

## Ti Synchronous Lyric Display of MP3 Based on Embedded System

ZHANG Gong-li<sup>\*</sup>, YAN Hai-yan, WANG Dong-ming

(Telecommunication Engineering School, Hangzhou Dianzi University, Zhejiang 310018, China)

**Abstract:** A design scheme of a lyric display on a LED screen is firstly realized on the embedded hardware platform in the Linux system. After the introduction of the whole system framework structure, this paper analyzes the programming of the MP3 lyric displayed on a LED screen and the lyrics synchronized application, by combining with the embedded Linux driver program framework. The system has completed multimedia function on a low-end LED display screen, and applies to the requirements for opera group and public teaching or rehearsal.

**Key words:** embedded system; MP3; synchronous lyric display; LED

EEACC: 6130

## 基于嵌入式 Linux 的 MP3 歌词同步显示

张公礼<sup>\*</sup>, 闫海燕, 王东明

(杭州电子科技大学 通信工程学院, 杭州 310018)

**摘 要:** 该系统给出了一种在嵌入式硬件平台及 Linux 操作系统上, 首次实现 MP3 播放与 LED 同步显示歌词的设计方案; 在介绍了本系统的总体构架的基础上, 结合嵌入式 Linux 的驱动程序设计框架, 给出了 MP3 歌词在 LED 显示屏上显示的驱动程序设计, 以及歌词同步的应用程序。本实验完成了在低端 LED 显示屏上实现多媒体化功能的设计, 适用于曲艺团体或公众性场合集体教学或练唱的需求。

**关键词:** 嵌入式系统; MP3; 歌词同步显示; LED

中图分类号: TN919

文献标识码: A

文章编号: 1005-9490(2008)05-1463-03

LED 显示屏从上世纪 80 年代后期在全球迅速兴起, 在短短的十几年中, 技术、产业都取得了长足的进步, 已发展成为重要的现代信息发布媒体手段, 在证券交易、金融、交通、体育、广告等领域得到广泛应用, 随着社会信息化的进程, LED 显示屏在信息显示领域的应用愈加广阔<sup>[1]</sup>。但是目前大多数 LED 显示屏需要上位机对显示过程进行实时控制<sup>[2]</sup>, 并且很难实现多媒体化。本文依据具体要求, 设计了一种优化的高性能、高可靠性的嵌入式大屏幕 LED 显示系统, 将 MP3 播放器直接移植在 S3C2410 上, 脱离了对计算机的依赖, 然后将 MP3 歌词同步显示在 LED 上, 从而首次实现了嵌入式、

LED 以及 MP3 的结合。

### 1 硬件构造

系统的硬件框图如图 1 所示, CPU 采用的是三星公司的 S3C2410, 该处理器内部集成了 ARM 公司 ARM920T 处理器核的 32 位微控制器, 并带有独立的 16 kbyte 的指令 Cache 和 16 kbyte 的数据 Cache, LCD 控制器、RAM 控制器、NAND 闪存控制器、并行 I/O 口、8 路 10 位的 ADC 等等, 主频最高可达 203 MHz。在此基础上, 平台还进行了相应的扩展, 在平台中接入了 CPLD, CPLD 负责对 LED 显示屏进行数据扫描, 同时将数据显示于 LED 上。

收稿日期: 2007-11-15

作者简介: 张公礼(1945-), 男, 教授, 研究生导师, 中国通信学会高级会员, 中国计算机学会模糊逻辑与多值逻辑专业委员会会员, 杭州电子科技大学通信学院网络研究所所长, 研究方向为无线通信、软件无线电技术;

闫海燕(1983-), 女, 研究生在读, 杭州电子科技大学通信学院, 研究方向为多媒体通信与技术, yoyo\_9958@126.com.

