

基于 DM6446 的 P2P 机顶盒 OSD 研究与实现

黄益贵, 王汝传

(南京邮电大学 计算机学院, 江苏 南京 210003)

摘要: OSD 是 P2P 机顶盒的重要组成部分, 它是用户与程序之间交互的接口。DM6446 平台虽然对显示硬件的使用提供有部分 API 接口, 但不足以开发出良好的 UI, 因此文中基于该平台, 通过对硬件抽象层进行研究, 对 OSD 的框架, 特别是图层混合显示的工作原理及应用进行了深入研究, 并通过图像分层与混合、alpha 通道等技术给出了 OSD 在该平台上的具体实现, 同时对该平台 OSD 实现所缺乏的中文字符支持给予了详尽的讨论, 并通过使用点阵字库技术给出了解决方案。最终效果表明所提出的方案具有可行性, 并且获得了良好的效果。

关键词: P2P; 机顶盒; OSD; 中文显示

中图分类号: TN941.3

文献标识码: A

文章编号: 1673-629X(2010)04-0220-04

Research and Implementation of OSD for P2P Set - Top - Box Based on DM6446

HUANG Yi-gui, WANG Ru-chuan

(College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

Abstract: On Screen Display(OSD) is an important component of peer-to-peer set-top-box, which links the consumers and procedures. The DM6446 platform supports many API to developer about how to use display hardware, but not enough to develop excellent UI, hence, deeply research in hardware abstract layer of the platform, especially principle and application of image layer mix display in framework of OSD, giving a realization by image delamination and mix, alpha channels technology etc. Also discuss thoroughly lack of display of Chinese characters in this platform, giving a solution by lattice character display technology. Ultimate effect indicates our proposal is feasible, and have good user experience.

Key words: P2P; set-top-box; OSD; Chinese characters display

0 引言

数字电视的迅速发展, 让越来越多的模拟电视面临淘汰的局面。但由于庞大的模拟电视用户群, 不可能在一夜之间全部消失, 因此作为一种过渡产品, 数字电视机顶盒应运而生, 它让人们在享受到高清晰的视频信号之外, 又可以进行电子商务、远程教育、在线游戏等交互式活动。

根据传输介质的不同, 数字电视机顶盒可分为卫星数字电视机顶盒(DVB-S)、地面数字电视机顶盒(DVB-T)、有线数字电视机顶盒(DVB-C)。由于目前国内有线电视网络普遍采用光纤同轴混合网(HFC), 全双工通信改造起来简单, 成本低廉, 因此有线数字电视机顶盒具有广阔的应用前景。

OSD 是机顶盒中的重要组成部分, 它关系到用户与软件系统的交互, 只有设计良好的 OSD, 才能获得良好的用户体验, 但在 TI 公司 Davinci 技术 TMS320 DM6446 开发平台上开发出良好的 OSD 还具有一些实现上的挑战, 文中通过深入研究 OSD 的原理与实现机制给予在该平台上的解决方案。

1 P2P 机顶盒 OSD 研究与实现

P2P 机顶盒是利用了先进的 peer-to-peer(P2P)技术, 在视频点播的过程中, 同时实现下载和上传视频文件流以达到分担视频文件服务器负载和提高下载速

收稿日期: 2009-07-23; 修回日期: 2009-10-12

基金项目: 国家自然科学基金(60973139, 60773041); 江苏省自然科学基金(BK2008451); 国家高科技 863 项目(2007AA01Z404, 2007AA01Z478); 省级现代服务业发展专项资金; 南京市高科技项目(2007 软资 127); 江苏高校科技创新计划项目(CX08B-08SZ, CX08B-086Z); 江苏省六大高峰人才项目(2008118)。

作者简介: 黄益贵(1984-), 男, 安徽寿县人, 硕士研究生, 研究方向为计算机网络、对等计算和信息安全等; 王汝传, 教授, 博士生导师, 研究方向是计算机软件、计算机网络和网络、对等计算、信息安全、无线传感器网络、移动代理和虚拟现实技术等。

度的目的,其原理图如图 1 所示,在图中每个数字电视机顶盒相当于一个 Peer 结点,它根据客户端电子节目单(EPGClient)列表项通过网络向 EPGServer 查询视频资源的地址,EPGServer 还回一个 URL,然后机顶盒

