

基于 FPGA 的 H. 264 码流实时传输系统设计

马 军, 王红亮

(中北大学 电子测试技术国家重点实验室, 太原 030051)

摘要: 在研究华为海思 Hi3516A 进行视频编码压缩的过程中, 针对其压缩码流仅支持本地保存和网络流媒体应用的局限性, 提出了一种基于 FPGA 和以太网接口的 H. 264 码流实时传输方案; Hi3516A 视频编码端采用 RAW_SOCKET 原始套接字协议构建 UDP 帧, 通过网口传输 H. 264 压缩码流到 FPGA 平台; FPGA 使用一片 2 Gbit 的 DDR3 作为数据缓存介质, 保证网口速率的匹配和一次传输的 H. 264 Nalu 包的完整性; 利用 USB2.0 接口回传码流到 PC 进行功能测试; 从模拟传输本地文件和实际传输视频两方面对系统功能进行测试; 模拟传输本地文件测试中, PC 端网口发送 55, 844, 864 字节本地文件到本系统, USB 上位机接收的系统返回数据大小与发送数据大小相同, 证明数据传输完整; 实际传输视频测试中, MilkPlayer 软件播放 USB 上位机保存的码流文件, 画面流畅, 无卡顿及明显丢帧, 使用 FFmpeg 软件解码码流文件, 测试表明, 数据压缩比均值达 143: 1, 与系统设定值相比, 存在 4% 左右的误差, USB 上位机 12 h 和 24 h 保存接收码流测试中, 数据量分别达到 22.3 GB 和 43.5 GB, 码流文件播放效果良好; 因此, 此系统能实现 H. 264 码流的实时传输, 满足设计要求, 具有很好的实用价值。

关键词: Hi3516 A; H. 264; 视频传输; FPGA; 以太网; DDR3; USB 2.0

Design of H. 264 Real-time Transmission System Based on FPGA

Ma Jun, Wang Hongliang

(North University of China, National Key Laboratory for Electronic Measurement Technology,
Taiyuan 030051, China)

Abstract: In the process of studying the video encoding and compression of Huawei HiSilicon Hi3516A, in view of the limitation of its compressed code stream that only supports local storage and network streaming media applications, a real-time transmission of H. 264 code stream based on FPGA and Ethernet interface is proposed. Hi3516A uses RAW_SOCKET original socket protocol to construct UDP frame, and transmits H. 264 compressed code stream to FPGA platform through network port; FPGA uses a piece of 2 Gbit DDR3 as data buffer medium to ensure the matching of network port rate and the integrity of the H. 264 Nalu package during one-time transmission; use the USB2.0 interface to return the code stream to the PC for functional testing. The system function is tested from two aspects of simulated transmission of local files and actual transmission of video. In the test of simulated transmission of local files, the PC-side network port sends 55, 844, 864 bytes of local files to the system, and the size of the returned data received by the USB host computer is the same as that of the sent data, which proves that the data transmission is complete; in the actual video transmission test, the MilkPlayer software plays the stream file saved by the USB host computer has a smooth picture, no freezes and obvious frame loss. The FFmpeg software is used to decode the stream file. The test shows that the average data compression ratio is 143: 1. Compared with the system setting value, there is an error of about 4%, the USB host computer save and receive the stream test during 12 h and 24 h, the data volume reached 22.3 GB and 43.5 GB respectively, the streaming files play well. Therefore, this system can realize the real-time transmission of H. 264 code stream, meet the design requirements, and has very good practical value.

Keywords: Hi3516A; H. 264, Video Transmission; FPGA, Ethernet; DDR3, USB 2.0

0 引言

伴随着移动通信技术标准从 1 G~4 G, 再到 5 G 的迭代更新, 数字视频质量也从标清, 高清, 全高清到超高清发生了翻天的改变^[1], 随之而来的是海量的视频数据和传输过程中的带宽压力。多媒体处理技术的发展产生诸多对

