

基于嵌入式 Internet 的远程视频监控系统设计

焦 铭^{1,2}, 易小波², 李仁发¹

- (1. 湖南大学 计算机与通信学院, 湖南 长沙 410082;
2. 衡阳师范学院 计算机系, 湖南 衡阳 421008)

摘 要:文中提出了一种基于嵌入式 Internet 的远程视频监控系统设计方案,视频监控终端采用 S3C2410 微处理器,视频监控设备采集的视频图像经过 MPEG-4 压缩算法编码压缩后送入该处理器,利用 RTP 实时传输协议以及无连接数据报协议 UDP/IP 等将该数据流打包后,利用 Internet 网络进行实时传输到用户端。基于 Internet 的远程视频监控系统是信息网络与控制网络结合的产物,它借助网络完成监视与控制任务,将监控范围扩展到更广的空间,进一步推进了控制技术向网络化、分散化及开放化的发展。

关键词:嵌入式 Internet;远程视频监控系统;MPEG-4;RTP

中图分类号:TP273+.5;TP393.9

文献标识码:A

文章编号:1673-629X(2009)05-0176-04

Design of Remote Video Supervisory System Based on Embedded Internet

JIAO Ge^{1,2}, YI Xiao-bo², LI Ren-fa¹

- (1. Computer and Communication College, Hunan University, Changsha 410082, China;
2. Computer Department, Hengyang Normal University, Hengyang 421008, China)

Abstract: Presents a remote video supervisory system based on embedded Internet, the video surveillance terminal uses the microprocessor S3C2410, the video captured by the video device is coded and compressed by the MPEG-4 algorithm and is sent to the microprocessor. After processed by it the MPEG-4 data stream is sent to video surveillance network. Then the MPEG-4 data stream is packed by the real-time transport protocol and UDP/IP. Finally the data packet is transmitted to the user by Internet. Internet-based remote video surveillance system is an information network and control network products, using its network to monitor and control tasks, monitoring will be extended to a broader space to further promote the technology to control the network, decentralized and open.

Key words: embedded Internet; remote video supervisory system; MPEG-4; RTP

0 引言

远程监控系统是当前工业自动化应用领域研究的热点之一。远程监控系统主要用于完成远程现场点的数据采集、处理、实时监控等功能。传统的远程监控系统没有或无法解决实时大数据量处理、远程通信等问题,其应用受到很大的限制。基于 Internet 的远程视频监控系统是信息网络与控制网络结合的产物,它借助网络完成监视与控制任务,将监控范围扩展到更广的空间,进一步推进了控制技术向网络化、分散化及开放化的发展^[1]。

1 系统硬件设计

图 1 所示的是基于嵌入式 Internet 的远程视频监控系统的硬件架构。整个系统主要由 CPU 芯片、FLASH 芯片、视频采集设备、SDRAM 内存、以太网接口等组成。CPU 采用的是韩国三星公司的 S3C2410,该处理器的内部集成了 ARM 公司 ARM920T 处理器核的 32 位微控制器,并带有独立的 16kB 的指令 Cache 和 16kB 的数据 Cache、LCD 控制器、RAM 控制器、NAND 闪存控制器、3 路 UART(通用异步收发器)、4 路 DMA、4 路定时器、并行 I/O 口、8 路 10 位 ADC、触摸屏接口、IIC 接口、IIS 接口、2 个 USB 接口控制器、2 路 SPI,主频最高可达 203MHz^[2]。USB 视频采集设备和 S3C2410 微处理器构成整个系统的视频监控终端设备,放在监控现场,而 Internet 网络与用户端监控设备一起构成监控网络平台,通过网络用户

收稿日期:2008-08-20

基金项目:国家 863 计划资助项目(2007AA01Z104);国家自然科学基金资助项目(60673061)