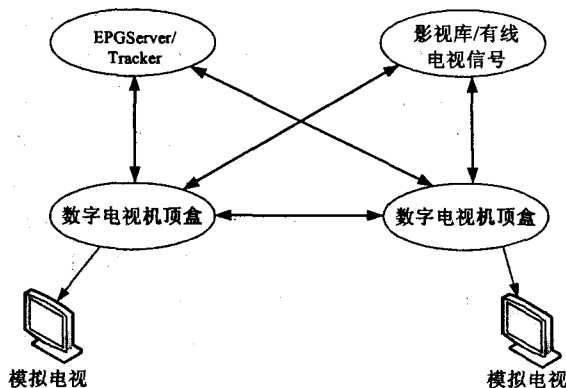


图 1 P2P 机顶盒网络原理

根据统一资源定位 URL 所给定的地址找到资源所在地服务器,下载资源并向 Tracker 登记自己的下载进度以及地址等相关信息,当有若干个机顶盒同时在请求同一个视频资源的时候,因为 P2P 资源是分片存放在服务器上的,所以每一个机顶盒都可以在下载的同时给其它的机顶盒提供自己已经下载好的资源片断,并且也可以同时从其它机顶盒上请求别人已经下载好的资源片断,这样就可以分担服务器的负载同时提高资源下载的速度^[1,2]。

1.1 OSD 开发模型

消息驱动模型是创建用户界面软件的高效方法,它有很多优点,如:可重用性、高移植性、实时性等。在本机顶盒 OSD 编程中就是采用应用层的消息驱动模型,通过共享内存实现与其他进程的通信,如图 2 所示,其中 IRD 为红外线遥控器守护进程,OSDComponent 为响应 OSD 菜单的控件,EPGClient 为 EPGServer 的客户端,具体实现如下:

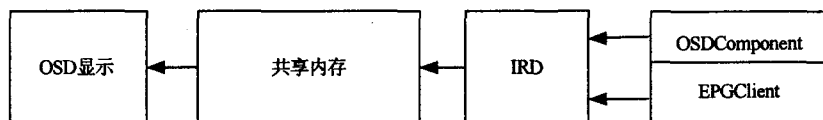


图 2 OSD 开发模型

客户通过遥控器遥控机顶盒,触发红外线遥控器守护进程 IRD,IRD 捕获触发事件并将要显示的 OSD 数据(如从 EPGClient 中读入的节目列表、从 OSDComponent 中读入的按键控件变更信息)存入共享内存中,然后 OSD 显示模块函数根据相应的驱动事件的消

息从共享内存中读取数据并显示在 LCD 上。

1.2 OSD 体系结构

DaVinci DM6446 开发平台处理器支持六层 OSD 体系结构,从下向上依次是背景窗口层(background)、两个视频窗口层(Video Window 0、Video Window 1)、两个 OSD 窗口层(OSD Window 0、OSD Window 1)以及一个鼠标窗口层(Cursor)^[3],其中视频窗口层采用 YCbCr 色彩编码方案,而 OSD 窗口层采用 RGB 色彩编码方案。如图 3 所示。

①YCbCr:主要用于视频图像中,其中 Y 是指亮度分量,也就是灰阶质;Cb 指蓝色色度分量;而 Cr 指红色色度分量。人的肉眼对视频的 Y 分量更敏感,因此在通过对色度分量进行子采样来减少色度分量后,肉眼将察觉不到的图像质量的变化。主要的子采样格式有 YCbCr 4:2:0、YCbCr 4:2:2 和 YCbCr 4:4:4。