庞大原始视频数据的视频压缩编码算法, 应用比较广泛的有 H. 264、H. 265、JPEG、MJPEG 等, 不同的编码算法适用于不同的场合, 从软件处理到硬件编码, 实现方式也不尽相同^[2-3]。国内可查的使用嵌入式平台进行视频编码的有中北大学仪器科学与动态测试教育部重点实验室, 该实验室以 TMS320C6678 DSP 平台实现的 HEVC 硬件 H. 265 视

收稿日期: 2021-03-11; 修回日期: 2021-04-06。

基金项目: 山西省“1331 工程”重点学科建设计划经费(1331KSC)。

作者简介: 马 军(1995-), 男, 山西吕梁人, 硕士研究生, 主要从事嵌入式音视频处理方向的研究。

通讯作者: 王红亮(1978-), 男, 河南安阳人, 教授, 博士, 主要从事测试系统集成、目标检测与识别、应用软件开发、超声成像等方向的研究。

引用格式: 马 军, 王红亮. 基于 FPGA 的 H. 264 码流实时传输系统设计[J]. 计算机测量与控制, 2021, 29(5): 225-229, 234.

频压缩算法, 压缩后视频数据减少大约 39~74%^[1], 效果较好。不少学者利用计算机软件进行编码算法研究, 如电子科技大学的朱艳玲提出的基于 H. 264 宏块编解码算法的研究^[4]。为了响应国家对集成电路行业的扶持政策, 本设计选用华为海思半导体的 Hi3516A 平台, 此平台支持 H. 264、H. 265、MPEG 等编码算法, 综合考虑编码效率、容错能力、网络亲和性等因素, 选择 H. 264 算法进行硬件编码。海思平台应用于安防监控领域的多种场景, 侧重 IP 摄像机的系统布局^[5-6]。但面对摄像头采集的视频数据生成的编码码流远距离传输问题, 海思平台就显得捉襟见肘^[7], 因此本系统利用海思平台网络接口创造性地使用 RAW_SOCKET 协议将编码码流传输至 FPGA 平台^[8-9], FPGA 平台能很好地实现较于 TCP (transmission control protocol) 协议相对简单的 UDP (user datagram protocol) 协议, 可成功对接海思平台, 完成 H. 264 码流的接收。本文着重于介绍如何利用 FPGA 进行 H. 264 码流传输, 最后介绍系统功能的测试。

1 方案设计

本系统主要由 H. 264 码流生成源 Hi3516A 平台、网口模块、DDR3 模块、USB2.0 模块和 USB 上位机软件组成, 系统框图如图 1 所示。

系统主控制器选择赛灵思公司 spartan6 xc6slx45, 此芯片含有 43 661 个逻辑单元, 54 576 个触发器, 116 个 18 kB 块 RAM, 最大用户 IO 可达 358 个, 且集成有 GTP、PCIE、MCB 等高速硬件控制器, 在电子工程领域应用广泛, 因此足以支持本设计的资源需求^[10-11]。Hi3516A 平台负责采集 HDMI 摄像头数据, 采用 H. 264 算法进行编码压缩, 平均码流速率控制在 4 Mbps; 生成的码流由 FPGA 平台网口模块接收, 该模块实现 UDP 协议; 系统选用 1 片 2 Gbit DDR3 芯片作为缓存介质, 以解决速率匹配和数据缓存问题; 系统功能的验证由 USB2.0 模块将码流数据 PC 回传, 上位机软件接收并本地保存, 而后线下进行数据完整性、视频播放流畅性等系统功能测试分析。

2 H. 264 码流源生成模块介绍

Hi3516A 是华为海思半导体公司推出的一款多功能通信媒体处理芯片, 旨在提供新一代 HD IP 摄像机 SOC 方案, 其内部集成 ARM Cortex-A7 处理器内核及 H. 264、H. 265、MJPEG、JPEG 等硬件编码器, 主频可达 600 MHz; 具有多

路视频实时编码能力, 支持的最大输入分辨率可达 5 M Pixel; 输入时序为 8/10/12/14 bit RGB Bayer DC 时序, BT. 601、BT. 656、BT. 1120 和 MIPI、LVDS/Sub-LVDS、HiSPi 等; 支持 RGMII/GMII 网口输出。此芯片能满足监控领域的各种场景需求^[12-13]。Hi3516A 采用 ARM+H. 264/AVC 硬件编码器的方案实现 H. 264 编码, 内部各硬件模块和控制器之间采用 ARM 公司先进微控制器总线结构 AMBA3.0 实现高速互联, 性能强大^[14-15]。