作者简介:焦 铭(1979-),男,硕士研究生,讲师,研究方向为嵌入式系统;李仁发,教授,博士生导师,研究方向为嵌入式系统。

可随时了解监控现场的情况。

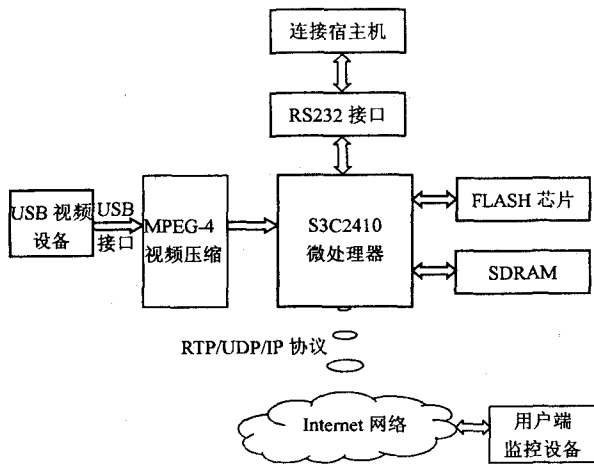


图 1 系统硬件架构图

2 系统软件设计

2.1 系统软件结构

视频监控终端 WEB 服务器的软件结构如图 2 所示,其主要由视频监控 WEB 服务器、CGI(公用网关接口)、嵌入式数据库、存储管理与调度模块、视频调度与传输模块、视频设备控制模块等组成。

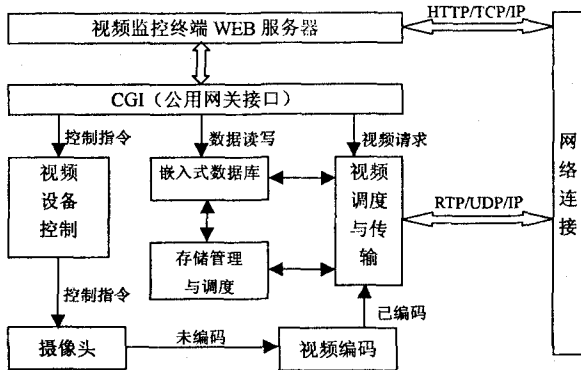


图 2 系统软件结构图

系统通过摄像头采集图像,对现场图像进行高速捕捉,然后将捕捉到的图像通过 UBS 总线传输到 S3C2410 处理器进行处理,并将图像压缩,保存为 JPEG 格式的文件。程序调用 EnCoder 编码器可以将多幅 JPEG 格式的图像合成一段 AVI 视频流,实现视频回放。最后,系统还可以通过以太网或 UART 将保存的图片和视频流传到服务器上,实现网络监控^[3]。视频监控终端 WEB 服务器通过 HTTP 协议与用户端监控设备浏览器软件进行信息交互,提供其他应用程序模块的接口以及视频数据浏览界面和摄像头控制界面。此外,它还要对用户端的访问权限进行控制,过滤用户端的请求和控制信息,处理多个用户端的请求和控制的同步和优先级问题。而十分适用于嵌入式系

统。同时,本系统采用了 CGI 来实现动态 WEB 技术,CGI 规定了 WEB 服务器调用其他可执行程序的接口协议标准。WEB 服务器通过调用 CGI 程序实现和浏览器的交互,也就是 CGI 程序接受浏览器发送给 WEB 服务器的信息,进行处理,将响应结果再回送给 WEB 服务器及浏览器。

2.2 Linux 内核配置

2.2.1 修改 Linux 配置文件

接下来需要对一些配置文件作一定的修改。

(1)打开根目录下的 Makefile 文件。

指定目标平台:添加 Arch:= arm

指定交叉编译器:添加 CROSS_COMPILE = arm-elf-gcc

(2)打开 arch/arm 目录下的 Makefile 文件。

包含 S3C2410 处理器:ifeq(\$(CONFIG_ARCH-S3C2410),y)

MACHINE=S3C2410

(3)添加内核起始运行地址,即 image.ram 应下载的位置,TEXTADDR=0xC0008000

(4)打开 arch/arm/boot 目录下的 Makefile 文件。

指定 Bootloader 的压缩内核解压后数据的输出地址。

ifeq(\$(CONFIG_ARCH-S3C2410),y)