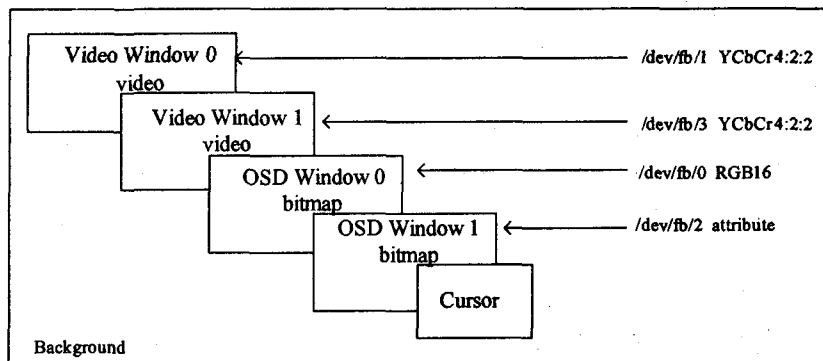


图 3 OSD 体系结构

②RGB:红绿蓝(RGB)是计算机显示的基色,RGB 有多种格式,如 RGB24(8:8:8)、RGB16(5:6:5)、RGB15(5:5:5),其中 RGB24、RGB16、RGB15 支持的色深分别为每像素 24 位、16 位、15 位,总共可以表达的色种类别分别为 $2 \cdot 24 = 16\text{M}$ 种、 $2 \cdot 16 = 64\text{k}$ 种、 $2 \cdot 15 = 32\text{k}$ 种。

因为这里重点研究 OSD 部分,所以下面只研究 RGB 图像的操作。

(1) 图像分层与混合。

对于 RGB 图像我们知道每一个像素点都由红、

绿、蓝三种通道,而红、绿、蓝三个通道里的像素都是(灰阶值为 0 - 255)灰色的,在两个图层上的图像不是直接“混合”的,而是通过各自的

红、绿、蓝通道进行“混合”的。由于混合模式算法有很多种,如线性减淡 $R = a + b$,滤色 $R = a + b - a * b / 255$,正片叠底: $R = a * b / 255$ 等。这里 a 表示“基层”,“基层”就是被混合的层(图层位置在下面),b 表示“混合层”,“混合层”就是去混合“基层”的层(图层位置在上面),混合模式算法中对 a、b 的位置是有要求的,有

些可以上下颠倒位置,有些则不能颠倒。这里仅讨论用于图像混合的正片叠底图层混合模式算法^[4]:

①a 表示“基层”,b 表示“混合层”,R 表示混合后的图像;

②a、b、R 又分别表示“基层”、“混合层”、混合后图像中通道的灰阶值,并且分别表示 R、G、B。

③a、b、R 分别取[0,255]之间的正整数;

则运算公式如下:

$$R = a * b / 255$$

举例如下:

一幅“基层”图像某点的{R、G、B}为{86、73、253},而一幅“混合层”图像某点的{R、G、B}为{82、42、214},则混合后图像的{R、G、B}计算结果如下:

对于 R 通道中的计算:

$$R = 86 * 82 / 255 = 27.655$$

对于 G 通道中的计算:

$$R = 73 * 42 / 255 = 12.024$$

对于 B 通道中的计算:

$$R = 253 * 214 / 255 = 212.322$$

即运算后图像的{R、G、B}为{27、12、212},因此可以得到混合后要输出的结果。

(2)Alpha 通道。

在 OSD 图像混合中,为了实现图像的透明效果,采取在图像文件的处理与存储中附加上另一个 8 位信息的方法,这个附加的通道代表图像中各个像素点透明度的信息就叫做 Alpha 通道^[5]。Alpha 通道使用 8 位二进制数,最多可以表示 $2^8 = 256$ 级灰度,即 256 级的透明度。这样一来,一幅完整的全信息图像应该是 24 位 RGB 彩色加 8 位 256 级的透明信息,一共是 32 位。Alpha 通道通常白色值为 255,用以定义完全不透明的像素,而黑色值为 0,用以定义完全透明像素,介于黑白之间的灰度(值为 0-255)的 Alpha 像素用以定义从完全透明到完全不透明之间的过渡像素。计算机图形学中总是把[0-255]映射到[0-1]区间上,即 Alpha 值为 0-1 之间的数值,这样一个完成的图像分量信号就由 RGBA 组成(其中 A 就是 Alpha),并且每幅图像的输出等于{R * A、G * A、B * A},图像混合后的输出就是

$$\text{Output} = \text{Alpha} * \text{foreground} + (1 - \text{Alpha}) * \text{background}$$

其中 Output 为输出混合后的图像,foreground 为“混合层”图像,background 为“基层”图像。调整这个 Alpha 值便可以得到不同的混合效果。当 Alpha 为 0 时,只显示“基层”图像;当 Alpha 值为 1 时,只显示“混合层”图像,当取其它介于 0-1 之间的值可取得按照