本设计选用 Hi3516A 作为 HDMI 高清摄像头数据压缩编码平台, 利用网口输出编码码流来为系统提供 H. 264 码流源。软件工作流程如图 2 所示。

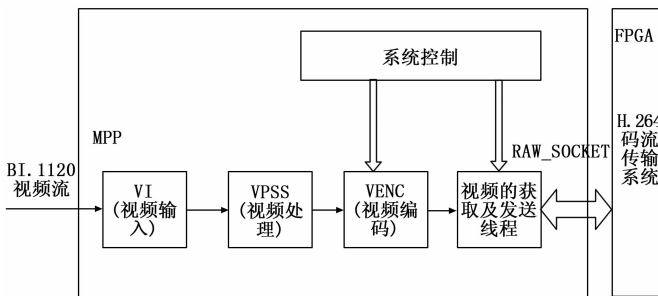


图 2 Hi3516A 软件实现流程图

鉴于 H. 264 码流接收端采用 FPGA 作为主控制器这一场景, 以太网帧的构建从前导码、帧开始符、Mac 帧头、IP 层到 UDP 层均需用户一一实现, 且开始接收以太帧的判定逻辑是识别 Mac 帧头的目的 Mac 地址是否为用户所设定的。如此, 在 Hi3516A Linux 嵌入式环境下编程实现普通 socket 协议难以满足该场景需求, 普通 socket 协议是以用户 IP 地址为网络通信基础的。与 Linux 系统可实现复杂的网络协议栈相比, FPGA 仅实现简单的 UDP 协议。为避免 Linux 系统和 FPGA 平台对接时与数据传输无关的其他协议对系统产生干扰, 在此 Hi3516A 嵌入式平台使用 RAW_SOCKET 协议实现 UDP 编程, 用户指定目的 Mac 地址使 FPGA 能准确接收 UDP 数据包。结果表明, 此种方式成功解决了 Hi3516A 和 FPGA 平台的对接问题。

3 FPGA 控制逻辑设计

系统由以太网控制模块, DDR3 控制模块和 USB 接口控制模块组成, 以太网控制模块完成 H. 264 码流的接收, DDR3 完成速率匹配和数据缓存功能, USB2.0 模块完成数据 PC 回传以便系统功能的测试分析。

3.1 以太网接收控制逻辑

UDP, 即用户数据报协议, 适用于图像视频传输及网络监控数据交换等对传输速度要求较高的场合, UDP 协议自带寻址信息, 只提供端到端的数据传输服务。与 TCP 的复杂性相比, UDP 协议在 FPGA 平台具有更高的可实现性。