ZRELADDR=0X50008000

2.2.2 编译 Linux 内核

在完成上述工作后,开始编译 Linux 内核,生成目标代码。首先以 ROOT 身份进入系统,然后完成下列步骤,就可以在 /Linux/arch/arm/boot/得到内核的映像文件 zImage。在 shell 命令下输入命令进行内核配置:make menuconfig 该命令执行之后生成文件 .cnofig。它保存这个配置信息,下一次再执行 make mneucnofig 的时候将生成新的 .cnofig 文件。

输入命令:make dep

输入命令:make clean

输入命令:make zImage

通过各个目录的 Makefile 进行,将会在各个目录下生成一系列目标文件,从而完成对 Linux 的编译工作。Linux 内核的编译、修改、移植与上层的应用程序(如命令解释器 shell、登陆程序 login)相关,将这些程序根据需要重新定制。利用终端仿真程序 minicom 和 Bootioader 引导程序通过串口就可以把 Linux 内核移植到目标系统板。移植完成后,Linux 就可以在目标系统板上运行。

2.3 视频采集摄像头驱动设计

视频信号的采集一般选择 USB 摄像头来实现,使

用 USB 摄像头虽然成本偏高,但是易于实现,节省 CPU 资源,而且驱动支持非常丰富。本系统使用的是 ZC301 摄像头,它可以用来采集一张完整的 jpeg 图片。

要使用 USB 设备需要正确地配置 Linux 内核,要启用 Linux USB 支持,首先进入“USB support”并启用“Support for USB”选项。在 Linux kernel 源码目录中 driver/usb/usb-skeleton.c 为我们提供了一个基础的 USB 驱动程序。通过它仅需要修改极少的部分,就可以完成一个 USB 设备的驱动。其驱动程序的编写主要包括下面的内容:提供基本的 I/O 操作接口函数 open、read、write、close 的实现、对中断的处理实现、内存映射功能以及对 I/O 通道的控制接口函数 ioctl 的实现等,并把它定义在 struct file_operations 中^[1]。Linux USB 驱动程序需要做的第一件事情就是在 Linux USB 子系统里注册,并提供相关信息。将驱动和设备绑定上了,任何用户态程序(如 open、close、read、write 等系统调用操作时)要操作此设备都可以通过 file_operations 结构所定义的函数进行了。

2.4 MPEG-4 视频压缩的实现

MPEG-4 具有高速压缩、基于内容交互和基于内容分级扩展等特点,并且具有基于内容方式表示的视频数据。它把任一个视频序列看成一个或多个视频对象(VO)的集合。VO 是场景中的某个物体,由时间上连续的许多帧构成,它是一个承上启下的概念,一个或多个 VO 组成一个视频场景(VS),每个 VO 可能有一个或多个 VOL 层次,VOL 引入了主要用来扩展 VO 的时域或空域分辨率,它把 VO 的一些属性信息进行编码。VOP 是某一时刻某一帧画面的 VO,VOP 编码就是对某一时刻该帧画面 VO 的形状、运动和纹理等三类信息进行编码。

基于内容的视频编码过程由以下三步完成^[4]:

(1) VO 的形成:先从原始视频流中分割出 VO。

(2) 编码:对各 VO 分别独立编码,即对不同 VO 的运动信息、形状信息和纹理信息这三类信息分别编码,分配不同的码字。

(3) 复合:将各个 VO 的码流复合成一个符合 MPEG-4 标准的位流。

XviD 是一个开放源码的、高效的、具有可移植性的 MPEG-4 编码软件,本系统选择 xvidcore 作为视频图像压缩模块中的核心算法。对 xvidcore-1.0.3 进行交叉编译比较简单,有以下步骤^[5]:

(1) 解压缩 xvidcore 源代码:tar -zxvf xvidcore-1.0.3.tar;

(2) 设置环境变量:export xvidcore="the path of xvidcore";

