

文章编号: 1007-2780(2012)01-0081-06

微投影视频信号的 USB 传输系统设计

程作霖, 郑天津, 刘云川, 龚向东*

(深圳大学 电子科学与技术学院, 广东 深圳 518060, E-mail: czl_128@126.com)

摘 要: 设计并实现了一种基于 USB2.0 的微投影视频信号传输系统。系统利用上位机程序实时抓取播放器画面图像来获取视频数据, 并通过 USB 接口进行数据传输, USB 设备端数据的采集控制通过 FPGA 外部逻辑电路实现。系统在 FLCOS 微投影实验平台上进行了验证, 结果表明系统传输速率达到 18.75 Mbyte/s, 能够实现 16 位高彩色、640×480, 30 帧/s 视频数据的实时传输。

关 键 词: 微投影; USB2.0; 视频传输; FPGA

中图分类号: TN873

文献标识码: A

DOI: 10.3788/YJYXS20122701.0081

Design of USB Transmission System for Microprojection Video Signal

CHENG Zuo-lin, ZHENG Tian-jin, LIU Yun-chuan, GONG Xiang-dong*

(College of Electronic Science and Technology, Shenzhen University, Shenzhen 518060, China, E-mail: czl_128@126.com)

Abstract: A USB2.0-based transmission system of micro-projection video signal is designed and implemented. In the system, frame images and the data of video on a media player are grabbed in real time, which are transmitted with USB interface and acquired with external FPGA logic circuit on the USB equipment. The design is verified on a FLCOS microprojection platform, and it shows that a transmission rate of 18.75 Mbyte/s is obtained, and video of 16-bit color, 640×480 resolution and 30 frame/s can be transmitted in real time with the system.

Key words: microprojection; USB2.0; video transmission; FPGA

1 引 言

微投影是基于微显示器件实现高清晰、大屏幕、便携式信息显示的新型投影技术, 具有广阔的应用市场^[1-2]。计算机中可以存放大量图像和视频文件, 是微投影的一种基本媒体数据源。在计算机的众多外设接口中, USB 具有即插即用、热插拔及高速传输等特点, 是用于视频数据传输的一个标准渠道^[3]。计算机中的视频图像文件有多种编码方式 (MPEG、AVI、WMV、MOV 等), 本

文所述的微投影视频 USB 传输系统利用现有的媒体播放器, 采用实时抓屏的方法实现视频图像帧的获取, 并由图像帧提取 RGB 三基色数据, 通过 USB 传输至微投影设备, 这样能有效避免上位机对不同视频格式文件繁琐的软件解码处理, 拓宽了系统对不同视频格式的适用性。此外, 采用图像处理与数据传输双线程技术, 解决了由于单线程进程间相互等待引起的传输不连续问题。本系统采用 FPGA 器件控制 USB 数据的采集, 其现场可编程的特点使系统电路的开发及改进更为

收稿日期: 2011-08-18; 修订日期: 2011-09-14

基金项目: 广东省大学生创新实验项目 (No. 1059010035)

作者简介: 程作霖 (1991-), 男, 广东茂名, 大学本科, 研究方向为电子科学与技术专业。

* 通讯联系人, E-mail: gongxd@szu.edu.cn

方便灵活。

2 系统硬件设计

系统硬件电路包括 USB 接口电路和 USB 外部逻辑电路,后者由 FIFO 接口、SDRAM 控制器、帧频控制、读写控制和 FLCOS(铁电硅基液晶)面板接口等模块组成,系统电路的硬件结构如图 1 所示。USB 接口电路负责解析 USB 总线协议,通过 USB 总线读取上位机传送的视频流数据,并在外部逻辑电路的控制下以图像帧的形式缓存于 SDRAM 中,再由 FLCOS 面板接口送至微投影设备显示。在外部逻辑电路中,帧频控制模块根据视频帧率调节视频流的传输速度;通过对 FIFO 接口和 SDRAM 控制器的读写控制, FIFO 接口从 USB 接口电路的 FIFO 中读完数据后方可写入 SDRAM 中,在数据存入 SDRAM 后,才可以继续从 USB 中读数据,避免了 SDRAM 重写或漏写数据。