一定比率混合的图像,这样就可以按照一定的方式实现文字叠加、图片混合等效果,可以调整 Alpha 值直到符合要求。

(3)DM6446 中的具体实现。

在 DaVinci DM6446 平台中,TI 公司把视频输出硬件抽象成一个帧缓冲设备,这样在 linux 操作系统下,可以直接对其执行像操作一般文件一样打开、读、写和关闭等操作,因此可以直接在其上进行快速 OSD 开发。在 OSD 体系结构图 2 中,OSDWIN1 是用于配制混合图像属性的窗口(例如透明度),OSDWIN0 是用于显示图像的窗口。由于 DM6446 支持 RGB565 或位图数据两种格式,又由于 RGB 图像格式表达的色彩种类可达 64k 种,因此这里选用 RGB565 格式,但是 RGB 数据是原始未压缩数据,不包含常用于图像编辑器识别图像的头信息,且种类众多,没有可以利用的图像编辑器直接生成 RGB565 图像文件。BMP 位图数据也是未压缩的,但它在每个位图数据后面附有文件格式头信息,且当每像素使用相同的位数表示时,位图数据格式和 RGB565 数据格式本质上是一样的。这样在实际应用中,就需要对 BMP 位图文件进行转换,使之生成 RGB16 图像文件。由于 BMP 图像是 24 位的,且其位图信息在文件中是按照颜色顺序为 BGR (24 位,每种颜色分量为 8 位)的顺序存放的,读出来后分别取 RGB 分量的高 5、6、5 位的颜色信息,并丢弃低 3 (R)、2 (G)和 3 (B)位的颜色信息,将之转换为 16 位的,且颜色顺序为 RGB (5 6 5)的 RGB565 数据,然后再存放到文件中去。使用 fread()库函数从位图文件流中分别读取蓝、绿、红的图像色彩分量,然后作为 OSD 数据写入 RGB16 数据中,这样就可以得到所需要的 RGB565 格式图像文件。将该图像文件拷贝到 OSD 数据缓冲区中,即可查看图像输出效果,调整 OSDWIN1 中的图像混合属性配制,修改其中的 Alpha 通道值,使其图像的混合输出达到满意效果。

1.3 OSD 菜单设计

OSD 菜单^[6]作为直接关系到用户体验的满意度,要求必须具有良好的人性化设计,还要能够实现强大的操作功能,因此我们对此进行了合理的规划与设计。

OSD 菜单主要包括四个组成部分:

1)本地节目(LocalWindow):该菜单读取本地硬盘上存取节目的列表,并对之进行播放、删除、上传、下载等操作等。

2)P2P 网络(NetworkWindow):该菜单与 EPG-Client 交互,读取 EPGServer 服务器上的当前频道列表以及节目信息并对之进行相应操作等。

3)系统设置(Setup):该菜单主要实现服务器 IP

和用户名、密码还有上传速度以及下载速度,同时允许连接数、端口等的设置。

4)工具(Tools):该菜单主要实现第三方软件的安装、使用以及升级,如浏览器、游戏等。

在本 OSD 菜单中,采用双向动态链表进行导航操作,实现父子窗口、父子控件以及同屏翻页、同屏滚动等事件的操作,数据结构如下:

```
typedef struct osd_ui_t
{
    struct osd_ui_t * prev;
    struct osd_ui_t * next;
    osd_item_t * head;
    osd_item_t * focus;
    OSD_UI_FUNCTION next_focus;
    OSD_UI_FUNCTION prev_focus;
}osd_ui_t;
```

当窗口链的状态发生变化时,必须动态分配和撤销这些存储区。窗口的创建、撤销、关闭等 OSD 内部函数及消息提供了很好的内存管理方法,它是整个 OSD 系统能否稳定运行的基础。

2 OSD 中文显示解决方案

由于 DaVinci DM6446 平台不支持中文显示,因此要在应用程序界面上使用中文字符,需要自己开发中文显示函数。在图形操作系统中,中文字符(包括英文)通常有两种存储格式:一种是存储字符的矢量图形^[7,8],例如,用样条的方式,拟合一个字符的所有笔画轮廓,存储样条的关键点,来实现字符的存储,这种字库称为矢量字库;另一种是存储字符的图形点阵,即是以位图形式存储字符,在数据中,用 0 或 1 区分汉字的笔画,这种字库称为位图字库或点阵字库。这两种方式各有优缺点,矢量字库不存在放大以后失真的问题,并且因为存储的是笔画样条,可以对字符做旋转、缩放、高斯模糊,甚至三维拉伸都不会失真,但是矢量字库存储的是样条,在显示的时候,增加了计算量,不利于快速显示。而对于点阵字体,虽然在放大的时候边缘会产生锯齿,影响整个字符的美观,但是由于每个相同大小的字符存储所用空间是固定的,且便于管理、易于显示,适应于性能和资源相对比较欠缺的嵌入式环境,因此在本嵌入式平台中,使用的是 16 点阵的位图字体库。