(3) cd \$ xvidcore/build/generic;

(4) 生成 makefile:./configure - host = localhost build= arm - linux - gcc;

(5) 编译源代码:make;make install;

(6) 将交叉编译生成的库文件 libxvidcore.so.* 拷贝到交叉编译器工作目录的 lib 子目录中,该库文件为系统的其它模块提供了编程接口。

2.5 MPEG-4 数据流的 RTP 封装

RTP 是由 IETF 组织制定的针对多媒体应用(如视频、音频等)的新型实时传输协议,它提供的服务包括负载类型标识、序列编号、时间戳和传输控制等^[6]。RTP 数据协议负责对流媒体数据进行封包并实现媒体流的实时传输。

由于视频数据一般都比较,所以不考虑几个访问单元串联封装的情况,采用以视频对象平面(VOP)为基本封装单元^[7],使用 SDP(会话描述协议)来指定视频流的属性(例如媒体类型、封装格式、编码结构等),直接将视频数据分片并映射到 RTP 包中的封装方案。其封装算法描述如下:

While (MPEG-4 数据流结束前)

```

{
    if(发现下一个 VOP 起始码)
    {
        if(当前分段长度 <= 去除头部字段长度的路径 MTU 值)
        {把此段数据打入 RTP 包}
        else
        {把尽可能多的宏块打入 RTP 包}
    }
    else
    {对剩余数据打包}
}

```

2.6 MPEG-4 数据流的 RTP 传输

MPEG-4 数据流分别被封装上 RTP 报头、UDP 报头和 IP 报头,然后 IP 数据包通过 Internet 向接收端发送。其传输模型^[8]如图 3 所示,当发送端收到压缩编码的 MPEG-4 流后,按照 RTP 数据传输协议的报文格式装入 RTP 报文的数据负载段,并配置 RTP 报文头部的时间戳、同步信息、序列号等参数,此时数据报文已经被流化。同时发送端周期性地接收 RTCP (RTP 控制协议)包,将 QoS 反馈控制信息发送到视频服务器,服务器利用这些信息动态地改变自身参数设置。接收端收到 IP 包后先分析 RTP 包头,判断版本、长度、负载类型等信息的有效性,更新缓冲区的 RTP 信息,如收到的字节数、视频帧数、包数、序列号数等信息。按照 RTP 时间戳和包序列号等进行信源同步,整理 RTP 包顺序,重构视频帧,最后根据负载类型标识

进行解码,将数据放入缓存供解码器解码输出,同时接收端根据 RTP 包中的信息周期性回送包含 QoS 反馈控制信息的 RTCP 包到数据发送端以检测发送端和接收端数据的一致性^[9]。

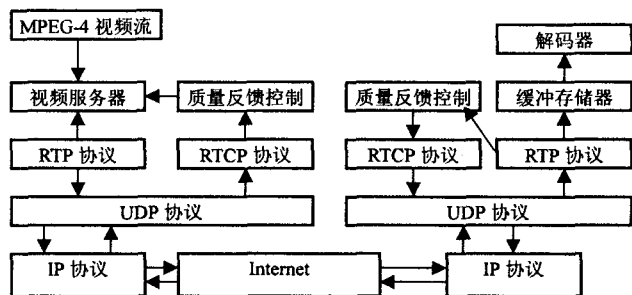


图 3 MPEG-4 数据流的 RTP 传输模型

3 结束语

基于嵌入式 Internet 的远程视频监控系统有着广阔的应用空间和美好的应用前景,能够在信息家电、电网运行监控、远程技术支持和远程故障诊断等多个领域得到应用^[10]。

参考文献:

[1] Hui S C. Remote Video Monitoring Over the WWW[J].

Multimedia Tools and Applications,2003,21(2):173-195.

[2] 王冬华,吴壮志. 边防视频监控系统的设计与实现[J]. 计算机技术与发展,2008,18(5):208-211.

