

文章编号: 2095—0411 (2012) 01—0065—04

ARM 无线视频传输系统的组建及初始化程序设计^{*}

辛元雪, 江 冰

(河海大学 计算机与信息学院 (常州), 江苏 常州 213022)

摘要: 通过分析无线视频传输系统的特点及原理, 提出一种利用 ARM9 构建移动视频终端, 搭建嵌入式 Linux 系统充分调配硬件资源的新方案, 从而以无线的方式实现了移动终端与远程 PC 机之间的视频采集、控制和传输等策略, 完成嵌入式 Linux 系统的移植与初始化程序设计。移动终端采用 TE2440, 利用 CMOS 摄像头采集图像。整机系统完成了测试, 达到预期功能要求。

关键词: 嵌入式系统; CMOS 摄像头; 无线传输

中图分类号: TH 873.7

文献标识码: A

Establishment and Design of Initial Program for Wireless Video Transmission System Based on ARM

XIN Yuan—xue, JIANG Bing

(College of Computer and Information Engineering, Hohai University, Changzhou 213022, China)

Abstract: Through an analysis of character and principle of wireless video transmission system, a new program was made, using an ARM9 as mobile video terminal, and setting up embedded Linux system to achieve the full use of hardware resources, thereby achieving the video capture, control and transmission strategies between mobile terminal and the remote PC with wireless technology, completing the embedded Linux system transplantation and the initial program design. TE2440 was used as mobile terminal, CMOS camera was used as image acquisition. Experiments show that the system has met the expected functional requirements.

Key words: embedded system; CMOS camera; wireless transmission

目前移动视频传输系统广泛应用于军事、政府、交通安防、酒店等领域。传统的视频传输系统应用有限的计算机资源, 传输有线化, 使得其系统庞大, 维护繁琐, 传输受限。如何使视频传输系统精简, 成本降低, 体积缩小, 传输无线化, 是开发人员应主要考虑的问题。

本文提出了一种基于 ARM 的无线视频传输系统, 为视频传输系统提供了一种新方案, 采用嵌入

式 Linux 操作系统, ov9650 摄像头, 基于 WLAN 的无线网卡, 在监控防范、安全警报、远程教育等方面有广阔的应用前景。

1 系统体系结构

1.1 系统硬件结构

系统的硬件主要包括嵌入式移动终端和 PC

^{*} 收稿日期: 2011—07—01

基金项目: 2008 年常州市创新基金计划项目 (CN2008033)

作者简介: 辛元雪 (1987—), 女, 河南郑州人, 硕士生。

机, 嵌入式微处理器采用三星公司的 S3C2440 芯片。该芯片采用了 ARM920T 的内核, 0.13 μ m 的 CMOS 标准宏单元和存储器单元, 采用了新的总线架构: Advanced Micro controller Bus Architecture (AMBA)。

终端和 PC 机之间在无线 WIFI 模块上建立 NFS 系统, S3C2440 处理器控制 CMOS 摄像头采集图像信号, 经过压缩编码实现图像的无线传输。系统硬件结构框图如图 1 所示。在图 1 中主芯片 S3C2440 为本系统的核心, 控制、处理图像的采集, 并配有丰富的接口; CAMERA 为连接终端和 CMOS 摄像头的专用 20 脚接口; SDRAM 和 FLASH 为本系统的存储单元, FLASH 用于保存代码和各种文件, SDRAM 用于代码的执行; USB 接口连接无线网卡和终端, 无线网卡与 PC 机之间通过建立无线网络实现图像的传输。