本系统的设计中选择了三星公司的 K9F1208 FLASH 作为存储器件以及 1 块三星公司的 K4S561632C SDRAM 芯片, K9F1208 FLASH 芯片是一款 64 M×8 bit 的 NAND Flash, 支持读、擦除、再编程、自动引导模式、硬件错误探测\错误更正等功能, 工作电压为 2.7~3.6 V, 采用 48 脚 TSOP 封装, 块擦除时间为 2 ms, 数据存储有效时间达十年以上。与 Flash 存储器不同, SDRAM 不具有掉电保持数据的特性, 但在存取速度上大大高于 Flash 存储器, 具有读/写的属性, 因此 SDRAM 在系统中主要用于程序的运行空间, 数据及堆栈区, 其存储容量为 16 组×16 Mbit (32 Mbyte), 工作电压为 3.3 V, 封装为 54 脚 TSOP, 兼容 LVTTTL 接口, 支持自动刷新和自刷新, 通过 S3C2410 处理器的内存控制器提供的内存控制信号进行控制。CPLD 根据 SPI 接口传送出来的控制信号, 进行相应的读写操作。如果控制信号是高电平则进行读操作, 反之, 进行写操作, 同时在 CPLD 进行写操作的时候将数据显示于 LED 显示屏上。

S3C2410 内部集成了 USB1.1 控制器, 传输速率为 1.5 Mbit/s, 片内包括两个 USB 控制器, 可设置为两个主机或 1 个主机与 1 个设备, 在本系统中选用的是 S3C2410 的 USB HOST 接口, 通过优盘来读取歌曲及歌词内容的信息。

在本系统中音频接口使用的是 UDA1341 音频芯片。UDA1341 是 Philips 公司推出的音频芯片, 支持 IIS 总线数据格式。IIS 总线具有正常传输、DMA、传输和接收三种工作模式, 本系统主要用到的是它的传输和接收模式, 系统可以通过 UDA1341 将 mp3 文件解码后的数字信号经过 D/A 转换后送入外接的音箱实现播放。

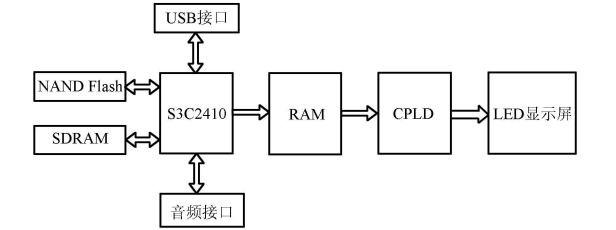


图 1 硬件构造框图

2 软件系统设计

2.1 LED 显示屏的驱动程序设计与实现

由于 Linux 系统具有稳定、高效、易定制、易裁减、开放源码等优点。在 Linux 下设备的控制是通过驱动程序实现的, 设备驱动程序是操作系统内核和机器硬件之间的接口, 设备驱动程序为应用程序

屏蔽了硬件的细节, 这样在应用程序看来, 硬件设备只是一个设备文件, 应用程序可以像操作普通文件一样对硬件设备进行操作。

设备驱动程序是内核的一部分, 它主要完成的功能有: ① 对设备进行初始化和释放; ② 把数据从内核传送到硬件和从硬件读取数据; ③ 读取应用程序传送给设备文件的数据、回送应用程序请求的数据以及检测和处理设备出现的错误<sup>[3]</sup>。

Linux 将设备当作文件处理, 编写 Linux 驱动程序的最主要的工作就是 file\_operations 结构中各入口点的实现, 即对应于 open()、release()、read()、write()、ioctl()等系统调用子函数的编写。同时本系统的设备驱动程序采用灵活性较好的动态加载方式, 驱动程序还应包括模块的初始化函数和模块注销函数。

下面介绍本系统中用到的几个主要的函数及其功能:

注册设备文件系统:

```
ret = register_chrdev (mp3-MAJOR, DE-
VICE-NAME, &mp3-fops);
```

```
devfs-handle = devfs-register (NULL, DE-
VICE-NAME, DEVFS-FL-DEFAULT, mp3-MA-
JOR, 0, S-IFCHR | S-IRUSR | S-IWUSR, &mp3-
fops, NULL);
```

