

# 基于 ARM 平台的播放器 MPlayer 的移植<sup>\*</sup>

刘 爽 刘传菊

(仲恺农业工程学院信息学院 广州 510225)

**摘 要** 该设计主要是要实现播放器 MPlayer 在 ARM 平台上的移植。通过在 PC 上编译,使其能在 PC-Linux 下实现,然后通过交叉编译形成可以在 ARM-Linux 上运行的执行程序,再向 ARM 平台移植。该设计的创新点在于进行了 MPlayer 播放器向更便携、多格式播放方面发展的研究。Linux 系统具有很好的可移植性,它的实现过程以及核心的代码对类似应用具有很好的可重用性,只需通过较小的修改就能移植到不同的操作系统和平台上。

**关键词** ARML; Linux; MPlayer; 移植

**中图分类号** TP391

## Mplayer Transplantation Based on the ARM Platform

LIU Shuang LIU Chuanju

(Information College, Zhongkai University of Agriculture and Engineering, Guangzhou 510225)

**Abstract** This is a design to achieve the player named MPlayer on an ARM platform transplantation. Compiled by on a PC, make it can realize in the PC-Linux, then through cross-compiling formation can run on Linux in the ARM-the execution procedures, again to ARM platform transplantation. This paper's innovation points is a more portable, the MPlayer can play more than the research of the development of the format to mp3 aspects. Linux system has good portability, its realization process and the core code for similar application has very good reusability, just through the small changes can transplanted into different operating systems peace table.

**Key Words** ARM, Linux, MPlayer, transplantation

**Class Number** TP391

## 1 引言

科技的发展日新月异,人们的生活也随之发生着快速的变化,对高科技产品的追求越来越多,对产品的功能和性能指标等要求越来越高。随着多媒体终端作为便携产品进入人们的日常生活,嵌入式操作系统 Linux 以其开源、易于移植、内核可裁剪等优势成为不可或缺的一部分。

在操作系统上开发媒体播放器已经成为当今便携产品中的技术热点之一。其追求的目标包括产品的先进性、价格的低廉性、项目的可移植性以及良好的播放效果等。

S3C2440 微处理器是一款由 Samsung 半导体公司推出的高性能、低功耗、高集成度并具有工业级温度范围和性能的微处理器。Mini2440 是一款低价实用的 ARM9 开发板,它采用 Samsung S3C2440 为微处理器,并采用专业稳定的 CPU 内核电源芯片和复位芯片来保证系统运行时的稳定性。

嵌入式微处理器是由通用计算机中的 CPU 演变而来的,具有较高的性能,在实际的嵌入式开发中,只保留和嵌入式应用有关的硬件功能,这样可以大大减小系统体积和功耗,以满足嵌入式应用的特殊要求。

由于 IT 技术发展的推动,32 位 ARM 体系结构已经成为一种事实上的标准,随着高端 32 位 CPU 价格的不断下

降和开发环境的成熟,促使 32 位嵌入式处理器日益挤压原先由 8 位微控制器主导的应用空间。随着 ARM 处理器在全球范围的流行,32 位的 RISC 嵌入式处理器已经开始成为中高端嵌入式应用和设计的主流。

本设计基于 ARM 平台,使用 Linux 操作系统进行对媒体播放器的移植,实现多格式音视频文件的播放。

## 2 系统开发环境的构建

### 2.1 软件环境构建

构建嵌入式开发环境包括安装 Linux 系统,安装嵌入式工具链,配置宿主机相关信息以及和开发板之间进行连接等关键步骤。

软件环境构建中需要在 Windows 下安装虚拟机,建立一个虚拟机需要指定硬盘、内存等,在 virtualbox 中可以使用实际硬盘也可以使用文件来模拟硬盘。本设计基于 Fedora12 进行开发,在装好的虚拟机上安装 Fedora 12 i386 映像文件。安装交叉编译工具 arm-linux-gcc3.3.2。来修改环境变量:

```
[root@localhost~]# vi .bash_profile
```

在 PATH=\$PATH:\$HOME/bin 后面增加:/mplayer/usr/local/arm/3.3.2/bin 后保存退出

<sup>\*</sup> 收稿日期:2011 年 12 月 11 日,修回日期:2012 年 1 月 16 日

基金项目:广东省科技计划资助项目“新农村农业信息化综合解决方案”(编号:2010B020315027)资助。

作者简介:刘爽,女,硕士,讲师,研究方向:信息管理与信息系统。刘传菊,女,教授,研究方向:计算机应用。

运行 `source /root/. bash_profile`;