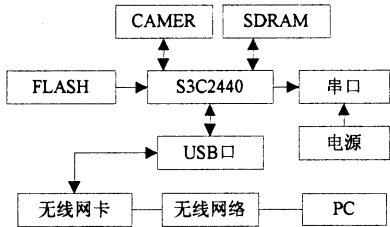


图 1 系统硬件结构框图
Fig. 1 Frame structure of hardware system

1.2 系统软件结构

终端资源丰富, 需要一个操作系统进行统一管理和分配。因此采用嵌入式 arm-Linux-2.6.12 操作系统作为底层软件。移动视频采集系统的软件结构框图如图 2 所示。图 2 中的 Bootloader, Linux 内核, 文件系统 3 者的建立就构成了嵌入式 Linux 操作系统。本课题中的任务是实现 CMOS 摄像头的视频传输, 因此必须开发 CMOS 摄像头的驱动程序, 驱动程序的开发是建立在操作系统的成功移植上。

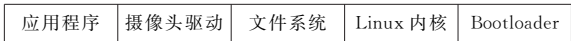


图 2 系统软件结构框图
Fig. 2 Frame structure of software system

2 系统的扩展接口设计

2.1 USB 驱动无线网卡

USB 接口用于连接终端和无线网卡, S3C2440 的 USB 接口在电路设计时 D+ 和 D- 两根信号线

需要加 15k Ω 的下拉电阻。

2.2 CMOS 摄像头接口

本设计采用的是 OmniVision 公司的 CMOS 彩色图像传感器 OV9650, 它采用专门定制的串行摄像头控制总线 SCCB (Serial Camera Control Bus), 通过对图像传感器的读写, 达到对输出图像的控制, SCCB 是一种双向两线制同步串行总线。两线制 SCCB 只包括串行时钟线 SIO_C 和串行数据线 SIO_D^[1]。OV9650 的视频输出格式为 8 位 YUV 时用到 8 根数据线 D2-D9 (D9 为 MSB 位, D2 为 LSB 位); 当数据格式为 10 位 RGB 时用到 10 根数据线 D0-D9 (D9 为 MSB 位, D0 为 LSB 位)^[2], 该系统用 YUV 格式。OV9650 与 S3C2440 控制接口电路如图 3 所示。

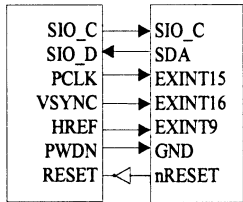


图 3 OV9650 与 S3C2440 的控制接口
Fig. 3 Control interface of OV9650 and S3C2440

图中 PCLK 为像素时钟, HREF 为水平参考信号, VSYNC 为垂直同步信号。OV9650 采用 I²C 总线功能集中的一个子集, S3C2440 可以通过 I²C 总线对 OV9650 内部的相关寄存器进行设置, 因此 SIO_C 和 SIO_D 分别连接 S3C2440 的 SCL, SDA 即可, 使得摄像头既能正常工作, 输出满足一定大小、一定格式和帧率等要求的图像, 同时又能保证输出图像具有一定质量, 如: 选择合适的自动增益控制参数, 充分利用摄像头提供的自动控制和校正功能如饱和度, 色度, 噪点控制等^[3]。

对 OV9650 寄存器的设置: 将 0x04 寄存器设置为 0x24, 实现隔行采集图像; 将寄存器 0x3A 设置为 0x08, 实现输出格式为 YUV 4:2:2; 将 0x11 寄存器设置为 0x81, 实现输出图像帧频为 60Hz。

3 移动视频系统软件平台的设计

终端选择用嵌入式 Linux 操作系统, Linux 系统内核可定制, 并且完全开源, 能够完成嵌入式功能的任务和用户界面。嵌入式 Linux 系统的搭建包括 4 个部分: 交叉编译环境的建立, Bootloader 的移植, 文件系统的移植, 内核的移植。

3.1 交叉编译环境的搭建

交叉编译, 简单地说, 就是在一个平台上生成另一个平台上的可执行代码, 需要交叉编译的原因有两个: 首先, 在项目的起始阶段, 目的平台尚未建立, 因此需要做交叉编译, 以生成所需要的 bootloader (启动引导代码) 以及操作系统核心; 其次, 当目的平台能启动之后, 由于目的平台上资源的限制, 当编译大型程序时, 依然可能需要用到交叉编译。将 cross-3.4.1.tar.bz 解压到 /usr/local 下, 编辑 /etc/bashrc 文件, 增加路径 export PATH=/usr/local/arm/3.4.1/bin: \$PATH。

