

文章编号:1008-7826(2009)01-0039-05

基于嵌入式 Linux 与 QT 的 MP3 播放器的设计

王灵芝, 陈磊松

(漳州师范学院 物理与信息工程系, 福建 漳州 363000)

摘要: 本文讨论了一种基于嵌入式 Linux 与 QT 的 MP3 播放器的设计方法. 系统完成了在 Linux2.6 下对语音解码芯片 UDA1341、按键等驱动程序的编写. 使用 QT 制作图形用户界面, 利用开源的 Madplay 解码器对 MP3 文件进行软件解码, 实现了播放和暂停、快进快退、上下曲、音量增减、显示歌曲状态信息等功能. 该系统界面友好, 功能完善.

关键词: A 嵌入式 Linux ; QT ; MP3 播放器

中图分类号: TP368.1 **文献标识码:** A

1 引言

MP3 作为高质量音乐压缩标准, 给音频产业带来了具大的冲击. MP3 技术使音乐数据压缩比率大, 回放质量高, 较小数据量和近乎完美的播放效果使其在网络上传输得以实现. 随着 MP3 播放器的出现及其技术的发展, 对 MP3 播放器的要求越来越高, 制造商在选型、设计、开发、附加功能和适用领域等方面做了很多努力, 设计了多种方案. 对 MP3 文件进行解码可以采用软件解码和硬件解码两种方法. 硬件解码如采用专用解码芯片 MAS3507D, 实现简单但增加硬件成本. 软件解码需占用大量 CPU 时间, 实现难度较大, 但硬件成本低、处理灵活^[1].

Samsung 公司推出的基于 ARM9 内核 32 位 RISC 处理器 S3C2410, 最高可运行在 203MHz. 该芯片的功能强大, 完全能够满足 MP3 定点或浮点解码程序的实现, 还能够将 Linux 操作系统移植进入该芯片, 实现包括播放 MP3 在内的多进程, 多任务处理, 是一种理想的解决方案.

Qt/Embedded 是基于 Qt 的嵌入式 GUI 和应用程序开发的工具包, 它可运行多种嵌入式设备上, 主要运行在嵌入式 Linux 系统上, 为嵌入式应用程序提供 Qt 的标准 API. 利用 QT 设计 MP3 图形用户界面, 可以大大提高人机交互的友好性和美观性.

2 硬件设计

系统总体框图如图 1 所示: 包括 S3C2410 中央处理器、外部存储器、串口、USB 主机、红外通讯口、音频接口、存储接口、LCD 和触摸屏接口、十六个小按键、时钟源、复位电路、调试及下载接口.

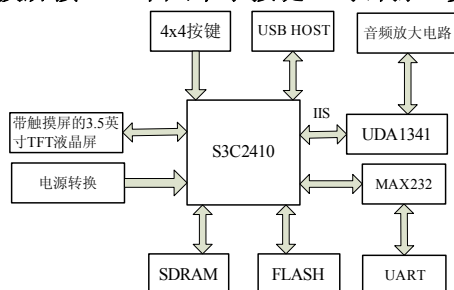


图 1 系统硬件总体结构

收稿日期: 2008-09-18

基金项目: 漳州师范学院科研项目基金项目 (SK07002).

作者简介: 王灵芝(1981-), 女, 福建省建瓯市人, 硕士, 助教.

处理器模块: S3C2410 拥有丰富的处理器资源: 独立的 16KB 指令 Cache 和 16KB 数据 Cache、MMU、支持 TFT 的 LCD 控制器、NAND 闪存控制器、3 路 UART、4 路 DMA、4 路带 PWM 的 Timer、I/O 口、RTC、8 路 10 位 ADC、触摸屏接口、IIC 总线接口、IIS 总线接口、2 个 USB 主机、1 个 USB 设备、SD 主机和 MMC 接口、2 路 SPI。为手持设备和一般类型应用提供了低价格、低功耗、高性能小型微控制器的解决方案。

UART 模块: 用于开发调试控制和返回调试信息。

USB 主机模块: 连接 U 盘等存储设备为 MP3 播放器提供歌曲来源。

音频接口: 采用 IIS 接口芯片 UDA1341, 一路立体声音频输出接口可接耳机或音箱, 通过解码后的信号经音频放大电路输出。

LCD 和触摸屏接口: 显示 MP3 播放控制界面, 通过触摸屏实现对播放和暂停, 停止, 音量递增, 音量递减等功能的控制。

按键接口: 除了利用触摸屏外, 还可以通过按键控制 MP3 的播放过程。

3 系统软件体系结构

此 MP3 播放器的软件体系结构采用分层模式, 总共包含四层: 硬件层、设备驱动层、操作系统及应用层。如图 2 所示:

硬件层包括了 MP3 播放器所需的物理设备: USB 接口、音频接口芯片 UDA1341TS、3.5 英寸的 TFT 液晶屏、十六个小按键等; 设备驱动层包括上述各个设备的驱动程序的编写; 操作系统层我们移植了 Linux2.6 的操作系统, 由操作系统来统一管理各个硬件设备; 最上层是应用层, 我们使用 QT 设计图形用户界面, 通过移植开源的 Madplay 解码器实现对 MP3 文件的解码, 播放、音量增减的控制。

嵌入式 Linux(Embedded Linux)是指对 Linux 经过小型化裁剪后, 能够固化在容量只有几十万字或几十亿字节的存储器芯片或单片机中, 应用于特定嵌入式场合的专用 Linux 操作系统。通常它的体积小, 性能稳定, 源代码免费, 将大大减少开发者的成本, 更具市场竞争力。

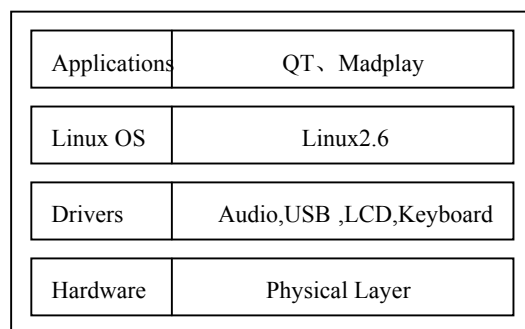


图 2 系统硬软件框图

最基本的嵌入式 Linux 系统需要 4 个基本元素: (1) 引导加载程序: 包括固化在固件中的启动代码和 Bootloader 两大部分, 用于完成机器加电后的系统定位引导。(2) Linux 系统内核: 为嵌入式应用提供一个软件环境, 为应用程序完成基本的底层的资源管理工作。(3) 文件系统: 包括根文件系统和 CRAMFS 文件系统的建立。(4) 为了最小嵌入式系统具有一定的实用性, 还需加上硬件驱动程序及一个或几个应用进程以提供必要的应用功能支持^[2]。

本设计的宿主机 Linux 操作系统是建立在 Vmware 虚拟机中, 这样可以灵活的与我们熟悉的 Windows 操作系统进行数据共享, 加快开发速度。之后在宿主机上设置各种环境变量, 建立交叉编译调试的开发环境。对于 Linux2.6 系统, 我们采用的是 3.3.2 版本的 arm-gcc 交叉编译工具, 使用它编译 linux2.6.x 版本的内核源码; 另外编译 QTE 嵌入式应用也需要 3.3.2 以上版本的交叉编译工具。

4 设备驱动程序的编写

设备驱动是 Linux 内核的重要组成部分, 在 Linux 操作系统下有 3 类主要的设备文件类型: 字符设备、块设备和网络设备。编写好的设备驱动程序可以编译到内核中, 在系统启动时与内核一起启动, 这样的方

法在嵌入式Linux系统中经常被使用。在通常情况下设备驱动的动态加载更加普遍,开发人员不必在调试过程中频繁启动机器就能完成设备驱动的开发。

4.1 linux的设备驱动程序与外界的接口可以分为三个部分:

(1)驱动程序与系统引导的接口:设备驱动在加载时首先需要调用入口函数 `module_init()` 完成对设备驱动的初始化以及向内核注册该设备,例如对字符设备调用 `register_chrdev_region()` 完成注册.注册成功后,该设备将获得系统分配的主设备号、自定义次设备号.同样设备驱动程序在卸载时需要回收相应的资源,注销设备;字符设备调用 `unregister_chrdev_region()` 实现。

(2)驱动程序与设备的接口:这部分描述了驱动程序如何与设备进行交互,与具体的设备密切相关。

(3)驱动程序与操作系统内核的接口:通过 `file_operations` 数据结构来完成的.系统对设备的操作包括: `open`、`read`、`write` 和 `ioctl` 等。

4.2 按键驱动

按键驱动属于字符型设备驱动,设计中采用的是中断扫描法读取键值.调用函数 `request_irq` 来申请中断,卸载时调用 `free_irq` 用来释放中断.此外函数 `disable_irq` 为中断禁止, `enable_irq` 为中断使能。

在对I/O端口的操作中,首先调用 `ioremap_nocache` 函数将I/O内存资源的物理地址映射到核心虚地址空间,之后就可以像读写内存一样读写I/O资源.调用 `writel` 函数向I/O写数据,调用 `readl` 函数向I/O读数据.其中l代表32的端口,而b和w分别代表8位和16位端口。