`[root@localhost ~]# source /root/. bash_profile`

## 2.2 硬件环境构建

主机:PC机;目标板:OMAPL138(CPU:ARM926EJ-S);外围设备:电源;串口线;网线。

## 3 系统的设计与实现

### 3.1 在Linux下安装Mplayer

需要用到下面四个压缩包:

```
MPlayer-1.0pre8.tar.bz2
all-20071007.tar.bz2
windows-all-20071007.zip
Blue-1.4.tar.bz2
```

进入主目录,完成用户转换。

在/usr/lib下新建目录codecs和wincodecs:

`[root@localhost ~]# mkdir /usr/lib/codecs`

`[root@localhost ~]# mkdir /usr/lib/wincodecs`

以上两个文件夹是用来存放解码文件库的。

解压文件并移动文件:

`[root@localhostmplayer]# tar jxvf all-20071007.tar.bz2`

`[root@localhostmplayer]# mv all-20071007/* /usr/lib/codecs`

`[root@localhostmplayer]# unzip windows-all-20071007.zip`

`[root@localhostmplayer]# mv windows-all-20071007/* /usr/lib/wincodecs`

安装MPlayer:

解压:`[root@localhost mplayer]# tar jxvf MPlayer-1.0pre8.tar.bz2`

进入文件目录:`[root@localhost mplayer]# cd MPlayer-1.0pre8`

输入:

`./configure --prefix=/usr/mplayer --enable-gui --enable-freetype --with-codecsdir=/usr/lib/codecs --with-win32libdir=/usr/lib/wincodecs --language=zh_CN`

配置参数说明:

`--prefix=/usr/mplayer` 是安装路径

`--enable-gui` 安装图形化用户界面

`--enable-freetype` 调节字体

`--with-codecsdir=/usr/lib/codecs/`

`--with-win32libdir=/usr/lib/wincodecs` 指定解码

位置

`--language=zh_CN` 中文

编译:make,然后安装:make install。

安装皮肤并通过链接设为默认皮肤:

`tar vjxf Blue-1.4.tar.bz2`

`mv Blue /usr/local/mplayer/share/mplayer/Skin/`

`cd /usr/local/mplayer/share/mplayer/Skin/`

`ln -s Blue default`

### 3.2 交叉编译MPlayer

得到编译参数:

`[root@localhost MPlayer-1.0pre8]# ./configure -help`

运用编译指令进行交叉编译:`[root@localhost MPlayer-1.0pre8]# ./configure --host=cc=gcc --cc=arm-linux-gcc --target=arm-linux --enable-static --disable-win32 --disable-dvdread --enable-fbdev --disable-mencoder --disable-liba52 --disable-live --disable-mp3lib --disable-mad`

交叉编译参数说明:

`--host=cc=gcc` 使用自带的编译器来编译一些需要在主机上执行的文件

`--cc=arm-linux-gcc` 运用自己安装的 arm-linux-gcc3.3.2 来交叉编译

`--target=arm-linux` 目标板是 arm-linux

`--enable-static` 静态连接

`--disable-win32` 不支持 win32 设备

`--disable-dvdread` 不支持 dvdread 库

`--enable-fbdev` 使能缓冲帧设备

`--disable-mencoder` 不支持 mencoder 编码方式

`--disable-liba52` 不使用 liba52 库

`--disable-live` 不支持 live.com 流媒体

`--disable-mp3lib` 不使用 mp3 库

`--disable-mad` 不使用 mad 库

编译通过,生成一个可执行文件 mplayer,以下是它的详细信息:`-rw-r--r-- x.1 root root 11808930 某月某日时:分 mplayer`。

### 3.3 Bootloader的移植

Bootloader 可以分为以下两种操作模式:

1) 启动加载模式 (bootloading)

上电后,bootloader 从板子上的某个固态存储设备上将操作系统加载到 RAM 中运行,整个过程并没有用户的介入。产品发布时,bootloader 工作在这种模式下。

2) 下载模式 (downloading)

在这种模式下,开发人员可以使用各种命令,通过串口连接网络连接等通信手段从主机下载文件(比如内核映像),将它们直接放在内存运行或是烧入 Flash 类固态存储设备中。

从固态存储设备上启动的 Bootloader,启动过程大多是两阶段。第一阶段使用汇编来实现,它完成一些依赖于 CPU 体系结构的初始化,并调用第二阶段的代码;第二阶段则通常使用 C 语言来实现,这样可以实现更复杂的功能,而且代码会有更好的可读性和可移植性。一般而言,这两个阶段完成的功能可以如下分类:

1) Bootloader 第一阶段的功能:硬件初始化、为加载 Bootloader 第二阶段代码准备 RAM 空间、复制 Bootloader 第二阶段代码到 RAM 空间中、设置好栈、跳转到第二阶段代码的 C 入口点。

2) Bootloader 第二阶段的功能:初始化本阶段要使用的硬件设备、检测系统内存映射、将内存映象和根文件系统映象从 FLASH 上读到 RAM 空间中、为内核设置启动参数、调用内核。

### 3.4 本设计的U-Boot 1.1.6

U-Boot,全称为 Universal Boot Loader,即通用 Boot-

loader,是遵循 GPL 条款的开放代码项目。它的名字包含两层意思:第一,可以引导多种操作系统、支持多种架构的 CPU。它支持如下系统:Linux、NetBSD、VxWorks、QNX 等,支持如下架构的 CPU:PowerPC、MIPSx86、ARM、NIOS 等。

选择 U-Boot 的理由:

1) 开放源码;2) 支持多种嵌入式操作系统内核,如 Linux、NetBSD、VxWorks、QNX、RTEMS、ARTOS、Lynx-OS;3) 支持多个处理器系列,如 PowerPC、ARM、x86、MIPS、XScale;4) 较高的可靠性和稳定性;5) 高度灵活的功能设置,适合 U-Boot 调试、操作系统不同引导要求、产品发布等;6) 丰富的设备驱动源码,如串口、以太网、SDRAM、FLASH、LCD、NVRAM、EEPROM、RTC、键盘等;7) 较为丰富的开发调试文档与强大的网络技术支持。

本设计使用 U-boot1.1.6。

### 3.5 Linux 内核 2.6.13

Linux 的启动过程,主要分为两部分:架构/开发板相关的引导过程和后续的通用启动过程。引导阶段通常使用汇编语言编写,它首先检查是否支持当前架构的处理器,然后检查是否支持当前开发板。通过检查后,就为调用下一阶段的 start\_kernel 函数作准备了。这主要分如下两个步骤:1) 连接内核时使用的虚拟地址,所以要设置页表、使能 MMU。2) 调用 C 函数 start\_kernel 之前的常规工作,包括复制数据段、清除 BSS 段、调用 start\_kernel 函数。第二阶段的关键代码主要使用 C 语言编写。可以运行初始化的全部工作,最后调用 rest\_init 函数启动 init 的过程,创建系统第一个进程:init<sup>[2]</sup>。

Linux 内核编译命令:

1) make clean:该命令清除以前构造内核时生成的所有目标文件、模块和临时文件。

2) make dep:该命令搜索 Linux 输出与源代码之间的依赖关系,并以此生成依赖文件。

3) make menuconfig:该命令是用来调用菜单式配制内核界面。

4) make zImage:该命令用来编译内核,生成压缩的 Linux 内核目标代码 zImage 文件。

5) make modules:该命令编译模块驱动程序

本设计使用的内核是 linux2.6.13。烧录的是 zImage\_n35,因为这个适用于 NEC3.5 寸 LCD,也就是本设计所用的 LCD。

### 3.6 NFS 服务配置和防火墙设置

1) 建立工作目录

mkdir -p /opt/FriendlyARM/mini2440

编辑 nfs 服务的配置文件: /etc/exports,

增加共享目录 /opt/FriendlyARM/mini2440/root\_nfs \* (rw, sync, no\_root\_squash)。

系统通过 nfs 挂载; \* 表示所有的客户机都可以挂载此目录; rw 表示挂载此目录的客户机对该目录有读写的权力; no\_root\_squash 表示允许挂载此目录的客户机享有该主机的 root 身份。

2) 建立共享目录

拷贝 root\_nfs.tgz 文件到 /tmp 目录,进入此目录,执行

以下命令:

```
# tar xvfz root_nfs.tgz -C /opt/FriendlyARM/mini2440/root_nfs。这个命令把 root_nfs 的内容解压安装到 /opt/FriendlyARM/mini2440/root_nfs 目录。这样就可以进行 nfs 挂载了,执行命令: # mount -t nfs -o nolock 192.168.1.111:/opt/FriendlyARM/mini2440/root_nfs /mnt。假设电脑上虚拟机的 IP 地址已经设置为 192.168.1.111。
```

