

文章编号:1007-2780(2013)06-0911-06

基于 DM6467T 的视频压缩存储和网络传输系统设计

张 晶^{1,2,3}, 戴 明², 孙丽娜², 赵春蕾^{1,2,3}, 冯 磊^{1,2,3}, 杨添星^{2,3}

(1. 中国科学院航空光学成像与测量重点实验室, 吉林 长春 130033, E-mail: 200731020027@whu.edu.cn;

2. 中国科学院 长春光学精密机械与物理研究所, 吉林 长春 130033; 3. 中国科学院大学, 北京 100049)

摘 要: 在 DM6467T 平台上, 设计了视频编码存储和网络传输系统。介绍了达芬奇平台的特征、系统的硬件框架和达芬奇开发平台的搭建。调用达芬奇平台的编解码引擎中 h. 264 编码算法, 进行了 App 端多线程应用程序设计, 详细介绍了主线程、视频线程、捕获显示线程、传输线程和写线程的流程以及相互关系。完成了软硬件的集成并进行了相关应用程序测试。测试结果表明: 视频回放清晰流畅, 实时性很高; 网络传输的视频播放性能可以满足实时视频监控要求; 存储后的视频数据得到大于 100 倍的压缩比。

关 键 词: DM6467T; 视频; 编码; 存储; 传输

中图分类号: TP274

文献标识码: A

DOI: 10.3788/YJYXS20132806.0911

Video Compression-Store and Network-Transmission System Design Based on DM6467T

ZHANG Jing^{1,2,3}, DAI Ming², SUN Li-na²,

ZHAO Chun-lei^{1,2,3}, FENG Lei^{1,2,3}, YANG Tian-xing^{2,3}

(1. Key Laboratory of Airborne Optical Imaging and Measurement, Chinese Academy of Sciences,

Changchun 130033, China, E-mail: 200731020027@whu.edu.cn;

2. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China;

3. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: On the platform of DM6467T, a video system was designed which integrates the function of video encode and store as well as network transportation. The feature of Davinci platform, hardware structure of the system and the process of creating Davinci developmental environment were introduced. Using the h. 264 encode algorithm in the codec engine, the App multi-threads application program was designed. This work presented the flow of threads and the interaction between the threads, which include main thread, video thread, capture and display thread, network transportation thread, writer thread. The integration of hardware and software were completed. The test results indicate that the image of video loop back is clear and fluent, with good real-time property. The display of network-transported video can meet the demand of real-time video monitoring. The stored video data is clear and complete and the compression rate is higher than 100.

Key words: DM6467T; video; encode; store; transportation

收稿日期: 2013-03-17; 修订日期: 2013-04-05

基金项目: 空军装备预研项目 (No. 402040205)

作者简介: 张晶 (1989—), 女, 湖北武汉人, 硕士研究生, 主要从事视频监控、图像处理方面的研究。

* 通信联系人, E-mail: daim@ciomp.ac.cn

1 引 言

随着智能监控的不断发展,特别是网络视频监控的逐渐兴起,如何更好的实现大量视频数据的处理、存储和传输越来越成为视频监控行业的关注焦点。目前视频处理系统多以 ARM、DSP、FPGA 等为控制核心^[1-2],本文以 ARM+DSP 双核芯片 DM6467T 为核心进行了视频压缩存储和网络传输系统设计,引入了时下流行的视频编码标准 h.264 来对实时图像进行压缩^[3]。

该系统完成了模拟视频的采集和实时回放,并对采集到的视频进行压缩,再将压缩后的数据通过千兆网实时地传输到上位机,同时存入硬盘文件系统。实验测试结果证明,通过千兆网传输到上位机的视频码流和通过硬盘存储的 h.264 文件播放效果与压缩前相比,图像清晰度良好,并且网络传输可以基本确保实时性,同时硬盘存储占用空间与压缩前相比大大减小。

