

# 基于 MIPS 平台的中文嵌入式 GUI 的设计与应用

赵振亮, 徐立鸿, 邓梧鹏, 徐惠惠

(同济大学电子与信息工程学院, 上海 200092)

**摘要:** 随着信息时代的到来, 数字视频监控越来越受到人们的重视。根据实际需求开发的一套灵活、美观的图形用户界面与监控系统相搭配就更加需要。对基于 MIPS 平台、以嵌入式 Linux 为操作系统的数字视频监控系统的中文图形用户界面进行了设计与开发, 提出了研究方法。

**关键词:** AT4012; 数字视频监控; 图形用户界面; 芯片显存

## Design and Application of Chinese Embedded GUI Based on MIPS Framework

ZHAO Zhenliang, XU Lihong, DENG Wupeng, XU Huihui

(School of Electronic and Information Engineering, Tongji University, Shanghai 200092)

**Abstract** With the information era coming, the digital video recording system is very important, but developing a suitable and beautiful GUI is more important. This paper describes the research and design of the Chinese graphical user interface, based on MIPS framework and embedded Linux OS.

**Key words** AT4012; digital video recording system(DVR); GUI; FONT RAM

随着信息时代的来临, 数字产品成为社会的主流; 同时人们的安全观念也逐步加强, 数字视频监控系统受到更多人的关注和使用。开发一套系统安全、性能稳定、操作方便的视频监控系统更加重要, 而开发相搭配的图形用户界面更是其中不可或缺的一个环节。

传统的PC硬盘录像机采用视频采集卡工作、功耗大、不稳定、性价比低等不足; 数字视频监控逐渐占据主导地位, 开发网络化的嵌入式视频监控开始成为主流; 由于嵌入式系统资源相对有限就需要开发更加符合要求的GUI, 不仅要求美观、稳定、操作方便, 更要体积小、直观、可靠、占用资源小且反应快速<sup>[1]</sup>, 以适应系统硬件资源有限的条件。

### 1 嵌入式 DVR 系统介绍

嵌入式系统主要由嵌入式处理器、相关支撑硬件、嵌入式操作系统及应用软件系统等组成, 它是集软硬件于一体、可独立工作的系统。本文介绍的嵌入式 DVR 系统具有四路音/视频输入, 一路 Composite(BNC)输出和一路 S-Video 以及 VGA 输出。

系统软件通过 AT4012 显示芯片对视频信号进行分割与合成, 作为预览输出或者传输给 AT2041 芯片进行编码、硬盘存储; 也可以通过 AT4012 芯片对 AT2041 的解码输出信号进行回放; 同时系统还具有网络传输, 支持 USB2.0 数据备份等功能。

本文实现了系统功能用户界面的中文图形显示, 使用户可以灵活方便地操作 DVR 系统, 并可以通过前面板或者遥控器, 实现配置系统性能, 如添加或修改用户资料、选择显示模式、外围设备控制和管理工具等功能。整个界面的字体是按 16×16 点阵格式显示, 可以实现 8 种前台和背景颜色的显示。

### 1.1 嵌入式 DVR 系统硬件

系统选择 AMD 公司的 Au1500(MIPS)高性能 32bit 嵌入式 CPU 作为主处理器, 最高工作频率可达 500MHz; 选择专业多路视频合成显示芯片 AT4012 进行多路视频的分割与合成显示; 运用专业高性能音/视频编解码芯片 AT2041 分别对音/视频进行 MPEG-1 layer III(MP3)和 MPEG-4 编解码。系统还具有 128 MB 的 SDRAM, 16 MB 的 FlashRom 与 32KB 的 NVRAM; PC 与 32bit PCI 总线, 及多媒体 PCIBridge 芯片 SAA7146、USB2.0 与 IDE 等接口。系统硬件框图如图 1 所示。