如果不通过 NFS 挂载,也可通过挂载 U 盘来使用外部资源。插上 U 盘后,执行挂载命令:

```
mount /dev/svsi/host0/bus0/target0/lun0/part1 /mnt。
```

### 3.7 登录 ARM-Linux 系统

经过上面的一系列工作,现在板子上的系统已经可以运行起来了。启动的时候,在 3s 内按下 C 键可以进入 U-boot 工作界面,可以在这里通过命令行操作。若在 3s 内没有按下 C 键则系统直接启动内核。图 1 是登录界面的截图。通过播放 .AVI、.mp3 等格式的多个音频和视频文件。播放效果图如图 2。

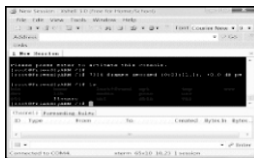


图 1 系统登录



图 2 PC 机上 MPlayer 成功播放

## 4 结语

Linux 功能强大,支持大量的微处理器体系结构、硬件设备、图形支持和通信协议,作为许多方案和产品的软件平台会越来越流行。相比于通用计算机,嵌入式系统具有突出的特点,如软硬件可裁剪、极其关注成本、对实时性和可靠性有严格的要求、开发需要专门工具和特殊方法、32 位芯片将能够执行上百万行 C 代码构成的复杂程序,具备高度复杂和智能化的功能。软件的实现从某种意义上说决定了产品的功能,已成为新产品成功与否的关键因素。从因特网设备到专用的控制系统, Linux 操作系统的前景都很光明,所有新造的微型计算机芯片中大约有 95% 都是用于嵌入式应用的。

本设计的创新理念是研究 MPlayer 播放器向更便携、多格式播放方面发展。Linux 系统具有很好的可移植性,它的实现过程以及核心的代码对类似应用具有很好的可重用性,只需通过较小的修改就能移植到不同的操作系统和平台上。而 ARM 是嵌入式设备优选的平台,所以本系统具有很强的前瞻性和实用价值。

### 参考文献

- [1] 代永陆,唐晓英.基于嵌入式系统的便携式多参数健康监护仪设计[J].电子技术应用,2006,33(9):55-57.  
DAI Yonglu, TANG Xiaoying. Portable multi-parameter health Monitor Design based on embedded systems[J]. Application of Electronic Technique, 2006, 33(9): 55-57, 2006.
- [2] 王运鑫,和卫基. ARM7 的新型嵌入式心电监护仪的研制[J].

(下转第 81 页)

题,使得 TC 能够对该系统中任意用户之间的通信信息进行解密,这样用户之间通信的保密性就无从谈起,TC 会成为该系统安全性的瓶颈。本方案在文献[4]的基础上,利用文献[10]中加密方案的一个变形,给出了一个新的一次性盲公钥方案,并证明了本方案的安全性;同时本方案借鉴无证书公钥密码方案中克服密钥托管问题的思想,在密钥生成阶段引入盲因子,使用户的私钥由 TC 和用户共同生成,对于 TC 来讲无法掌握用户完整的通信私钥。本方案在满足用户匿名通信的基础上,克服了密钥托管问题,其安全性要优于现有的方案,适用于用户对通信安全性要求更高的场合。