2 方案设计与实现

2.1 达芬奇平台简介

TI 公司的 TM320DM6467T 是一款为了迎合网络多媒体技术,特别是编解码技术需求而推出的一款高性能的双核片上系统^[4]。除了基本的系统控制部分和仿真接口,DM6467T 主要包括以主频为 500 MHz 的 ARM926EJ-S 为核心的 ARM 子系统和以主频为 1 GHz 的 TM320C64+ DSP 为核心的 DSP 子系统,另外该芯片还包含了两个高清视频协处理器 HDVICP0 和 HDVICP1,这两个协处理器分别拥有 1 GHz 的工作主频,与 C64+ DSP 合作总共可以达到 3 GHz 的主频,大大提高了芯片的视频处理能力。同时,DM6467T 还拥有丰富的外设资源,通过 SCR 资源交换中心和双核系统连接,大大降低了该芯片的硬件开发难度。

达芬奇平台的软件架构以嵌入式 linux 操作系统为基础,通过 TI 公司以及第三方公司的不断完善,已经拥有了非常完备的体系,双核结构通过 DSPLINK 实现通信。

达芬奇平台的算法实现基础是编解码引擎,编解码引擎包含了算法的创立(Codec)、服务的集成(Server)和引擎的集成应用(App)3 个部分^[5],TI 公司提供的 codec 中包含了各种音视频

的编解码算法库,例如 h.264、mpeg4、mpeg2、G711 等,供用户直接调用,用户只需在相应的 server 中添加所要用到的算法库,完成编译后得到 DSP 端可执行程序,再编写相应的 app 端应用程序即可完成编码功能的实现。

2.2 硬件方案

系统在电源方面,输入电源为 12 V/5 A,输出电源则有 4 种,其中 DM6467T 的内核电压为 1.2 V、I/O 电压为 3.3 V,DDR2 工作电压为 1.8 V,而 SATA 硬盘电源电压为 5 V。

系统在存储器方面,在 DM6467T 的基础上扩展了 256 MB 的 DDR2 存储器和 128 MB 的 Nor Flash 存储器。

本方案用到的仿真器为 SEED 公司的 XDS560PLUS,通过 JTAG 接口与板卡相连。

系统在视频接口部分,采用 TVP5146 解码芯片,通过复合视频接口与模拟摄像头相连,解码后的数字视频信号输入 DM6467T 的视频接口进行预处理后转换为相应的 yuv 格式视频提供给编解码引擎,在 DSP 内完成相应的编解码算法和其他图像处理算法。视频输出接口采用的是 ADV7343 编码芯片,通过复合视频接口连接 LCD 显示屏。

本方案通过 RS232 串口完成对板卡的控制,通过 MAX232 芯片与 DM6467T 的 UART 接口相连来实现 RS232 的电平转换,从而完成串口接口设计。

在网络接口方面,DM6467T 本身包含 EMAC,我们选用 VSC8211 作为 PHY 芯片,两者通过 GMII/MII 协议相连,同时 VSC8211 输出直接连接 RJ-45,该网络接口具有 10/100/1 000 M 的自适应功能。千兆网因其简易性、扩展性、可靠性、经济性、可管理维护性和广泛应用性成为了目前局域网方面主流的解决方案^[6],为本方案实时网络传输功能提供了良好的理论支持。

DM6467T 外设接口中包含了 ATA 接口,本方案为了将其转换成为时下流行的 SATA 接口,选择了 ARC772 作为转换芯片,实现 SATA 接口设计。SATA 正在成为目前主流存储设备接口,而 ARC772 在 SATA 模式下最大支持高达 3.0 Gbyte/s 的数据率,可以极好地提供各种操作,因此本方案将 SATA 硬盘作为文件系统的存储载体。

2.3 环境搭建

达芬奇平台的开发环境^[7]主要包括 4 个部

分:目标板、仿真器、windows 控制台和 linux 服务器。其中 windows 控制台通过串口与目标板相连;XDS560PLUS 仿真器通过 JTAG 接口与目标板相连,通过 USB 接口与 Windows 控制台相连;Linux 服务器则安装于 VMware 虚拟机中,方便与 Windows 系统的同时使用。Windows 控制台和 Linux 服务器以及目标板通过目标板和 PC 机的网络接口以及 PC 机的虚拟网卡实现相互之间的网络连接,只需将 3 者网络地址设置到同一网段。