4.3 音频驱动电路

S3C2410 通过 IIS (Inter-IC Sound bus) 总线与音频控制芯片 UDA1341 进行通信. IIS 是飞利浦公司提出的串行数字音频总线协议. IIS 总线只处理声音数据,使用了四根串行总线: 串行数据输入(IISDI)、串行数据输出(IISDO)、左右声道选择(IISLRCK)和串行位时钟(IISCLK). 系统通过 IIS 总线对 UDA1341 的各寄存器进行初始化. 并通过 UDA1341 外接麦克风和耳机等音频输入输出设备. 由于 IIS 总线只处理音频数据, 因此 UDA1341 还内置了用于传输控制信号的 L3 总线接口. L3 接口相当于混音器控制接口, 可以控制输入/输出音频信号的低音及音量大小等。

音频设备驱动程序主要通过通过对硬件的控制实现音频流的传输,同时向上层提供标准音频接口.音频接口驱动向上提供2个标准接口: 数字音频处理 (DSP), 负责音频数据的传输即播放数字化声音文件和录音操作等; 混音器(MIXER), 负责对输出音频进行混音处理, 如音量调节、高低音控制等. 这2个标准接口分别对应设备文件 `dev/sound/dsp` 和 `dev/sound/mixer`. 整个音频驱动的实现分为初始化、打开设备、DSP驱动、MIXER驱动和释放设备等部分:

(1)设备初始化主要完成对 UDA1341 音量、采样频率、L3 接口等的初始化, 并且注册设备、打开设备. 为设备分配 DMA 通道, 根据采样参数计算出缓存内段的大小. 当缓存区和 DMA 设置好后, 读写操作主要对缓存操作。

(2) DSP 驱动的实现: DSP 驱动实现了音频数据的传输即播放和录音的数据传输. 同时提供 `ioctl` 对 UDA1341 中的 DAC 和 ADC 采样率进行控制。

(3) MIXER 驱动的实现: MIXER 驱动只控制混音效果, 并不执行读写操作, 通过 `ioctl` 函数完成混音效果、音量的增减、静音等功能。

5 QT/Embedded 程序的编写

QT/Embedded 是一个为嵌入式设备提供的图形用户接口和应用开发而定制的 C++ 工具开发包, 提供了信号与插槽的组件化编程机制和多种多样的图形设计、事件处理、进程控制的类^[3]。

QT 开发环境的建立需要包括三个工具包:

(1) `tmake 1.11`: 用于生成 Qt/Embedded 应用工程的 Makefile 文件。

(2) Qt/Embedded 2.3.7:Qt/Embedded 安装包。

(3) Qt 2.3.2 for X11:Qt 的 X11 版的安装包, 将产生 x11 开发环境所需要的两个工具。

本系统使用 QT/Embedded 制作图形用户界面, 利用开源的 Madplay 解码器对 mp3 文件进行软件解码, 控制界面如图 3 所示, 其功能包括打开文件、播放和暂停、快进快退、上下曲、音量加减、显示时间进度、显示歌曲路径、歌曲名、总的播放时间、各种状态信息、进度条等。下面就一些重要功能的实现做简要的介绍。

5.1 桌面图标的实现

QT 提供了建立各种各样图形和按钮的类, 如 QLabel、QPushButton、QLCDNumber、QProgressBar 等, 还有一些调整图形和按钮位置、大小的类, 如 setGeometry、resize 等, 通过继承这些类, 我们可以绘制出多功能美观的控制画面。

5.2 外部事件的响应

当用户通过触摸屏单击某个按钮时, QT 的窗口在事件发生后将激发信号如 clicked(), 可以通过调用 connect() 函数把这个信号和一个插槽(函数)连接起来, 完成对事件的响应。对于其他外部的事件(如按键), 可以通过 read 函数来读取设备的值, 并根据返回值转向相应的程序。

5.3 MP3 文件打开功能

QT 中的 QFileDialog 类提供了允许用户选择文件或者目录的对话框。我们可以通过继承这个类, 调用其中的 getOpenFileName 函数, 实现建立打开 MP3 文件和选择文件的对话框:

```
QString fileName=QFileDialog::getOpenFileName(".",fileFilters,this);
```

其中 fileFilters 指文件过滤器, 只有符合 fileFilters 中设置的文件才能被读出。该函数将返回所选择的 MP3 文件的文件名和路径到 fileName 中。

5.4 MP3 播放功能

通过调用开源的 MADPLAY 解码程序对打开的 MP3 文件进行解码播放。MADPLAY 程序播放歌曲的命令是: ./MADPLAY filename

相应程序段如下:

```
int pid=fork(); //操作系统创建一个新的进程
```

```
if(pid==0); //如果进程号为 0, 说明子进程创建成功
```

```
{execl("/tmp/udisk/MADPLAY","./MADPLAY",namedesc,NULL);} //子进程创建成功后, 调用外部的 MADPLAY 解码程序播放歌曲。
```