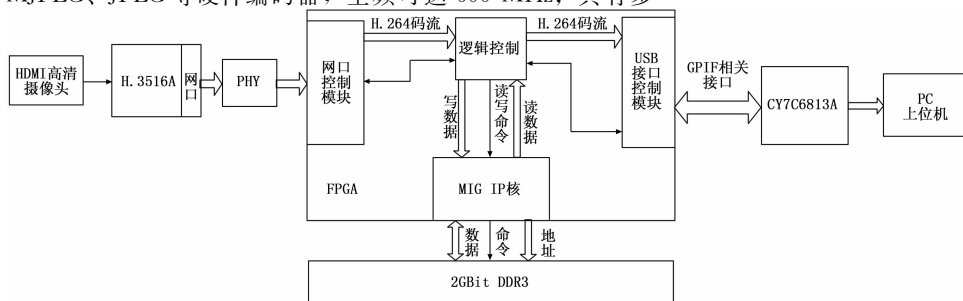


图 1 系统总体框图

综合考虑, 系统选用 UDP 协议来完成码流接收。完整的以太网帧结构层次和 UDP 接收数据状态转换逻辑如图 3~4 所示。

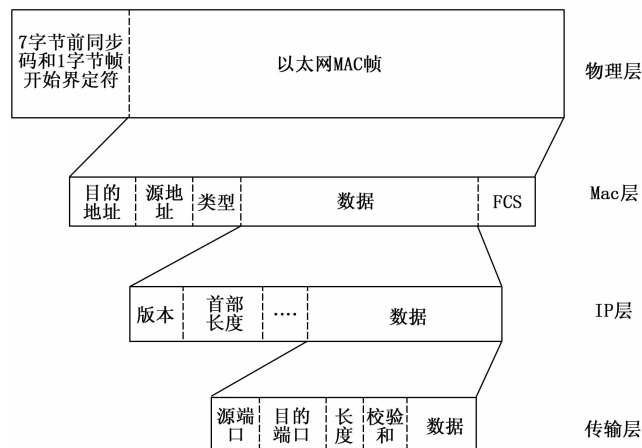


图 3 以太网帧结构层次

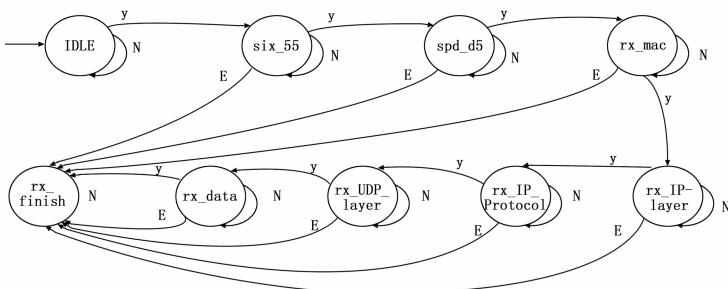


图 4 UDP 接受数据状态转换逻辑

3.2 DDR3 控制逻辑

硬件上选用 Micron 公司 2Gbit 容量的 DDR3-SDRAM 存储芯片 MT41J128M16LA-187E 作为缓存介质, 赛灵思公司 ISE 开发软件提供 MIG (Memory Interface Generator) IP 核来为 xc6slx45 生成 DDR3 控制器设计模块, MIG IP 核其逻辑框图如图 5 所示^[16]。此种方式屏蔽 DDR3^[17] 复杂的接口时序设计, 简化为简单命令 FIFO 和数据 FIFO 的操作, 用户可通过 USER Logic 来灵活快速地设计自己的 DDR 控制逻辑。

本系统将 MIG IP 核配置为双端口实现码流的边写边读功能, 端口 Port 0 和 Port 1 均为 64 bit 宽度, 时钟速率为 312.5 MHz。DDR3 写 FIFO 时有数据 64 bit 对齐要求, 此模块将以以太网模块接收的 H. 264 码流字节组成 64 bit 宽度, 以突发写模式连续写进 DDR3 中, 突发长度 1。DDR3 的读取控制信号由 USB 2.0 模块产生, 以突发读模式读取 DDR3 中数据, 突发长度为 64, 即一次读取 512 字节 (64 * 64 bit/8) 的数据块。为保证每次读 DDR 信号到来时读 FIFO 中有足够的有效数据, 本模块设计了写计数变量 receive_counter 和读计数变量 read_counter, 只有当写计数大于读计数时才生成有效的读命令信号。图 6 为 DDR 读写数据流模式, 图 7 为 DDR 读写时 chipscope 抓取到的