[3] 操龙敏,蒋建国,齐美彬. RTP 协议在嵌入式网络摄像机中的设计及实现[J]. 计算机技术与发展,2008,18(3):214-216.

[4] 韩云,陈祖爵,郑尚志. MPEG-4 编码技术应用及 FPGA 实现[J]. 计算机技术与发展,2007,17(10):219-222.

[5] 杨晖,胡永健,林志泉. 基于 Linux 和 S3C2410 嵌入式图象传输系统设计[J]. 微计算机信息,2007,23(8):20-21.

[6] 张建. 基于 S3C2410 和嵌入式 Internet 的家庭视频监控系统设计[D]. 上海:上海交通大学,2007.

[7] 吴百锋,彭澄康,孙晓光. 一种基于监测的嵌入式系统设计技术[J]. 计算机学报,2003,26(12):1728-1733.

[8] LIU Quan, QU Xuehong. Research on Remote Video Monitoring System Used for Numerical Control Machine Tools Based on Embedded Technology[J]. 武汉理工大学学报,2006,28(2):617-620.

[9] Cha Kyung - ae. MPEG-4 STUDIO: An Object - Based Authoring System for MPEG-4 Contents[J]. Multimedia Tools and Applications,2005,25(1):111-131.

[10] VIAL P J. Using Embedded Internet Devices in an Internet Engineering Laboratory Set - up[J]. The International Journal of Engineering Education,2003,19(3):441-444.

(上接第 172 页)

技术与发展,2006,16(6):107-109.

[2] 黄建忠,谢长生. 网络存储安全研究趋热[J]. 中国教育网络,2006(8):41-42.

[3] 胡天翔. 网络存储技术在企业中的发展及应用[J]. 计算机技术与发展,2006,16(7):218-220.

[4] 李文红. 网络存储安全技术研究[J]. 武汉理工大学学报:信息与管理工程版,2006(8):54-56.

[5] 蔡涛,鞠时光,赵俊杰,等. 存储网层次安全模型的研究[J]. 计算机应用,2007(6):1534-1537.

[6] 伍小龙,温雅敏. 网络存储的安全问题与对策[J]. 江西理

工大学学报,2006(3):31-33.

[7] 韩德志,傅湘林,黄建忠. 基于 iSCSI 的附网存储安全系统的研究与实现[J]. 小型微型计算机系统,2004,25(7):1223-1227.

[8] 金红,王煜. 网络存储技术在网络数据备份系统中的应用[J]. 高性能计算技术,2003(6):50-53.

[9] 赵俊杰,詹永照,蔡涛. 网络存储安全系统研究综述[J]. 计算机应用与软件,2008(2):271-274.

[10] 姬耀,刘海涛. 文件数据的安全存储[J]. 信息安全与通信保密,2008(1):68-70.

(上接第 175 页)

2003,20(11):32-36.

[2] Wiederhold G. Mediators in the architecture of future information systems[J]. IEEE Computer,1992,25(3):38-49.

[3] Papakonstantinou Y, Garcia - Molina H, Widom J. Object exchange across heterogeneous information sources[C]//ICDE Conf. Taipei, Taiwan: IEEE Computer Society, 1995:251-260.

[4] Adali S, Emery R. A uniform framework for integrating knowledge in heterogeneous knowledge systems[C]//ICDE. Taipei, Taiwan: IEEE Computer Society, 1995:513-520.

[5] Tomasic A, Raschid L, Valduriez P. Scaling heterogeneous

databases and the design of disco[C]//International Conference on Distributed Computing Systems. Hong Kong: IEEE Computer Society, 1996:449-457.

[6] 李振,曹谢东,刘世齐. 基于 CORBA 的油气田异构信息系统多源集成[J]. 计算机技术与发展,2006,16(6):60-62.

[7] 李亚红,吴江. 基于 Web Services 实现异构数据库集成技术研究[J]. 计算机应用研究,2006(2):81-84.

[8] Lenzerini M. Data Integration: A Theoretical Perspective[C]//PODS. [s. l.]: ACM Press, 1997:233-246.