5.5 上下曲功能

实现上一曲播放功能时, 利用 QStringList 类打开 MP3 存放路径中的 MP3 歌曲列表:

```
QStringList files=dir.entryList("*.mp3",QDir::Files);
```

该函数将返回播放列表, 将当前播放的文件与列表中文件一一比较, 并且计数, 当比较到相同的文件后, 将此时的计数值减 1, 就得到了下一曲的文件名, 然后将文件播放。

5.6 快进快退播放功能

实现快进快退功能前, 必须先停止当前播放的进程。然后提取当前播放的时间, 在这个时间的基础上加 5 秒或减 5 秒。调用以下 MADPLAY 命令可实现从歌曲的某个时间点开始播放: ./MADPLAY -time=HH:MM.DDD (HH:MM.DDD 为开始播放的时间)

相应的程序段如下:



图 3 基于 QT 的控制界面

```

printf(killer,"kill -9 %d",pid2); //pid2是当前子进程(歌曲播放进程)的进程号
system(killer);                  //系统调用 killer 的外部命令, 执行结束子进程
.....
execl("/tmp/udisk/MADPLAY","./MADPLAY",advancetime,musicfile,NULL);

```

5.7 音量调节功能

通过调用 UDA1341 的驱动程序 mixer 接口的 ioctl 函数, 可以实现音量控制功能, 相应的程序段如下:

```

Int mixer_fd; mixer_fd = open("/dev/sound/mixer", O_RDONLY); //打开 UDA1341 的驱动设备 mixer 接口
arg+=3; //arg 是音量码级的存储, 加音量时每次加 3, 减音量时每次减 3
ioctl(mixer_fd, SOUND_MIXER_WRITE_VOLUME, &arg); //对 UDA1341 进行控制, 调节音量

```

5.8 MP3 歌曲计时功能

计算 MP3 歌曲的时间, 必须先读取 MP3 文件的数据流, MP3 帧头的第二个字节里保存着 MP3 版本信息和 Layer 版本信息; 第三个字节保存比特率信息, 通过查表可以获帧头中的比特率, 然后计算 MP3 的时间^[4]. MP3 的比特率 rate 与播放时间 time 符合如下公式:

$$\text{Time} = (\text{M}/1024 \times 8) / \text{rate}; \text{ (其中 M 等于 MP3 文件的大小)}$$

相关程序段如下:

```

f.open(musicfile2, ios::in|ios::out|ios::binary); //定位 MP3 数据流的第三个字节
//打开 MP3 文件的数据流
f.seekp(ios::beg+1); //读入第三个字节
//定位 MP3 数据流的第二个字节
f.read ( mem, 16); //读入第二个字节
printf(mp3data, "%X", *mem); //查表求出比特率
strcpy(bitrate1, mp3data); //读入第二个字节
f.seekp(ios::beg+2); //读入第二个字节
f.read ( mem, 16); //根据公式计算时间

```

6 结论

本文完成基于嵌入式 Linux 与 QT 的 mp3 播放器的设计. 实现了 Linux2.6 下对硬件驱动程序的编写, 使用 QT 制作图形用户界面, 界面友好, 功能完善.

参考文献:

- [1] 李 冰, 孙建平, 谭 悦. 基于嵌入式 Linux 与 S3C2410 的网络视频监控[J]. 华北电力大学学报, 2006, 33(4): 9-11.
- [2] 孙天泽, 袁文菊. 嵌入式设计及 Linux 驱动开发指南[M]. 北京: 电子工业出版社, 2007.
- [3] Alan Ezust, Paul Ezust, 李仁见, 战晓明. C++设计模式—基于 Qt4 开源跨平台开发框架[M]. 北京: 清华大学出版社, 2007.
- [4] 汪 勇, 熊前兴. MP3 文件格式解析[J]. 计算机应用与软件, 2004, 12: 126-128.

Design of MP3 Player Based on Embedded Linux and QT

WANG Ling-zhi, CHEN Lei-song

(Department of Physics and Electronic Information Engineering, Zhangzhou Normal University, Zhangzhou, Fujian 363000, China)

Abstract: Based on embedded Linux and QT, this paper proposes a design method of MP3 player. Under Linux 2.6 kernel, the system accomplishes the driver codes of audio CODEC UDA1341 and key-press, etc. Using the QT cross-platform application framework, this project designs a graphical user interface. The system uses the open-code Madplay audio decoder to decode MP3 files, and implements these primary functions: play and pause, move back, skip forward, adjust volume, and song statue show, etc. This multifunctional system has friendly interface.

Key words: Embedded Linux ; QT ; MP3 Player

[责任编辑: 喻玉萍]