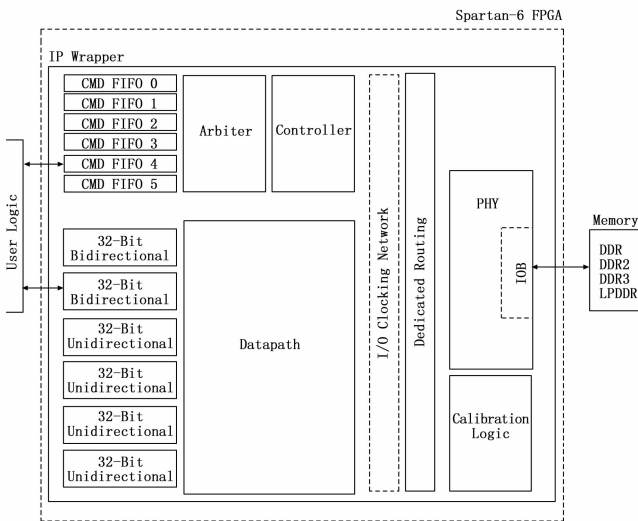


图 5 MIG IP 核逻辑功能示意图

H. 264 码流 I 帧 Nalu 单元起始码 (00 00 00 01)。

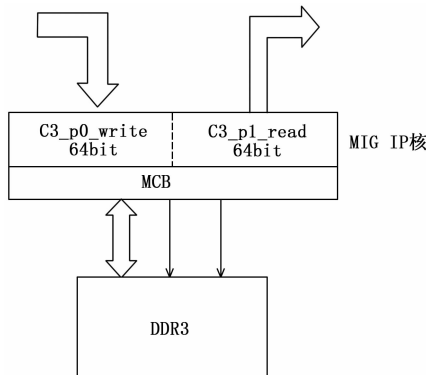


图 6 DDR 读写数据流模式

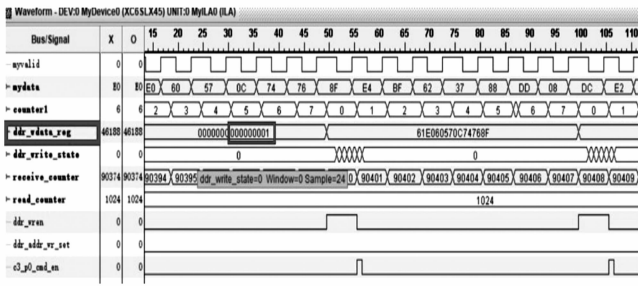
3.3 USB2.0 控制逻辑

采用 Cypress 公司 EZ-USB FX2LP 系列的 CY7C68013A 作为 USB 接口芯片, 此芯片内部集成 8 051 微控制器, 可支持的时钟为 48 MHz, 24 MHz, 12 MHz, 且含有小容量片上 RAM, 满足其固件程序的运行。该芯片自带的 4 kB FIFO 足以支持其在 Master 模式或 Slave 模式下所有数据传输应用场合^[18-19]。

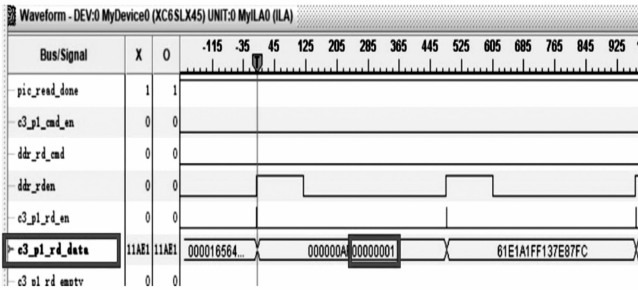
本模块通过配置 FX2 固件程序使该芯片工作在 Slave FIFO 异步读写模式, EP2 输出端点和 EP6 输入端点各配置 2 个 512 B 的 FIFO, 其中一个 FIFO 写满或读空的情况下会转向另一个 FIFO 的读写。FX2 固件相关寄存器配置如表 1 所示。

4 系统功能测试

通过修改 FX2 官方提供的上位机源码添加接收数据的本地保存功能, 此上位机可实现接收成功时数据自动保存到本地文件功能, 输入输出端口选择功能, USB 设备选择功能, 传输数据块大小选择功能 (仅支持 512 倍数的数据块大小), 数据接收成功失败计数功能 (以 512 B 或 kB 为单



(a) 写DDR抓取到的起始码



(b) 读DDR抓取到的起始码

图 7 DDR 读写过程抓取 Nalu 起始码

位)。界面如图 8 所示。

表 1 FX2 固件相关寄存器配置

寄存器	值	功能
IFCONFIG	0X4B	配置 FX2 GPIF 接口为异步 Slave fifo 模式
PINFLGASAB	0XC8	配置 FLAGA 为 EP2 空标志, FLAGB 为 EP2 满标志
PINFLAGSCD	0XDDE	配置 FLAGC 为 EP6 满标志
PORTACFG	0X40	配置 PA7 为 SLCS#
FIFOPINPLOAR	0X00	配置所有信号低电平有效
EP2CFG	0XA2	配置 EP2 为 OUT 端口, 双 512 字节 FIFO
EP6CFG	0XE2	配置 EP6 为 IN 端口, 双 512 字节 FIFO

