通过网络交换机 104 连接成统一 loader 系统 100。复用器 200 同 SSU 数据服务器 102 连接, EPG(Electronic Program Guide)系统 301 和 OSS/SMS (Operational Support System/Subscriber Management System) 系统 302 通过网络交换机 104 与统一 loader 系统 100 连接。

图 2 示出了本发明一个实施例的系统框架。如图 2 所示, 统一 loader 系统 100 可划分为配置管理模块 111、播发调度模块 112、SSU 数据服务器 102 以及 SSU 信息数据库 103。其中 SSU 数据服务器 102 部署在使用 Linux 操作系统的服务器上, 而其余模块均部署在使用 Windows 操作系统的客户端计算机上。

下面结合一个实例来说明统一 loader 系统 100 中各个模块的实现过程。在该实例中, 某机顶盒生产厂商提供了一个 loader 文件以及对应的 loader 描述文件。该 loader 描述文件内容如下: 生产厂商名称: ABC, 硬件版本: 0082, 软件版本: 0X44, 下载格式: 私有格式, 文件格式: TS 文件, 以及该 loader 文件存放在 SSU 数据服务器 102 上的 D 盘根目录下等信息。如果要升级, 通常情况下需要生成硬件版本相同而软件版本更高的一个 loader 文件用于在线升级, 因而在这里假定其待升级的软件版本为 0X43。

厂商向运营商提供用于升级的 loader 文件和对应的 loader 描述文件后, 配置管理模块 111 根据 loader 描述文件将相应的 loader 文件传输到 SSU 数据服务器 102 的指定路径中。同时, 配置管理模块 111 在 SSU 信息数据库 103 中新增一条对应的 loader 记录, 该记录的格式依次为: 厂商信息、硬件版本、软件版本、loader

文件格式、loader 文件在 SSU 服务器 102 上的存储位置。在这里，新增加的记录是“ABC、0082、0X44、TS 文件、D:\”。每一个不同的 loader 文件都对应不同的 loader 记录，而所有的 loader 记录组成一个 loader 记录数据库。

当然，配置管理模块 111 也可以不通过提取 loader 描述文件的信息而是根据厂商提供的相关信息直接手动添加 loader 记录。配置管理模块 111 还可检测 loader 描述文件是否有错误，能够判断 loader 文件是否可以用于升级广播，如果 loader 文件已经是 TS 文件，也可判断其格式是否满足统一规范的要求。配置管理模块 111 可再细分为 loader 记录管理子模块和 loader 文件升级历史管理子模块。loader 记录管理子模块可查询、添加、修改和删除 loader 记录数据库中的 loader 记录信息。Loader 文件升级历史管理子模块可查询播发调度模块 112 记录在数据库中的升级历史信息，包括系统升级软件的厂商、硬件版本、软件版本、升级开始时间和升级结束时间等信息。

然后，播发调度模块 112 连接 SSU 数据服务器 102。在上述的配置管理模块 111 中，可以将一个厂商设置成一项服务 (Service)，将对应该厂商的 loader 文件配置成该服务下的基本流 (Elementary Stream)。运营商选择当前需要进行升级的厂商 (即选择服务)，选择当前用于升级的 loader 记录 (即对应服务下的基本流 ES)，并设置 loader 文件播发信息的参数，比如设置升级的起止时间，设置升级的范围选择、loader 文件升级使用的 PID 和带宽等。设置完成后即可将开始该 loader 文件升级播发的指令发送给 SSU 数据服务器 102。播发调度模块 112 还可通过与 SSU 数据服务器 102 的通信来获得当前播发的 loader 文件升级广播的参数，还可将所有的播发调度操作记录到 SSU 信息数据库 103 中，供配置管理模块 111 使用。

其中，升级的范围选择可以是根据机顶盒序号、机顶盒批次号或者仅根据机顶盒的硬件软件版本来选择。按机顶盒序号或批次号或硬件软件版本进行升级的意义在于，如果针对全部网络进行升级，有线网络由于信号质量原因，在很大的覆盖面上出现故障的可能就很多，会导致运营商难以应付的局面。而如果按序号、批次号或硬件软件版本进行升级，则可以将升级所影响的范围限定在一个相对较小的范围，出现的故障数量相对要少很多。这样就可以通过连续的小批量升级将整个网络内的机顶盒都完成一次升级。