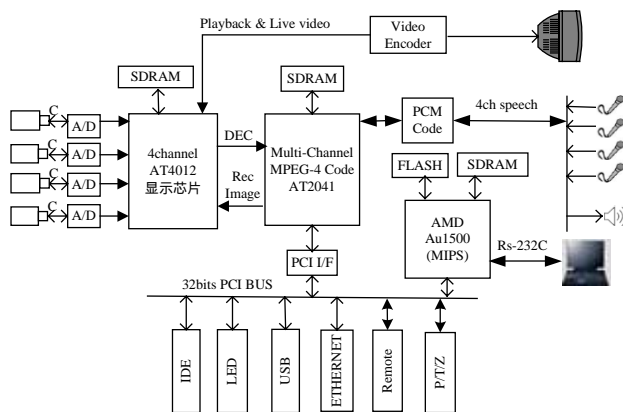


图 1 嵌入式 DVR 硬件结构

**基金项目:** 上海市科委重大专项资助项目(03DZ19303)

**作者简介:** 赵振亮(1982 -), 男, 硕士研究生, 主研方向: 嵌入式 GUI, 网络图像传输; 徐立鸿, 教授、博士生导师; 邓梧鹏、徐惠惠, 硕士研究生

**收稿日期:** 2006-07-08 **E-mail:** ivy\_zhao999@yahoo.com.cn

嵌入式处理器AMD Aul500(MIPS)是该设计的核心,为互联网边缘设备市场上一个高性能、低功耗的片上系统(SOC)<sup>[2]</sup>,主要应用于视频处理和无线电子消费产品。在本系统中与嵌入式Linux操作系统相搭配,能更好地发挥其高性能优势、满足性能稳定、指令精简、处理速度快等要求。

## 1.2 嵌入式 DVR 系统操作系统

本DVR系统采用嵌入式Linux作为操作系统。该系统由裁剪过的内核和根据需要定制的系统模块组成,是一种实时的、支持嵌入式系统应用的操作系统,是嵌入式系统极为重要的组成部分,支持ROM、RAM、FLASH等多种存储器;具有多任务多进程的特征;通信网络支持完整,支持TCP/IP等常见协议,能够实现数据的网络通信和实时处理<sup>[3]</sup>;具有数据库和多媒体支持功能;能根据需求设计,可移植与可裁剪性好;具有一系列开发工具,并可实现各个功能模块化等<sup>[4]</sup>。

本系统的开发平台借助Mont Vista公司的交叉编译工具包Preview kit MVL 3.0 建立。Preview kit提供面向目标板CPU架构的交叉编译环境,同时提供根据目标板CPU定制的Linux内核——Hardhat Linux<sup>[5]</sup>。

## 1.3 系统显示芯片

AT4012是Pent Micro推出的一款多路视频分割和合成显示芯片,具备OSD功能,是本系统中GUI显示的核心芯片;系统的中文菜单显示建立在AT4012的OSD功能之上。OSD RAM对应的就是物理屏幕,存储的字符链接都是在屏幕OSD层上直接显示的字符,大小为3000×13bits,最多可以显示64×48个字符(3076),但是存放入OSD RAM中的字符链接必须是预先存储在Font RAM中的字符;底层最终通过调用系统函数将Font RAM中存储的字模以指定的属性显示在指定位置上。Font RAM为芯片的显示缓存,大小为4000×16bits,最多可以存放256个16×16字模点阵。

## 2 嵌入式 DVR 系统 GUI 的软件设计

嵌入式GUI的设计既要灵活、安全、性能稳定,又要显示美观、反应迅速、更好地满足项目和客户的要求,尤其在中文显示过程中,可能会遇到例如每个汉字的机内码占2B而英文字母为1B、显示延迟等很多问题。所以,设计软件时考虑多方面因素,以更好地设计嵌入式DVR的GUI。

### 2.1 汉字点阵显示格式

国家标准汉字字符集GB2312-80共收集了7445个汉字和图形符号,其标准字库文件HZK16为16×16的国标汉字点阵文件,按汉字区位码的值从小到大依次保存点阵,每个汉字占用32B,以二进制格式存储<sup>[6]</sup>。汉字图形符号根据其位置将其分为94个“区”,每个区包含94个汉字字符,每个汉字字符又称为一个“位”,区的序号和位的序号都是从01到94。在PC的文本文件中,汉字是以机内码的形式存储的,每个汉字占用2B:第1个字节为区码,为了与ASCII码区别,范围从十六进制的0xA1H开始(小于80H的为ASCII码字符),对应区位码中区码的第1区;第2个字节为位码,范围也是从0xA1H开始,对应某区中的第2个位码。将汉字机内码减去0xA0A0就得该汉字的区位码。

### 2.2 AT4012 中文显示原理设计

GB2312的HZK16是标准的16×16字模字库,有2种方法可以直接利用它:(1)直接把HZK16文件(260KB)放入根文件系统中,在需要生成菜单时,直接从文件中读取<sup>[7]</sup>。但是Linux中的read函数被系统调用时需要完成进程从用户态到核心态的转换,占用系统资源大,成本高且效率较低;而且

系统进入系统调用后就不能很好地响应中断,影响系统的实时性。(2)自己制作菜单点阵字库。将所有菜单中的汉字都拷贝到一个文件中,运行在后台编写的一个自动生成字库程序,在相应的目录下就会生成一个新的点阵字库文件:Mylib.h;文件包含了所有菜单中不同汉字的内码和对应的字模,在程序中直接调用这个文件的字模结构。因为菜单上所要显示的字符都是事先确定的,且所有菜单的不同中文字符不超过300个,所以采用第2种方法。