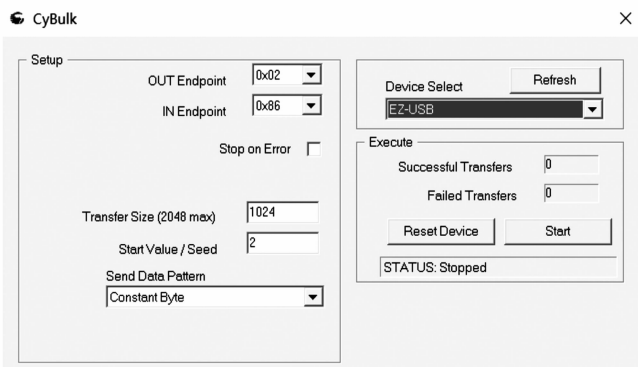


图 8 USB 上位机界面

系统功能的测试从模拟传输本地文件和实际传输 Hi3516A 编码生成的 H. 264 码流两方面进行。模拟本地文

件传输示意图如图 9 所示, PC 端使用 RAW_SOCKET 编程实现 UDP 协议, 将本地 8 023-2 015.pdf 文件 (55 844 864 字节) 通过 PC 端网口发送到此 H. 264 码流传输系统。

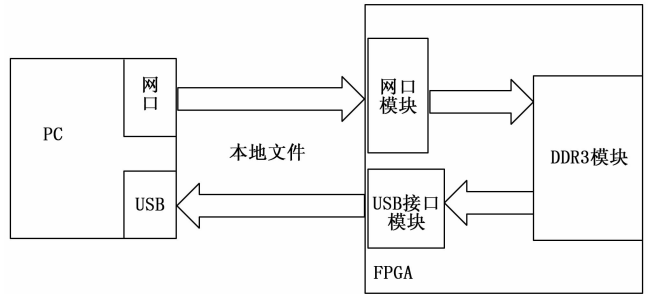
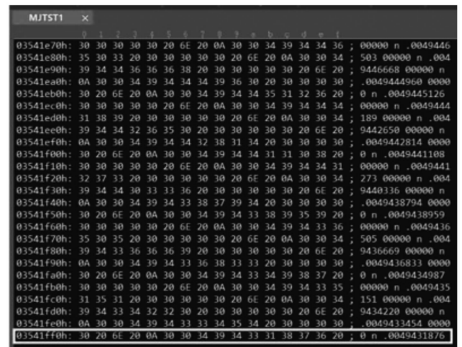
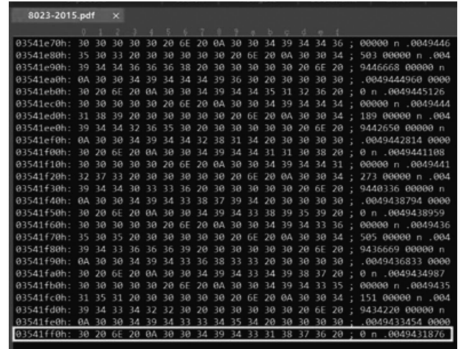


图 9 模拟本地文件传输示意图

通过对比原始文件和上位机接收到的数据文件来测试系统传输数据的完整性。使用 UltraEdit 软件打开原始文件和接收到的数据文件, 对比结果如图 10 所示。



(a) 系统传输后接收文件



(b) 原始本地文件

图 10 上位机接收文件和原始本地文件数据对比

可发现经系统传输后接收文件从开始到最后地址 0x03541f10 和原始文件数据相匹配, 数据大小皆为 55 844 864 字节, 因此可得出数据传输完整的结论。

H. 264 码流传输实物图如图 11 所示。

HDMI 摄像头, Hi3516A, FPGA 依次上电, 系统稳定 3 s 后, 在 Hi3516A 终端串口输入命令 ./sil9135_test, 测试 HDMI 摄像头是否成功识别, 如图 12 所示。