在开发环境的搭建过程中,Windows 通过超级终端实现对目标板的控制,具体参数设置为波特率 115 200、数据位 8、奇偶校验无、停止位 1,数据流控制无。达芬奇相关的软件,主要包括 dvsdk、mv_pro、nfs 等,安装在 Linux 服务器的文件系统的 /opt 文件夹下,在 Linux 服务器中用到的交叉编译工具为 arm_v5t。通过 Windows 中 CCS 软件实现目标板的 U-boot 烧写,而目标板的内核以及文件系统的烧写则通过 Linux 服务器中的 tftp 实现。

2.4 软件设计

2.1 中提到,达芬奇的编解码引擎的应用包含 3 个部分。

Codec 端,本方案直接用到达芬奇 Codec Engine 中提供的编码包 h264enc,里面封装了 h.264 编码算法库文件 h264venc_tii_e.l64P,供 server 端直接调用。

Server 端,在配置文件 encode.cfg 中添加 h.264 编码模块 H264ENC,同时在该配置文件的 Server.args 数组中添加 h.264 编码选项。保存配置后编译得到 DSP 端可执行文件 encodeCombo.x64P 供 App 端调用。

App 端则在其配置文件 encode.cfg 中利用函数 Engine.createFromSever() 创建 DSP 端算法,链接到 Server 端的 DSP 可执行文件。App 端的应用程序^[8]如图 1 所示,包含 5 个线程,即主线程、捕获显示线程、视频线程、传输线程和写线程,其中除了主线程之外的所有线程均产生于主线程。

这些线程的运行是基于优先级的,其中视频线程和捕获显示线程占有较高优先级、其次是传输线程和写线程。捕获显示线程通过捕获设备捕获视频信息,并通过显示设备进行实时显示,捕获

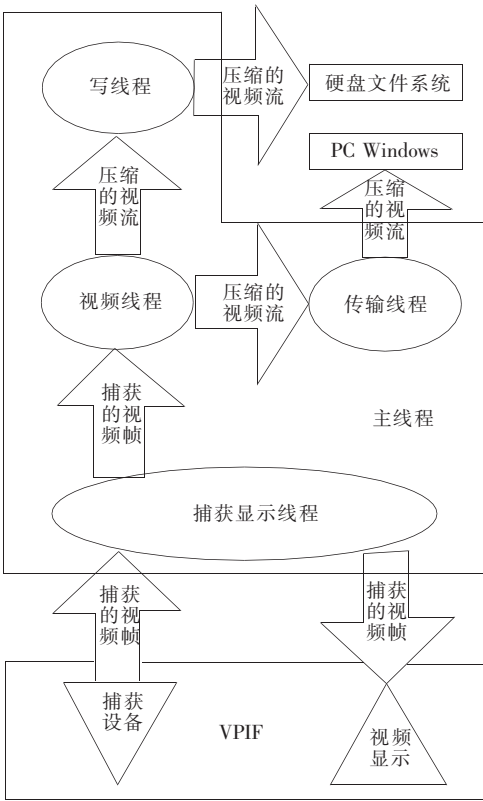


图 1 程序线程
Fig. 1 Program thread

的视频信息经 VPIF 视频接口预处理后传入视频线程进行编码,并将编码后的视频流送入传输线程和写线程,传输线程则负责将获得的编码视频流传输到相应的 IP 地址,写线程则负责将编码后的视频信息存入硬盘文件系统中指定的文件。

主线程的流程图如图 2 所示,首先在清空所有线程的环境变量以后利用 ParseArgs() 解析命令行并设置 App 环境参数,然后通过 Thread_mutex_init() 初始化互斥空间以保存全局数据,利用 setpriority() 将整个过程的优先级设置为最高,利用 CERuntime_init() 初始化编解码引擎,利用 Dmai_init() 初始化 dmai 数字多媒体应用接口,随后通过 Pause_create() 创建中断对象,通过 Rendezvous_create() 创建表征线程初始化和清空的对象,再通过线程属性相关的函数初始化并设置线程属性,接着按照一定顺序利用 Fifo_create() 函数创建线程 fifo,利用 setpriority() 函数设置线程优先级,并利用 pthread_create() 函数创建线程,具体顺序为:创建捕获显示线程 fifo、设置捕获显示线程优先级、创建捕获显示线程、创建传输线程 fifo、创建写线程 fifo、设置视频线程