众所周知,在计算机内一个英文字符用一个字节的 ACSII 码表示,该字节的最高位经常用于奇偶校验位,因此实际表示字符的只有 7 位,即可以表示 $2^7=128$ 个字符,但是由于汉字的复杂性,需要使用两个字节表示一个汉字,针对这种情况国家制定了国标码。

国标码规定,组成两字节代码的各字节的最高位均为 0,即每个字节仅只使用 7 位,这样在机器内使用时,由于英文的 ASCII 码也在使用,可能将国标码误看作两个 ASCII 码,因而在机器内使用汉字时,每个字节的最高位均为 1,以表示该码表示的是汉字,这些国标码两字节最高位加 1 后的代码称为机器内的代码,俗称“内码”。

汉字内码与区位码有固定的转换关系,例如汉字字符“国”字的内码 0xb9fa,它所对应的区码和位码分别为:

$$qm = 0xb9 - 0xa0;$$
$$wm = 0xfa - 0xa0;$$

因而该汉字在 16 * 16 点阵汉字库中距离字库文件头的偏移位置 offset(以字节为单位),可计算得:

$$offset = (94 * (qm - 1) + (wm - 1)) * 32L$$

获取偏移位置后,调用 fseek()函数和 fread()函数从事先已经 fopen()的字库中提取要显示汉字的字模,将获取的汉字字模存储到一个字符数组中。这样,顺序读取 32 字节的点阵信息后,利用画图函数 drawpixel(int x,int y,int r,int g,int b)在屏幕上指定的(x,y)位置上对一个 16 * 16 像素的图形区域进行色彩填充(设置 r、g、b 值),如果点阵汉字中某位为 1,则填充该像素点;若为 0,则不填充该像素点,从而就可以显示出来一个汉字。

查找字库显示点阵有两种方式:一种是边查找边显示,即在显示每个汉字之前,先根据区位号读取点阵字模,对于 16 * 16 的点阵,需要 $16 * 16 / 8 = 32$ 个字节的存储空间,这样在显示所有汉字点阵之前需要大量地频繁查找字库,消耗了大量的 CPU 时间。另一种是先从字库中读取所有要显示汉字的点阵字模,放到一个缓冲区中,然后再显示,这样对于 100 个汉字所占用的缓冲区空间大约为 3kB($100 * 32 = 3200$)。由于机顶盒 CPU 需要处理大量的视频和音频数据,并且对于嵌入式系统,点阵缓冲所占用的空间并不算大,综合考虑,采用第二种方案更加合理。

3 结束语

文中所描述的基于 TI 公司 DaVinci 技术 TMS 320DM6446 的 P2P 机顶盒 OSD 研究与实现,较好地解决了该平台适应于 P2P 流媒体播放技术时界面操作和面向本土化开发所欠缺的问题,从理论和工程实践上研究了 OSD 的原理以及具体实现,汉字显示的原理以及具体实现。实践证明,该方案获得了良好的效果。

(下转第 227 页)

点密度专题图页面、等级符号专题图页面的设计与实现。当单击界面左侧的“范围专题图”链接时,页面就会利用 CallBack 机制回调 function 函数 setTheme(),在控件中实现了 ICallbackEventHandler 接口,该接口有两个方法,分别是 RaiseCallbackEvent 和 GetCallbackResult。RaiseCallbackEvent 方法是回调执行的方法,该方法处理回调的内容,它没有返回值,而是从浏览器接收一个字符串作为实践的参数,即该方法接受客户端 JavaScript 所传递的参数。它是首先出发的。接下来出发的是 GetCallbackResult 方法,它将所得到的结果传回给客户端的脚本^[3,5]。

当单击界面左侧的“点密度专题图”和“等级符号专题图”链接时,其实现原理与上面介绍的“范围专题图”的原理是一样的,不一样的地方是地图的表现不一样,“点密度专题图”是利用了 ArcGISObjects 中的 I-DotDensityRenderer 接口,此方法也放在了公共操作类 GisFunction 中^[3,5],而“等级符号专题图”利用了 ArcGISObjects 中的 IProportionalSymbolRenderer 接口,此方法也放在了公共操作类 GisFunction 中^[3,5]。