自动生成点阵字模程序代码主要为

```
void get_mat(unsigned char hz[], unsigned char buff[])
{ hzk_p=open("../HZK16.txt", O_BINARY|O_RDONLY);
//打开 HZK16 字库文件
offset=(94*(hz[0]-0xa1)+(hz[1]-0xa1))*32L; //核心算子
lseek(hzk_p,offset,SEEK_SET); //计算字模偏移量
read(hzk_p,buff,32); //读取点阵字模
...}
```

Mylib.h 的数据结构为

```
struct hz_mat //代表单个汉字的数据结构
{ unsigned short incode; //单个汉字的内码 16 bits
  unsigned char mat[32]; //该汉字的字模 32 B};
struct hzlib //代表整个自生成的字库的数据结构
{ int n; //含多少个不同的汉字
  struct hz_mat lib[NUM]; //多个汉字数据结构组成的数组
}
```

### 2.3 系统 GUI 的主要窗口实现

在本系统中一共实现14个中文窗口的显示。每个窗口各自实现不同的功能,共同构成嵌入式DVR的整个用户界面,它们分别为:用户登录窗口,登录后可以调用系统设置窗口;系统设置窗口主要包括:系统信息,图像参数,录像方式及时间设置,检测报警,用户管理和外围设备控制等窗口;在用户管理及检测报警窗口中还可调用添加或删除用户,设置检测区域等底层窗口。

系统主要通过调用程序中的各个菜单结构,实现各种界面菜单的中文显示。主要的菜单结构有:

```
struct{ S16 xstart; //菜单 x 坐标的起始位
        S16 ystart; //菜单 y 坐标的起始位
        S16 color; //字体的显示颜色
        S8 str[50]; //菜单的内容内码
        S16 attribute; //显示的属性
}MENU_FONT_t;
```

通过建立一个结构数组实现各个菜单,数组的每一项为一个菜单实现。具体形式为

```
Static MENU_FONT_t setup_menu_font_chinese[SETUP_MAX_ID][REC_MAX_ITEMS] =
{ { //用户登录窗口 0
  { font_xstart(20), font_ystart(8), FONTCOLOR, "用户登录\0", 0, },
  { font_xstart(12), font_ystart(10), FONTCOLOR, "用户名:( )\0", 0, },
  { font_xstart(18), font_ystart(10), FONTCOLOR, "用户密码:( )\0", 0, },
  { font_xstart(18), font_ystart(12), FONTCOLOR, "...\0", 1, } ...},
};
```

### 2.4 整个系统功能程序设计

系统启动预览和显示进程时,把要显示的中文窗口的汉字内码提取出来,存入一个全局数组并将其中的汉字内码按大小排序且剔除重复的,再根据数组中的机内码在字库文件Mylib.h中查找相应的字模点阵,保存到数组结构

FONT\_RAM \*gp\_fontram 中去, 通过调用芯片 AT4012 的底层驱动把结构数组中的字模载入到 Font RAM 中, 再通过显示函数调用底层驱动在 OSD RAM 中建立与 Font RAM 中字模的连接和设置属性, 在 OSD 上实现字符的显示。系统详细的中文显示流程如图 2 所示。

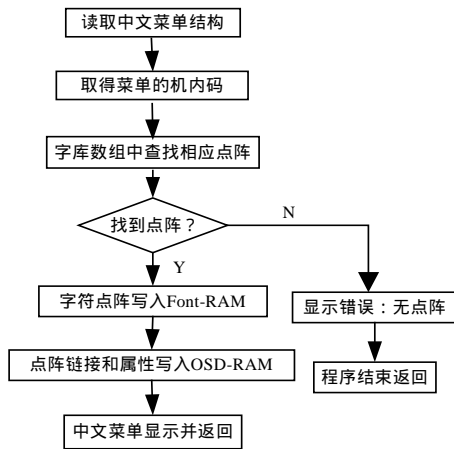


图 2 中文显示流程

显示过程的程序代码实现主要如下所示:

(1) 提取窗口汉字存入全局数组并排序, 根据汉字内码从字库文件 Mylib.h 中提取相应的字模, 并保存到数据结构 FONT\_RAM \*gp\_fontram 的内存区中, 主要实现函数为

```
RETURN get_chnmat_fromstruct (UNS16 incodes[], struct FONT_RAM *m_fontram, // incodes 是 menufont_incode 数组, m_fontram 为 FONT_RAM 结构指针
```