优先级、创建视频线程、设置传输线程优先级、创建传输线程、设置写线程优先级、创建写线程,最后等到所有线程结束后,删除全局互斥空间,退出主线程。

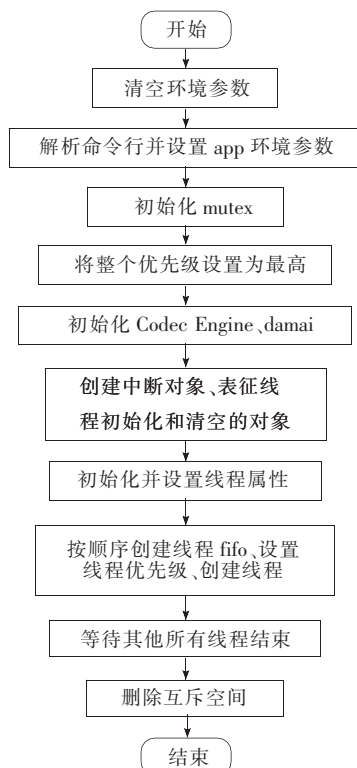


图 2 主线程流程图

Fig. 2 Flow of main thread

除了主线程以外,视频线程也是贯穿整个程序的前后,其流程图如图 3 所示,首先利用 `Engine_open()` 函数打开编解码引擎,并设置基于码率的编码参数,随后通过 `Venc1_create()` 函数创建视频编码器,并在环境变量中储存输出缓冲区大小,然后通过 `Rendezvous_meet()` 函数说明编码器创建完毕,输出缓冲区的大小可知,根据缓冲区的大小通过 `BufTab_create()` 创建相应的缓冲区,并将缓冲区送入捕获显示线程等待其它线程同步后进入主循环。在视频线程主循环中,首先利用 `Pause_test()` 测试循环是否中断,在未中断的前提下利用 `Fifo_get()` 分别从捕获显示线程、传输线程和写线程获得相应的缓冲区,然后通过 `Venc1_process()` 函数对视频数据进行编码,利用 `Fifo_put()` 函数将装有编码后的视频数据的缓冲区送入传输线程和写线程,供传输线程传入 PC 机、写线程存入硬盘文件系统,并且返还缓冲区至捕获显示线程,以便下一个循环使用。

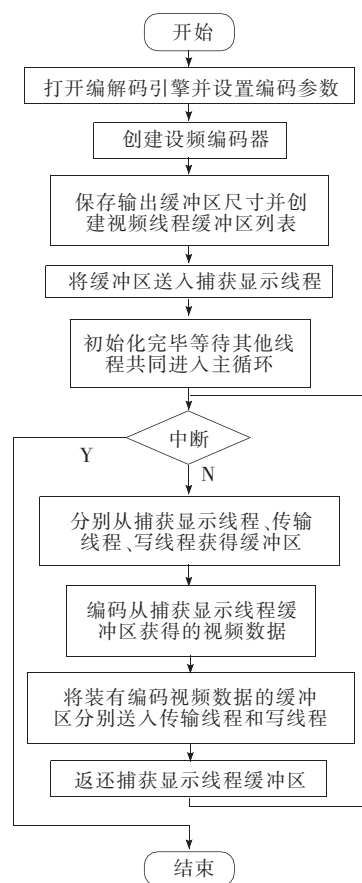


图 3 视频线程流程图

Fig. 3 Flow of video thread

本方案中视频显示是实时采集视频的复制回放,并且视频的采集和显示均在视频接口 VPIF 中进行,因此可以将捕获功能和显示功能融合在一个线程中,无需拆分为两个线程。捕获显示线程初始化的流程图如图 4 所示,首先通过视频格式和色彩空间计算所需缓冲区大小并通过 `BufTab_create()` 函数创建缓冲区列表对缓冲区进行分配,所分配的缓冲区包括捕获缓冲区和显示缓冲区两个部分,然后通过 `Capture_create()` 函数创建捕获设备驱动实例,由于本系统只支持 720×1280 像素的视频格式,因而需要检测视频格式以确保符合要求,然后通过 `Display_create()` 创建显示器件驱动实例,随后通过 `Ccv_create()` 创建 $422 \sim 420$ 色彩空间转换工作,通过 `Fifo_get()` 获取视频线程缓冲区,完成待显示和编码的图像尺寸的设置和色彩空间的配置之后便完成了捕获显示线程的初始化工作,等待其他线程同步进入主循环。在捕获显示线程主循环中,首先通过 `Capture_get()`