可用 echo \$PATH 来查看环境变量中是否有 /usr/local/arm/3.4.1/bin 路径变量。若显示路径正确, 则说明交叉编译环境成功建立。

3.2 文件系统的移植

文件系统存放了嵌入式系统使用的所有应用程序、库文件以及常用服务, 系统启动内核后需要加载文件系统。文件系统的制作比较复杂, 先解压 qte_yaffs.tar.bz2 这个压缩包。解压后得到 qte_yaffs 目录, 文件系统的所有文件都在该目录下, 可根据需要修改。用 mkyaffs2image 工具来制作文件系统。qte_mouse.yaffs 和 qte_touch.yaffs, 分别是支持鼠标和支持触摸屏的文件系。先擦除 flash 中文件系统对应的分区, 然后使用 USB 口烧写文件系统, 在 bootloader 选择界面选择烧写文件系统, 然后进行烧写即可。

3.3 内核的移植

Linux 系统除了开源外最大的优势就是其可裁剪性, 依据用户需要对外核定制, 既精简系统也做到了充分利用资源。内核在硬件之上定义了一个高阶的抽象界面, 应用一组原语 (或者叫系统调用) 来实现操作系统的功能, 例如进程管理, 文件系统, 和存储管理等等, 这些功能由多个运行在核心态的模块来完成。

移植内核前需要修改 makefile, 进入解压的目录后, 运行命令 “emacs Makefile”, 修改 “CROSS_COMPILE=/usr/local/arm/3.4.1/bin/arm-Linux” 用于指明交叉编译器的版本和位置。修改 “ARCH:=” 为 “ARCH:=arm” 指定处理器。

在 Linux 的终端窗口中输入内核配置命令:

make menuconfig 进入配置栏的 “LoadanAlternateConfigurationFile” 输入配置文件名 kernel_2440.cfg。

3.4 摄像头驱动程序的优化与加载

驱动程序的代码庞大, 分为 3 部分: 自动配置和初始化, I/O 端口请求, 中断服务。在修改代码的时候遵循这 3 部分按需修改。驱动程序的开发流程见图 4 所示。

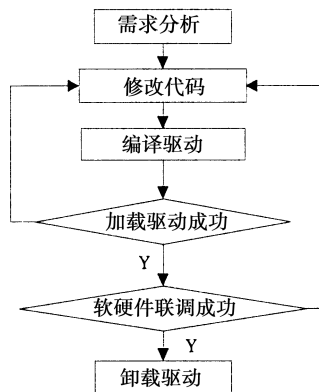


图 4 驱动程序的开发流程

Fig. 4 Development flow of the driver

本系统驱动采用动态加载的方法, 驱动程序将会以模块形式存储在文件系统里, 需要时动态载入到内核即可, 实现按需加载, 完成了模块化驱动程序的测试后, 可以通过 insmod 和 rmmod 命令方便地加载和卸载^[4]。具体实现方法见图 5 所示。

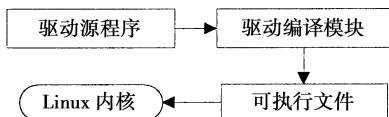


图 5 驱动程序的动态加载

Fig. 5 Dynamic loading of the driver

4 无线传输网络的实现

整个视频传输系统由 CMOS 摄像头采集图像, 通过无线模块以 NFS 文件共享的方式在 PC 机的 Linux 操作系统下查看图片。本设计采用的是雷凌公司的 RT73 无线网卡, USB 接口可以方便的与开发板相连, 该无线网卡的驱动实现以下接口函数:

- 设备初始化函数 usb_rtsub_exit ();
- 设备销毁函数 usb_rtsub_open ();
- 打开设备函数 usb_rtsub_close ();
- 命令控制调用函数 RTMPSendPackets ();
- 数据包发送函数 RTUSBBulkReceive ()。

设计完无线网卡的驱动后，要把它编译成为可执行的驱动文件。

4.1 配置无线网络

将主机 ip、PC 机 Linux 系统 ip、终端的无线网卡 ip 3 者配置在同一网段。具体过程如下：在 PC 机的网上邻居中设置 PC 机 ip 为 192.168.242.5。打开超级终端，关闭 Linux 虚拟机上的串口，此时一定要拔掉网线，在命令窗口中配置 ip 为 192.168.242.3。用超级终端配置开发板的无线网络。首先加载无线网卡的驱动并给无线网卡分配 ip：insmod rt73.ko
ifconfig rausb0 192.168.242.6

查看可用的无线网络并修改与之对应的无线网卡参数：
cd/usr
. /iwlist rausb0 scanning
. /iwconfig rausb0 mode # # # #
. /iwconfig rausb0 channel # # # #
. /iwconfig rausb0 essid # # # #
最后用 ping 命令检查终端与主机，PC 机 Linux 系统 3 者之间的连通情况。

4.2 建立 NFS 系统挂载文件实现图像传输

NFS（网络文件系统）是一种在网络上的机器间共享文件的方法。要建立 NFS 系统，主要包括启动 NFS 端口和设置共享文件路径两方面：
gedit/etc/default/portmap
/etc/init.d/portmap restart
gedit/etc/exports
在打开的文件中写入：
/home/xin/arm/rootfs * (rw, sync, no_root_squash)
该命令的实现的功是：在建立好 NFS 系统后，使终端共享 PC 机的 Linux 系统/home/xin/arm/rootfs 目录中的文件，“*”代表所有的 ip 地址都可以访问，rw 代表终端可以对共享的文件进行读写操作，sync 表示二者同步，no_root_squash

表示有 root 权限。然后重启 NFS 系统：/etc/init.d/nfs restart。最后挂载文件：mount -t nfs 192.168.242.3: /home/xin/arm/rootfs/tmp 表示将 ip 地址为 192.168.242.3 的 PC 机下的/home/xin/arm/rootfs 目录中的文件挂载到开发板的 tmp 目录下。

4.3 图片采集和无线传输测试

首先加载摄像头驱动并且初始化摄像头：insmod S3C2440a_camif.ko. /ov9650_init 采集图片，并在 PC 机的 Linux 系统的/home/xin/arm/rootfs 目录下查看图片。
用 CMOS 摄像头采集图像，利用无线 WIFI 模块传输图片到 PC 机。调试环境见图 6 所示。

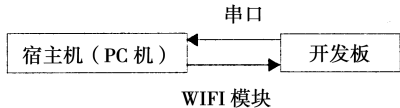


图 6 系统的调试环境
Fig. 6 Debug enviroment of the system

5 结 论

本系统针对基于 ARM 的图像采集和传输提出了一种采用小巧的 CMOS 摄像头和方便的 USB 无线网卡方案，对其软硬件实现进行了描述，具有安全可靠，操作简便，成本低的优点，图像传输的无线化使得嵌入式系统在网络技术高速发展的 21 世纪有着广阔的发展空间和应用前景。

参考文献：

[1] 阮越广，赵伟胜．基于 WLAN 的家用安全监控系统的硬件设计 [J]．计算机工程，2008（5）：250—252.
[2] 柳迪，黎福海，闫旭．基于 Blackfin DSP 的图像数据采集设计 [J]．计算机系统应用，2008（1）：117—119.
[3] 许雪梅，郭远威，吴爱军，等．基于嵌入式 Windows CE5.0 的无线监控系统研究 [J]．现代电子技术，2009（2）：25—26.
[4] 孙琼．嵌入式 Linux 设备驱动开发详解 [M]．北京：人民邮电出版社，2008：19—20.