由于本系统考虑的主要是将用户空间的数据传递到内核空间, 同时将所应用到的数据控制口进行相应的设置, 并在数据传输的过程中将相应的数据传给 CPLD。所以在初始化函数中的 register\_chrdev()调用之后, 使用 write 函数来实现用户的数据传递。

write 函数声明如下:

```
static int mp3-write (struct file *filp, const
char *buf, unsigned int count, long long *f_pos)
```

其中参数 buf 是指向存放用户数据数组的指针, 用来将用户数据传递给内核函数的, 参数 count 是共传递的数据的字节数。

在 Linux 中提供了很多函数和宏, 用于内核空间与用户空间之间拷贝数据。本文中用到了 copy\_from\_user, 它负责把数据从用户空间拷贝到内核空间。

S3C2410 的 I/O PORTS 含 GPA、GPB、……、GPH 八个端口。它们的寄存器是相似的: GPxCON 用于选择引脚功能, GPxDAT 用于读/写引脚数据, GPxUP 用于确定是否使用内部上拉电阻 (x 为 A、B、……、H, 没有 GPAUP 寄存器)。本系统根据设

计要求用到了 PORT F 以及 PORT E 的引脚 11、引脚 13。PORT F 的 8 个引脚分别作为数据接口, PORT E 的引脚 11、引脚 13 分别作为控制信号, 控制读入的整个字节的数据和整屏数据。内核提供了两个函数来设置 GPIO 的控制寄存器和数据寄存器, 它们分别是 set-gpio-ctrl write-gpio-bit。在本系统中 set-gpio-ctrl 用于将 GPFCON、GPECON 中的所有引脚设为输出, 同时使用 GpxUP 将相应的引脚设为内部上拉。write-gpio-bit 用于在读取数据以及传送数据时设置数据寄存器。

2.2 MP3 播放与歌词的同步显示

本系统的应用程序主要是完成调用 S3C2410 中的播放器, 在播放歌曲的同时将 MP3 的歌词数据转化成 LED 所需的显示码, 并且将此数据通过调用驱动的 write 函数将数据写入 S3C2410I/O 口的数据寄存器, 然后通过驱动程序的运行将数据传输给 CPLD 并通过 CPLD 的扫描将数据显示在 LED 显示屏上。

在此应用程序中主要考虑的是如何将数据同步显示的问题, 不仅要将相应的歌词转化为 LED 显示屏所要求的显示码, 而且要将歌词解析为同步的显示数据。当前, lrc 文件作为一种能实现唱词同步显示的数据格式, 也是本系统所采用的最佳的文件格式。事实上, .lrc 文件可以看成是以一定格式写的. txt 文件, 歌词的正文部分是一个或多个时间标志对应一句歌词, 单句歌词以 \n 结尾, 最后一句歌词无 \n, 时间标志用 [] 括起来, 时间形式是 "xx: xx. xx", 歌曲信息如歌曲名、作者之类用 [] 括起来放在正文的前面。

歌词播放的原理为, 找到时间标志, 将其与当前播放时间比对, 如果一致则显示该时间标志对应的歌词, 为方便起见没有对歌曲信息做处理(如果要显示歌曲信息, 可以通过比对关键字如 "ti: ", "ar: " 等来获取)。

.lrc 文件如下:

[ ti: 六月的雨  
[ ar: 胡歌  
[ al: 仙剑奇侠传  
[ by: sunteng]  
[ offset: 500]  
[ 00: 12. 37]《仙剑奇侠传》插曲  
[ 01: 59. 30] [ 00: 21. 00] 一场雨 把我困在这里  
[ 02: 03. 12] [ 00: 24. 73] 你冷漠的表情 会让我伤心  
[ 02: 07. 72] [ 00: 29. 61] 六月的雨 就是无情的你  
[ 02: 12. 70] [ 00: 34. 42] 伴随着点点滴滴 痛击我心里