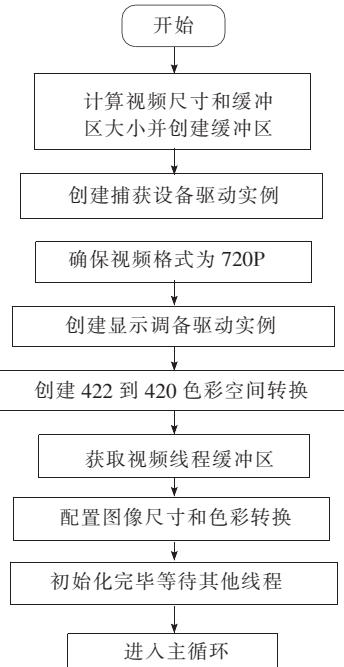


图 4 捕获显示线程初始化流程图

Fig. 4 Flow of capture and display Thread initialization

函数获取捕获设备缓冲区,通过 Display_get()对捕获的视频进行预览,然后通过 Ccv_execute()函数执行捕获缓冲区中视频格式 422~420 色彩空间的转换,并通过 Fifo_put()函数将转换后的缓冲区送入视频线程进行编码,通过 Capture_put()函数将预览视频信息送入显示设备驱动后通过 Capture_put()函数返还捕获驱动缓冲区。

本方案在网络传输方面选择的是基于 UDP 的 socket 编程,传输线程初始化过程比较简单。首先通过 socket()函数创建套接字,然后创建一个传输线程缓冲区并将其送入视频线程,即可进入主循环。在主循环中,获取装有编码后视频数据的缓冲区后,通过 sendto()函数将该数据发送至 PC 机,返还缓冲区以供下一个循环使用。

写线程首先通过 fopen()函数打开待写入视频数据的文件,然后创建一个写线程缓冲区并将该缓冲区送入视频线程以供存储待写入数据,通过 Rendezvous_meet()告知准备就绪后进入主循环。在主循环中,从视频线程获得装有编码后视频数据的缓冲区后利用 fwrite()函数将数据存入之前打开的文件,最后返还缓冲区。

由以上流程介绍可知,除主线程以外的其他线程的主循环之间以视频线程为主线,存在一定的相互作用关系,如图 5 所示。视频线程通过相

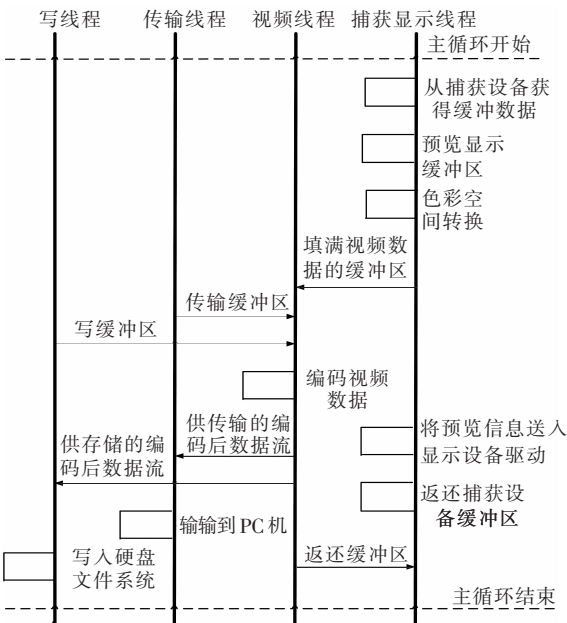


图 5 线程相互作用

Fig. 5 Interaction of threads

应的缓冲区获取捕获显示线程的视频数据实现编码功能,传输线程和写线程则通过相应的缓冲区获得视频线程编码后的视频数据,从而实现 UDP 网络传输功能和硬盘存储功能,各个线程相互作用又各自实现自己的功能,共同完成本方案的软件实现。

3 结果与讨论

3.1 图像质量

实验的硬件平台如图 6 所示,程序运行后,通过 LCD 显示屏即可看到编码前图像,截图如图 7(a)所示,图像实时性很高;传输到 PC 机上的视频流可通过 VLC 媒体播放器播放,相应的截图如图 7(b),与编码前图像相比,画面质量良好,存在一定延时,但能满足实时视频监控的要求;编码后