串口打印结果 mode: progressive 表示视频为逐行扫描格式, w: 1920 表示视频分辨率为 1 920 * 1 080。终端输入 ./Hi3516A_hdmi_rawsocket 命令启动系统编码, 2 s 等待编



图 11 系统实物图



图 14 FFmpeg 解码 H. 264 码流文件

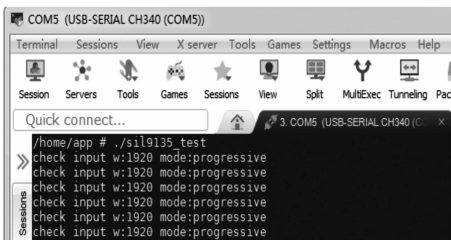


图 12 HDMI 摄像头视频输入测试

如图 14 所示, 从解码结果中可提取视频时长、图像大小、比特率、帧率、帧数、文件大小等信息, 统计结果如表 2 所示。

表 2 FFmpeg 解码 H. 264 码流文件结果统计

	MJTST1	MJTST2	MJTST3	MJTST4	MJTST5
H. 264 文件大小/MB	58.2	116	178	234	295
解码后 YUV 文件大小/MB	8 315	1 6591	2 5451	33 362	42 021
压缩比	143	143	143	143	142
帧数	2 803	5 593	8 580	11 247	14 166
视频时长/s	112	223	343	449	566
码率/kbps	4 256	4 261	4 251	4 271	4 269
图像大小	1 920 * 1 080	1 920 * 1 080	1 920 * 1 080	1 920 * 1 080	1 920 * 1 080
帧率/fps	25	25	25	25	25

码延迟后, 启动 USB 上位机接收码流数据, 上位机显示接收数据量大于 51 200, 即 50 MB 字节时, 停止接收数据。

通过 MilkPlayer 软件播放保存的码流文件, 系统传输后码流文件播放和摄像头原始图像显示对比如图 13 所示。



(a) 码流文件播放图像



(b) 摄像头原始图像

图 13 码流文件播放图像和摄像头原始图像对比效果

由图 13 可知, 画面显示完整, 细节清晰, 播放流畅, 无明显卡顿及花屏绿屏等丢帧乱码现象。

利用 FFmpeg 软件^[20]解码上位机接收到的码流文件为 YUV4: 2: 0 格式, 解码命令为 `ffmpeg -i MJTST.h264 -vcodecrawvideo -an MJTST.yuv`。参数 `i` 指定输入文件, `vcodec` 指定解码对象为视频文件, `rawvideo` 指定解码为原始 yuv 格式视频数据。如图 14 所示。

其中, 平均码率为 H. 264 文件大小与视频时长的比值, 帧率为帧数与视频时长的比值, 压缩比为解码后 YUV 文件大小与 H. 264 文件大小的比值。由表 2 统计结果可知图像大小为 1 920 * 1 080, 视频帧率为 25 fps, 符合编码设定参数, 由 5 组数据可计算平均码率为 4 261.6 kbps, 与设定的码率 4 096 相比, 有 $(4 261.6 - 4 096) / 4 096 = 4\%$ 的误差, 平均压缩比为 143, 编码效果良好。

经测试, 系统在 12 h 左右本地保存数据量达 22.3 GB, 24 h 左右数据量达 43.5 GB, 且播放存在部分卡顿现象, 2 s 左右能自动恢复流畅播放, 可认为系统能够完成长时间传输及存储码流的任务。

综合上述分析, 可得出系统数据传输完整, 编码效果良好, 传输及存储码流时长达到设计要求的结论, 此系统成功实现了 H. 264 码流实时传输的设计任务。

5 结束语

此设计实现的 H. 264 码流传输系统能成功对接 Hi3516A 平台的网口, 将编码码流通过 FPGA 进行处理和转发, 在海思半导体平台原有 IP 摄像机和网络流媒体应用方案的基础上, 极大拓展了其应用可能性, 如在航空航天等一些需要大量图像数据远距离传输的场合, 此系统可突破原网络双绞线

(下转第 234 页)