息的读取及码字转换, MP3 歌曲的播放, 歌词的同步显示, 系统总的流程图如图 2 所示。

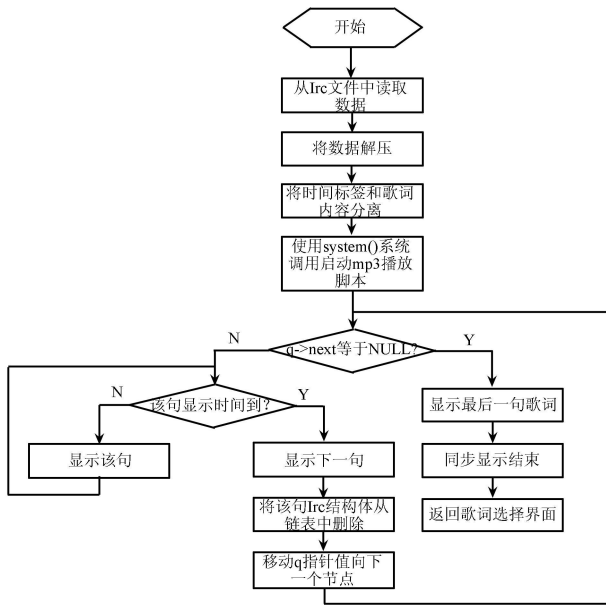


图 2 系统流程图

(1) 在歌词信息的读取模块中, 用到的是一个数组结构以及数据结构体, 在不同的数组中分别存放了时间信息和歌词内容<sup>[4]</sup>, 有:

```
struct lrc {  
    float time; 用来确定一句歌词所需的时间;  
    int len; 用来确定每句歌词的长度, 这个长度就是用来判断一句歌词是否结束;  
    gecineirong[] : 歌词的具体内容就存放在此;  
    buf_hzk16[] : 歌词转换的显示码的存放;  
    buf[] : 存放的是歌词的 LED 最终显示码;  
};
```

在这个模块中主要实现的是将歌词数据依据要求转换为区位码, 再由区位码来确定准确的显示码, 由于本系统的 LED 显示屏采用的是共阴极的, 所以还要对得到的数据进行取反。在进行歌曲播放之前, 要将所要显示的数据进行排序, 以达到完整文字显示的效果。

(2) MP3 歌曲的播放是通过在 S3C2410 上移植一个 madplay 的播放器来实现播放的。首先从网上下载: madplay-0.15.2b.tar.gz, libmad-0.15.1b.tar.gz, libid3tag-0.15.1b.tar.gz 和 zlib-1.1.4.tar.gz 四个压缩文件。然后在我们建立的交叉编译环境的根目录下建立一个名为 mp3 的目录, 并把它们都解压到这个目录下面, 同时还要在 mp3 这个目录下建立一个名为 libz 的目录, 作为存放所生成的文件的地方。根据具体的步骤进行文件的安装和解压即可<sup>[5]</sup>。

因此, 在本应用程序中主要分为 3 部分: 歌词信

(下转第 1471 页)