播发调度模块 112 可根据当前的 loader 文件升级广播导出 SI/PSI 需要的描述升级信息的描述符内容，即当前有哪些厂商的哪些 loader 文件在进行升级，他们各自的标签 Tag（也即 PID）、升级范围的选择等信息。SI/PSI 信息使用 EPG 系统 301 来播出。EPG 系统 301 接收上述的描述符，并将其插入到指定的 SI 表中，完成对 NIT 的设置。只要保证 EPG 系统 301 配置并播出的 SI/PSI 信息与播发调度模块 112 使用的信息协调一致，顺利地进行 loader 文件升级就得到了保障。

SSU 数据服务器 102 接收播发调度模块 112 的播发指令，将 loader 文件及与之相关的控制信息通过 ASI 输出口传输给复用器 200 等设备实现广播。

SSU 数据服务器 102 可接收遵循统一系统技术应用规范打包的传送流（TS，即 Transport Stream）文件、未打包的 loader 文件以及各厂商专用格式的 TS 文件。SSU 数据服务器 102 中的码流封装模块根据接收的 loader 文件的类型，采用不同的处理方式，确保发出正确的升级数据码流。处理方式如下：在遇到厂商专用格式的 TS 文件时，先检验其是否是合法的 TS 文件，并根据指定的 PID 进行重新映射；在遇到厂商专用格式的未打包的 loader 文件时，根据指定的 PID 从 Section 数据打包成 TS 文件；而对于 DC 格式的未打包的 loader 文件，首先采用统一系统技术应用规范中使用的数字存储媒体命令与控制（DSM-CC，即 Digital Storage Media-Command&Control）两层数据轮播（Data Carousel）协议进行数据的封装，打包生成 TS 文件，然后再行广播。

loader 文件广播之后，机顶盒随时检测到 NIT 中发生的变化，如果 NIT 中携带上述的描述符，机顶盒将逐一判断当前广播的 loader 文件中是否有符合它的 loader 文件。判断顺序依次为厂商标识、硬件版本、软件版本，如果设置了升级范围则还需判断升级范围是否符合自身。如果上述条件都通过了，则机顶盒根据描述符中指定的 Tag（PID），开始接收具有相同 PID 的 loader 文件，接收并校验完毕之后自动重新启动机顶盒完成系统软件的升级。

整个统一 loader 系统还可以包括一个信息交互模块，该信息交互模块用于提供通信接口，通过该接口实现和外部系统的通信。比如提供升级前后分析接口给 OSS，用于提供分析数据。

此外，统一 loader 系统还定义了通用的链接下载描述符（linkage downloader descriptor），与机顶盒约定了包括升级机顶盒号范围、升级区域、升级群体号、升

级版本号、升级批次和升级厂商等与升级范围相关的参数。这些参数的组合保证运营商可以任意选定部分机顶盒进行升级，从而避免了由于厂商、版本号、批次的不同引起的机顶盒升级带来的负面影响。运营商可先进行小范围的升级，之后再进行大批量的升级，保证用户在升级过程中的问题不至于一下子爆发。

链接下载描述符的具体结构如下表所示：

Syntax	No. of bits	Identifier
Linkage_Descriptor() {		
Descriptor_tag =0x4A	8	Uimsbf
Descriptor_length	8	Uimsbf
Transport_stream_id	16	Uimsbf
Original_network_id	16	Uimsbf
Service_id	16	Bslbf
Linkage_type = 0x09	8	Uimsbf
OUI_data_length	8	Uimsbf
for (I=0;I<N;I++) {		
OUI	24	Bslbf
selector_length	8	Uimsbf
for (j=0; j<N; j++){		
Update_type	6	Bslbf
Region_flag	1	Uimsbf
Colony_flag	1	Uimsbf
Component_tag	8	Uimsbf
Hardware_major_version	16	Uimsbf
Hardware_minor_version	16	Uimsbf
Software_type	8	Uimsbf
Software_major_version	16	Uimsbf
Software_minor_version	16	Uimsbf
Serial_number_start	32	Uimsbf