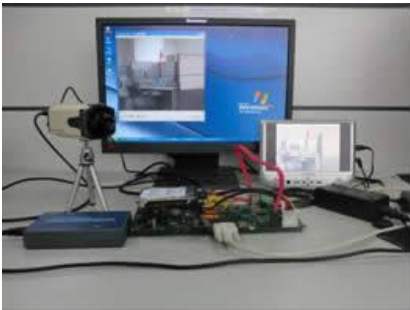
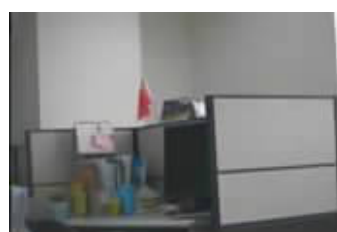


图 6 实验硬件平台

Fig 6 Platform of experimental hardware



(a)LCD 显示



(b)网络传输



(c)encode.264 文件

图 7 视频显示

Fig. 7 Video display

的码流文件存入硬盘文件系统中的 encode. 264 文件中,通过暴风影音即可进行视频播放,播放清晰流畅,画面截图如图 7(c)。

3.2 视频压缩比

本实验录制了 5 min 的视频,输入视频格式

为 PAL,帧率为 25 帧/s,分辨率为 720×576 ,未压缩的视频数据率为 125 Mbit/s,通过计算可以得到未压缩的视频数据量为 4687.5 MB,而压缩后的文件 encode. 264 容量仅为 36.3 MB,可知压缩比 129。可见,本系统在保证视频质量的前提下,大大节省了存储空间。本系统用到的 SATA 硬盘容量为 160 GB,按照该压缩比计算可以存储大约 15 天的视频信息。

4 结 论

本方案基于 TI 公司的达芬奇芯片 DM6467T,设计了视频回放和编码存储以及网络传输一体化的视频服务系统。测试结果表明:视频回放清晰流畅,实时性很高,可见 DM6467T 是视频监控领域的首选平台之一;视频编码稳定,编码后的数据流可以直接存入 SATA 硬盘,为后续的调用提供了方便,可用于机载视频监控系统;存储数据流播放清晰完整,编码压缩比可以达到 100 以上,大大节省了存储空间,方便海量数据的存储;编码后的数据经网络传输性能稳定,画面清晰流畅,满足实时视频监控要求,适用于时下流行的网络视频监控领域。DM6467T 支持多种视频输入输出格式,并且支持多路视频同时处理,对该系统的视频接口进行一定扩展,整个系统在网络视频录像机和视频服务器方面将会拥有更大的应用前景;另外该系统的 SATA 硬盘接口也是可扩展的,因此该方案在大容量视频存储方面也有很大的应用前景。

参 考 文 献:

- [1] 张秋林,夏靖波,邱婧,等. 基于 ARM 和 FPGA 的双路远程视频监控系统设计 [J]. 液晶与显示, 2011, 26(6): 780-784.
- [2] 吕耀文,王建立,曹景太,等. 移动便携图像存储系统的设计 [J]. 液晶与显示, 2012, 27(5): 697-702.
- [3] 徐兵,肖洪祥. 基于 ARM9 与 DM642 的 H. 264 视频编码服务器的设计 [J]. 计算机测量与控制, 2009, 17(2): 405-406.
- [4] Texas Instruments. TMS320DM6467T digital media system-on-chip [EB/OL]. [2013-01-17]. <http://www.ti.com.cn>.
- [5] 沈沛意,张亮,周梦,等. DAVINCI 技术剖析及实战应用开发指南 [M]. 西安:西安电子科技大学出版社, 2012: 15-17.
- [6] 郭锋. 基于千兆以太网的视频图像存储系统 [D]. 陕西:西安电子科技大学, 2009.
- [7] 合众达电子. SEED-VPM6467 用户指[EB/OL]. [2013-01-17]. <http://www.seeddsp.com>.
- [8] Texas Instruments. Encode Demo for the DVEVM/DVSDK 1. 2 [EB/OL]. [2013-01-17]. <Http://www.ti.com.cn>.