- PRESS, 2003, 4-6.
- [7] Behzad Razavi. Design of Analog CMOS Integrated Circuits [M]. Xi'an Jiaotong University Press, 2002, 47-49.
- [8] Kuboki S, et al. Nonlinearity Analysis of Resistor String A/D Converters[J]. IEEE Transactions on Circuits and Systems, 1982, 29(6): 383-390.
- [9] Baker R J, Li H W, Boyce D E. CMOS Design of Layout and Simulation [M]. China Machine Press, 2003, 132-137, 685-703.
- [10] Venes A G W, Van R J de Plassche. An 80 MHz 80 mW 8-b CMOS Folding ADC with Distributed Track-and-Hold Pre-processing[J]. IEEE J. Solid-state Circuits, 1996, 31(12), 1846-1853.
- [11] Figueiredo P M, Vital J C. Kickback Noise Reduction Techniques for CMOS Latched Comparators [J]. IEEE Transactions on Circuits and Systems-II, 2006, 53(7): 541-545.
- [12] Allen P E. CMOS Analog Circuit Design [M]. New York Oxford; Oxford University Press, 2002, 682-686, 483-487.
- [13] Van R de Plassche. CMOS Integrated Analog-To-Digital and Digital-To-Analog Converters [M]. Boston, MA; Kluwer, 2003, 485-499.
- [14] Doernberg J, Gray P R and Hodges D A, A 10 bit 5 Msample/s CMOS Two-step Flash ADC [J]. IEEE J. Solid-State Circuits, 1989, 24(2): 241-249.
- [15] Kinget P, Steyaert M. Impact of Transistor Mismatch on the Speed-Accuracy-Power Trade-off of Analog CMOS Circuits [C]//Proceedings Custom Integrated Circuits Conference, 1996, 333-336.
- [16] Sedra A S and Smith K C. Microelectronic Circuit [M]. 4th ed. New York; Oxford University Press, 1988.
- [17] Millman J and Halkias C C. Integrated Electronics: Analog and Digital Circuits and Systems. Now York; McGraw-Hill, 1972.
- [18] Robins W P, Phase Noise in Signal Source [M]. London; Peter Peregrinus, 1982.
- [19] Portmann C L, Meng T H Y. Power-Efficient Etestability Error Reduction in CMOS Flash A/D Converters [J]. IEEE J. Solid-State Circuits, 1996, 31(8): 1132-1140.
- [20] Michael J. Demler. High-Speed Analog-to-Digital Conversion [M]. San Diego; Academic Press, INC, 1991, 31-35.

(上接第 1465 页)

在应用程序中写了一个 shell 脚本文件通过 system() 系统用来直接调用程序中要播放的文件, 使播放器可以直接找到需要播放的文件, 即时的播放所需的歌曲。

(3) 歌词同步显示的应用程序中, 主要是通过一个判断语句来执行时间上的同步的。通过定义一个时间流逝变量 timeif, 将 timeif 与相应的每一句歌曲所需的时间进行比较: if (timedif \* 10 <= (1000 \* q -> next -> time)), 来判断应该同步的是那一句歌词, 以及歌词显示的时间, 这样可以比较准确地将所需的歌词及时的传送给 CPLD, 通过 CPLD 的数据接收, 然后直接将要显示的歌词显示在 LED 显示屏上。

### 3 结束语

本系统实现了对 mp3 文件的解码和播放, 并对现有 Lrc 文件进行解析, 实现了嵌入式系统下的 mp3 唱词同步显示。此设计方案真正实现了 LED 舞台屏在脱离计算机后的多媒体化, 使得嵌入式系

统的应用真正与 LED 显示屏结合起来, 不仅可以在室外播放歌曲供人欣赏, 而且可以满足爱好者集体练唱、教学的需求。

### 参考文献:

- [1] 关积珍, 陆家和. LED 显示屏的技术及其市场和产业发展 [J]. 激光与红外, 2003, (06).
- [2] 熊雨凯, 吴光敏. 嵌入式大屏幕 LED 显示屏的设计与实现 [J]. 现代电子技术, 2006, (22).
- [3] Stevens W R. UNIX 环境高级编程 [M]. 北京: 机械工业出版社, 2002; 142-146.
- [4] Alessandro Rubini 著. 魏永明译. Linux 设备驱动程序 [M]. 北京: 中国电力出版社, 2002.
- [5] 陆志洋. MPEG-1/ Audio Layer3 解码器实现及应用 [D]. 大连: 大连海事大学, 2006.
- [6] 阳进. 基于单片机的 LED 显示屏的汉字显示 [J]. 中国科技信息, 2005, (12).
- [7] Promod Chandraiah. Specification and Design of a MP3 Audio Decoder [D]. California: California university, 2005.
- [8] O'REILLY TAIWAN. Building Embedded Linux System [M]. 北京: 中国电力出版社, 2005.