Serial_number_end	32	Uimbsf
Control_code	8	Uimbsf
Private_data	32	Bslbf
if(region_flag =0x1)		
region_no	16	Bslbf
if(colony_flag ==0x1)		
conlony_no	16	Bslbf
}		
}		
For(I = 0 ;I < N; I ++)		
{		
private_data_byte	8	Bslbf
}		
}		

下面是对链接下载描述符中各字段的描述:

Descriptor_tag: 描述子表 ID, 其值定为 0x4A。

Descriptor_length: 此后 Descriptor 循环中字节总数。

Transport_stream_id: 存放下载频点信息, QAM 信息, 和符号率信息的 NIT 表的 ID。

Original_network_id: 存储升级数据包下载所在频道的原始网络的全国唯一网络标识值。

Service_id: 下载数据流的 ID, 该值应为描述升级数据包的 PMT 在该频道的节目号, 一般定义一个 Service-id 对应一个厂商。

Linkage_type: 其值固定为 0X09

OUI: 是一个标准定义, 在本实施例中高字节暂未定, 低两字节用于厂商识别。

Selector_length: 为以下循环的字节总长度。

Update_type: 大小为 6 位 (bit), 定义软件下载格式如下:

Bit5	Bit4	Bit3~2	Bit1	Bit0
		Serial_Number_Source(2 bits)	Download_flag	TS_format

	定义	值	说明
Bit5	未定义	1	
Bit4	未定义	1	
Bit3	Serial_Number_	见下表	序列号来源
Bit2	Source		
Bit1	Download_flag	1	自动提示下载,收到新的软件版本之后,自动弹出新版本软件下载提示窗口。
		0	手动下载,收到新的软件版本之后,只是记录下来,不自动弹出新版本软件下载提示窗口。
Bit0	TS_format	1	厂商提供的根据 DC 方式封装的 TS 流文件
		0	为厂商提供私有格式 TS 流

Serial_Number_Source	说明
00	序列号为机顶盒号
01	序列号为智能卡卡号
10	序列号为配对时的配对号
11	保留

Region_flag: 1bit, 定义区域寻址控制标志, 0x0 不按区域寻址, 0x1 按区域寻址。

Colony_flag: 1bit, 群体寻址控制标志, 0x0 不按群体寻址, 0x1 按群体寻址。

Component_tag: 8bit, 定义本版本的 Loader PID, Service_id 对应的 PMT 中所对应的 elementary stream 必须带 component descriptor (descriptor_tag = 0x50)。

Hardware_major_version: 16 bit 硬件主版本,由各接收机厂商自行定义。

Hardware_minor_version: 16 bit 硬件次版本,由各接收机厂商自行定义。

Software_TYPE: 8 bit 软件类型,参见下表。

软件类别	CODE 值	备 注
完整软件	0X00	系统软件以及应用软件完整整体
驱动软件	0X10	Miniload 升级、硬件诊断等
测试用系统软件	0X20	
软件模块	0x30-0x7F	为终端提供的附加外设设备专用驱动程序,每个驱动软件码(CODE)值需事先单独分配,例如 USB
保留	0X80~0XFF	

Software_major_version: 16 bit 软件主版本。由厂商自行定义。

Software_minor_version: 16 bit 软件主版本。由厂商自行定义。

Control_code: 8bit 升级类型定义,见下表。

升级类型	CODE 值	备 注
强制升级	0X00	对软件版本不等于当前流中软件版本的用户升级
基本升级	0X01	对所有低于当前播发软件版本的用户升级
按批次升级	0X02	对在批次范围中并且软件版本低于当前播发软件版本的用户升级
按序列号升级	0X03	对在序列号范围中并且软件版本低于当前播发软件版本的用户升级
保留	0X4-FF	保留

Serial_number_start: 大小为 32 bit, 需要时可定义为:

需要按批次升级时,是升级该程序的机顶盒批次范围起始;