以“范围专题图”为例,当查询“水稻稻飞虱”的发病区域,在病虫害列表下拉列表中选择相应的病虫害名称,即可用不同的颜色等级显示出病虫害分布情况,反应不同区域的病虫害的受害程度信息。

2.2.3 病虫趋势图模块设计

在病虫趋势图模块主要是数据曲线图页面的设计与实现。通过查看趋势图可找到病虫害发病的一些规律,有利于防治,减少病虫害对小麦和水稻产量的危害。

实现病虫趋势图所用到的是 DotNetCharting 控件, DotNetCharting 控件是一个 .NET 图标控件,对中文支持良好,而且操作方便。当选择水稻稻飞虱,长丰站,时间 2005.1.1~2009.1.1,可显示相应的曲线图,做出相应的预测。

3 结束语

该研究充分利用 WEBGIS 技术,结合网络数据库及 .NET 技术,设计开发了基于 WEBGIS 的病虫害监测系统。系统基本实现了地图操作、信息查询、预警分析和趋势图的显示等功能,为实现对农产品的生产检测过程提供了有力的工具。

WEBGIS 具有用户平台独立、低成本和易操作等特点,大大降低了技术难度和使用成本,所面向的用户既可以是普通的农户也可以是管理部门。在应用中,系统运行时是可靠的,适应性强,操作简单,在降低系统的建设的费用同时可提高对农产品生产的及时检测和做出方案实施。

应当指出的是,该系统所涉及空间数据的分析和查询功能还有待进一步完善,难以大范围的应用,需要后期设计出新的知识模块去提高系统的利用率,以便更好地查询和分析病虫害的发病时间和发病原因。

参考文献:

- [1] 戴光明,何文斌.万维网地理信息系统构建技术分析[J].现代电子技术,2003(1):46-48.
- [2] ESRI 中国(北京)有限公司. ArcGIS 应用于气象行业解决方案[EB/OL]. 2009-06. <http://download.csdn.net/source/803248>.
- [3] 刘南,刘仁义. Web GIS 原理及其应用——主要 Web GIS 平台开发实例[M]. 北京:科学出版社,2002.
- [4] 官彦萍,黄文江,潘瑜春,等.基于 WebGIS 的作物病虫害监测预报系统构建[J].自然灾害学报,2008,12(6):36-41.
- [5] 刘光,唐大仕. Web GIS 开发——ArcGIS Server 与 .NET[M]. 北京:清华大学出版社,2009.
- [6] 郭巧银,赵传德.基于模型和 GIS 的数字模型系统的设计和实现[J].农业工程学报,2008,24(11):139-144.
- [7] 张洪岩,李瑞改,吴秀芹,等. ArcGIS 9 地理信息系统应用和实践(上册)[M]. 北京:清华大学出版社,2008.

(上接第 223 页)

参考文献:

- [1] Shah P, Póris J F. Peer-to-Peer Multimedia Streaming Using BitTorrent[C]//In IPCCC 2007. New Orleans, USA: [s. n.], 2007.
- [2] Janardhan V, Schulzrinne H. Peer assisted VoD for set-top box based IP network[C]//Proceedings of the 2007 workshop on Peer-to-peer streaming and IP-TV. Kyoto, Japan: [s. n.], 2007:27-31.
- [3] 唐忠平,李晓勇,白英彩.基于 TI DM642 VOD 系统中 OSD 功能的设计与实现[J].微型电脑应用,2005,21(1):43-

46.

- [4] 常明,朱林.计算机图形学基础[M].武汉:华中理工大学出版社,1997.
- [5] 范庆辉,阳富民.基于 TW2824 的 OSD 技术研究与实现[J].计算机工程与设计,2007,28(10):2469-2471.
- [6] 周晓,于鸿洋.有线高清数字电视机顶盒用户操作界面的设计与实现[J].中国有线电视,2007(1):50-54.
- [7] 王瑞胡.基于 OSD 技术的数字电视机顶盒汉字显示解决方案[J].电子设计应用,2003,9(1):102-105.
- [8] 刘菁,于鸿洋.有线数字电视机顶盒界面及 EPG 的设计与实现[J].中国有线电视,2006(3):67-69.