### 参考文献

- [1] 张秋璞,郭宝安.基于 ID 的一次性盲公钥[J].电子学报,2003,31(5):769-771.  
ZHANG Qiupu, GUO Baoan. One-Off Blind Public Key Based on ID[J]. AGTA ELECTRONICA SINICA, 2003, 31(5): 769-771.
- [2] Shamir A. Identity-based cryptosystems and signature schemes [A]. Advance in Cryptology-Crypto 84[C]// Germany: Springer-Verlag, 1984, LNCS, 196: 47-53.
- [3] Boneh D, Franklin M. Short Signatures from Weil Pairing[A]. Boyd C ASIACRYPT 2001[C]// Berlin: Springer-Verlag, 2001: 514-532.
- [4] 张胜,徐爱国,胡正名,等.一种基于身份一次性公钥的构造[J].电子与信息学报,2006,28(8):1412-1414.  
ZHANG Sheng, XU Guoai, HU Zhengming, et al. Construction of the One-Off Public Key Based on Identity[J]. Journal of Electronics & Information Technology, 2006, 28(8): 1412-1414.
- [5] 李毅,张少武,张远洋,等.基于身份一次性公钥的分析与改进[J].计算机工程与设计,2008,29(7):1636-1640.  
LI Yi, ZHANG Shaowu, ZHANG Yuanyang, et al. Analysis and improvement on identity-based one-off public key[J]. Computer Engineering and Design, 2008, 29(7): 1636-1640.
- [6] 鲁荣波,何大可,王常吉,等.改进的基于身份一次性公钥[J].计算机应用研究,2008,25(4):1139-1141.  
LU Rongbo, HE Dake, WANG Changji, et al. Improvement on one-off public key based on identity[J]. Application Research of Computers, 2008, 25(4): 1139-1141.
- [7] 甄鸿鹄,陈越,李乐,等.基于身份的一次性公钥分析与重构[J].计算机工程,2010,36(1):187-196.  
ZHEN Honghu, CHEN Yue, LI Le, et al. Analysis and Reproduction of One-off Public Key Based on Identity[J]. Computer Engineering, 2010, 36(1): 187-196.
- [8] Menezes A, Okamoto T, Vanstone S. Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field[J]. IEEE Trans on Information Theory, 1993, 39(5): 1639-1646.
- [9] S. S. Al-Riyami, K. G. Paterson. Certificateless Public Key Cryptography[J]. Advances in Cryptology ASIACRYPT, 2003 (11): 452-473.
- [10] 文卉,胡剑波. RSA 公钥密码的威胁-Shor 量子算法[J].舰船电子工程,2008(7).  
WEN Hui, HU Jianbo. Menace to RSA Public-key Cryptography-Shor Algorithm[J]. Ship Electronic Engineering, 2008(7).
- [11] 拾以娟.基于身份的公钥密码学关键问题研究[D].上海:上海交通大学,2007.  
SHI Yijuan. Research On Key Problems in Identity Based Public Key Cryptography[D]. Shanghai: Shanghai Jiao Tong University, 2007.

(上接第 46 页)

- 微计算机信息,2006(12):97-99.  
WANG Yunxin, HE Weiji. The Design Of A New Kind of Portable ECG Monitor Based On ARM7[J]. Control & Automation, 2006(12): 97-99.
- [3] 刘传菊,肖明明.网络环境下数字化生产模式的研究[J].微计算机信息,2006(12):160-162.  
LIU Chuanju, XIAO Mingming. Research and Implementation of Digital Production Pattern[J]. Microcomputer Information, 2006(12): 160-162.
- [4] 刘传菊,肖明明.基于网络一无线技术的分布式测控系统的优化设计[J].微计算机信息,2005(12):153-155.  
LIU Chuanju, XIAO Mingming. Scatternet-Wireless Measure and Control System Based on Bluetooth Technology[J]. Microelectronics & Computer, 2005(12): 153-155.
- [5] 游国福,吴宏. JX 处理器内嵌 PLL 中 VCO 的设计[J].微电子学与计算机,2007(2):184-187.  
YOU Guofu, WU Hong. The Design of the VCO of PLL Embedded in JX Processor[J]. Microelectronics & Computer, 2007(2): 184-187.
- [6] 吴志伟,邹雪城,吴鑑铭.嵌入式存储器的内建自修复设计[J].微电子学与计算机,2007(2):79-81.  
WU Zhiwei, ZOU Xuecheng, WU Jianming. A Built-In Self-Repair Design for Embedded Memory[J]. Microelectronics & Computer, 2007(2): 79-81.
- [7] 熊开盛,董兆鑫.基于 DSP 技术的 MP3 播放器的硬件设计[J].现代电子技术,2007(6):164-166.  
XIONG Kaisheng, DONG Zhaoxin. Hardware Design of MP3 Player Based on DSP Technology[J]. Modern Electronics Technique, 2007(6): 164-166.
- [8] 崔世海,肖惠惠,李山.基于 ARM 的汽车综合信息记录仪的研究[J].现代电子技术,2007(6):17-19.  
CUI Shihai, XIAO Huihui, LI Shan. Study on Synthetical Information Recorder of Automobile Based on ARM[J]. Modern Electronics Technique, 2007(6): 17-19.
- [9] 唐立强,王行祥.嵌入式 VxWorks 系统下的 CPCI 设备驱动程序开发[J].计算机与数字工程,2011(1):183-185.  
TANG Liqiang, WANG Xingxiang. Development of CPCI Device Driver for Embedded VxWorks System[J]. Computer & Digital Engineering, 2011(1): 183-185.
- [10] 吕俊怀,巩固.嵌入式 GPRS 系统的设计与实现[J].计算机与数字工程,2011(3):72-75.  
LV Junhuai, GONG Gu. Design and Implement of Embedded GPRS System[J]. Computer & Digital Engineering, 2011(3): 72-75.