需要按序列号升级时，是升级该程序的机顶盒序列号范围起始。

Serial_number_end: 大小为 32 bit，需要时可定义为：

需要按批次升级时，是升级该程序的机顶盒批次范围结束；

需要按序列号升级时，是升级该程序的机顶盒序列号范围结束。

Private_data: 保留用于将来携带与升级有关的参数：

region_no: 16bit，存储在机顶盒内的区域号码；

colony_no: 16bit，存储在机顶盒内的群体号码。

上述实施例是提供给本领域普通技术人员来实现或使用本发明的，本领域普通技术人员可在不脱离本发明的发明思想的情况下，对上述实施例做出种种修改或变化，因而本发明的保护范围并不被上述实施例所限，而应该是符合权利要求书提到的创新性特征的最大范围。

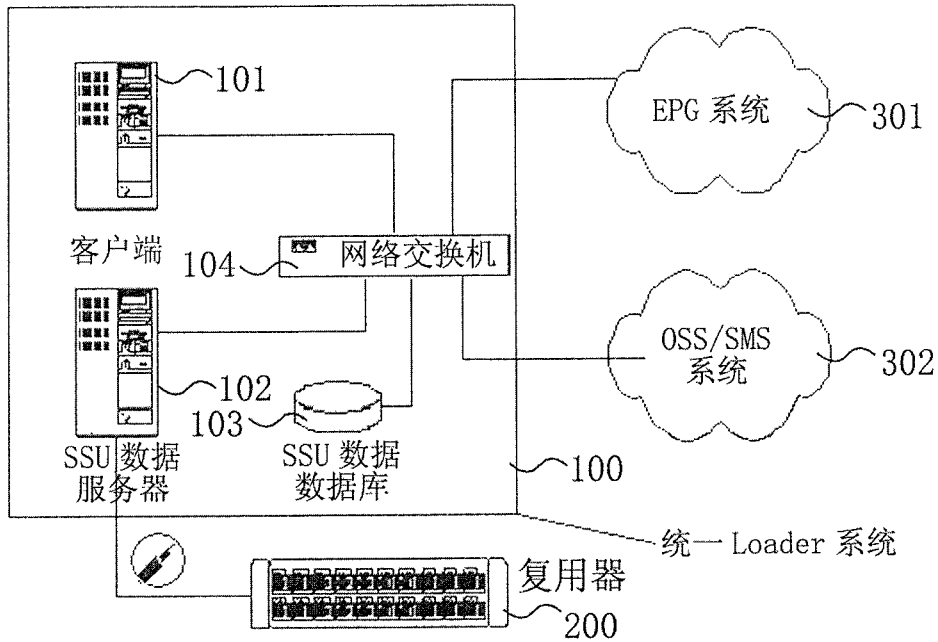


图 1

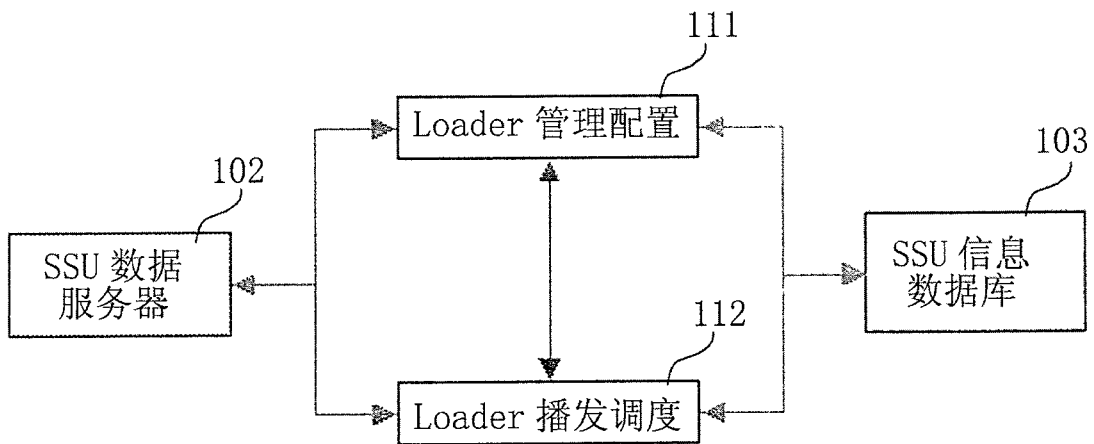


图 2