(2) 将 FONT\_RAM \*gp\_fontram 中的汉字字模加载到 AT4012 的 Font RAM 中去; 具体的实现函数为

```
RETURN load_chnfont_toram(FONT_RAM *pfontram, UNS16 norechnfontnum)
{ ...
  if(at4012e_write_font(((ii*16)+(jj/2)+16*ENG_FONT_NUM), &data,1)==FAILURE)
  { //字模点阵按照一定格式存储到 AT4012 的 Font RAM 中
    m_ERROR("Error:In function'load_chnfont_toram'\n");
    fflush(stdout);
    return FAILURE; }
}
```

(上接第 273 页)

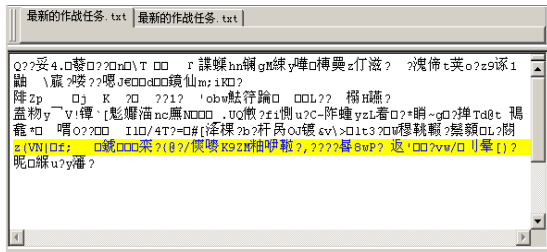


图 8 加密后的文件“最新的作战任务.txt”

串口安全模块的自毁功能通过在面板上点击“自毁”, 串口安全模块程序提示串口协议出错、读写线程出错, 此时表示串口安全模块上运行的服务程序进程已经被关闭并删除, 串口安全模块已经无法与移动计算机上的串口安全模块管理与应用软件进行通信。

通过监视计算机察看串口安全模块内部的情况, 可以看到所以与串口安全模块应用有关的数据、算法和程序都已经

(3) 打印整屏菜单到显示屏幕, 包含了在 FONT RAM 对中英文字符和特殊符号的相对位置的查找, 并将相应字符的位置, 颜色, 显示属性等设置载入到 OSD RAM 中最终显示整屏菜单; 相关函数为:

```
RETURN print_allfont (S16 x_offset, S16 y_offset, UNS8 *p_str, S16 color, S16 blend)
{ ...
  if (at4012e_write_osd(sub_adr, &data, 1) == FAILURE)
  { //字模点阵链接等信息保存到 AT4012 的 OSD RAM 中, 并 //进行最终显示
    m_ERROR("print_allfont.c:error In function '.....'\n");
    fflush(stdout);
    return FAILURE; }
}
```

### 3 总结

在本系统中采用后台程序自己生成字模点阵字库, 调用字库点阵更加方便灵活, 很好地满足了系统显示迅速、灵活安全、稳定性和实时性等要求。本文利用嵌入式 Linux 开发的嵌入式 GUI, 不但使用更加灵活方便, 满足高性能, 低成本和更稳定等要求, 且在同类产品中具有较高技术含量和良好的市场竞争力。本嵌入式数字视频录像机已经投放市场, 设计的 GUI 系统运行简洁稳定且高效率, 很好地满足了人机交互和其他应用的需要。

### 参考文献

- 1 魏永明. 基于 Linux 和 MiniGUI 的嵌入式系统软件开发指南[Z]. IBM Developworks, 2001.
- 2 Dominic S. See MIPS Run[M]. USA: Elsevier Inc., 1999.
- 3 Alessandro R. Linux Device Driver[M]. USA: O'Reilly Media Inc., 2005.
- 4 毛德操, 胡希明. Linux 内核源代码情景分析[M]. 浙江: 浙江大学出版社, 2001.
- 5 Karim Y. Embedded Linux System[M]. USA: O'Reilly Media Inc., 2003.
- 6 Wayne W. 嵌入式计算系统设计原理[M]. 北京: 机械工业出版社, 2002.
- 7 李善平, 刘文峰, 王焕龙. Linux 与嵌入式系统[M]. 北京: 清华大学出版社, 2003.

被毁掉。此时需要重新往串口安全模块烧录服务程序才可以应用串口安全模块。

### 3 结束语

本文针对移动计算机系统信息存储和传输安全水平较低, 提出了一个串口安全模块的设计方案, 实现了信息加/解密、自毁等功能, 满足移动计算机系统无线通信的安全需求, 具有较好的实用价值。

### 参考文献

- 1 冯登国. 密码工程实践指南[M]. 北京: 清华大学出版社, 2001.
- 2 ISO/IEC 7816-4. 识别卡-带触点的集成电路卡-第 4 部分: 交换用行业间指令[S]. 1995
- 3 AIM-Advanced INFOSEC Machine[Z]. 2006-09-10. <http://www.gdc4s.com/aim>.
- 4 Handschuh H, Paillier P. Smart Card Crypto-coprocessors for Public-key Cryptography[J]. CryptoBytes, 1998, 4(1): 6-11.