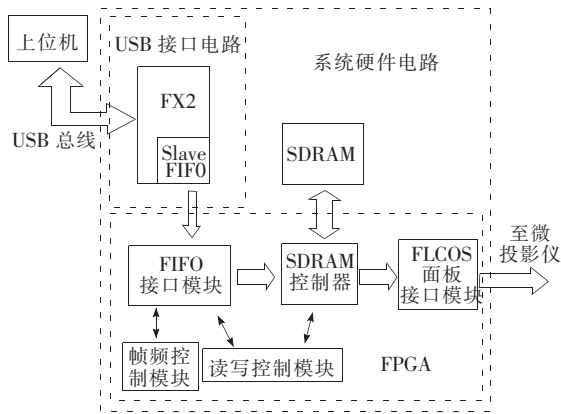


图 1 系统硬件结构

Fig. 1 Hardware structure of the system

2.1 USB 2.0 接口

USB 接口电路采用 CYPRESS 公司的 EZ-USB FX2(CY7C68013A)^[4]实现,其内部结构如图 2 所示。作为一款高效的 USB2.0 控制器,FX2 集 USB 通信引擎和增强型 8051 内核于一体,无需外加芯片即可完成高速 USB 传输。FX2 提供了一种“量子 FIFO”的独特处理架构,使 USB 接口和外部应用环境直接共享 FIFO,而 8051 微控制器可以不参与数据传输,较好地解决了 USB 高速模式的带宽问题。

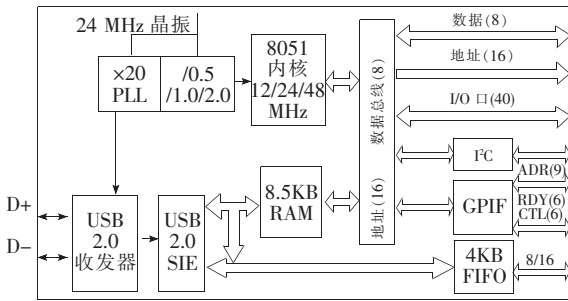
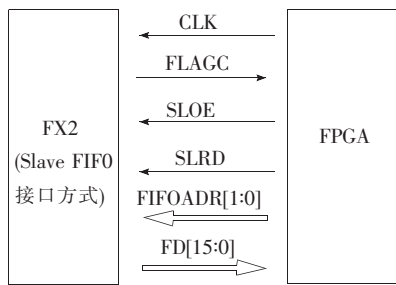


图 2 FX2 的内部结构

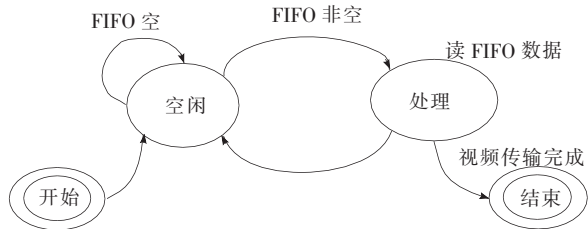
Fig. 2 Inner structure of FX2

2.2 USB 外部逻辑电路

FX2 具有 Slave FIFO 和 GPIF 两种接口方式^[5]。Slave FIFO 是从机方式,即 FX2 的 8051 内核不直接参与 USB 数据传输,而是由 FX2 内部的 USB 接口引擎直接把数据传到 FX2 的 FIFO 中,再通过外部逻辑器件(如 FPGA)读出。在此方式下,FX2 提供满、空、读、写等控制信号,以供外部逻辑器件实现对 FIFO 的读写控制。本设计选用 Slave FIFO 接口方式,USB 外部逻辑电路采用 FPGA 设计实现,其与 FX2 芯片的接口电路如图 3(a)所示,其中,FLAGC 为 FIFO 的空标志信号,SLRD 为 FIFO 读控制信号,SLOE 为读允许信号,FIFOADR [1:0]为 FIFO 的端点选择地址线。FPGA 中的 FIFO 接口根据 FLAGC



(a)



(b)

图 3 FX2 同步 Slave FIFO 传输的接口及其状态图
Fig. 3 Synchronous slave FIFO interface of FX2 (a) and its state diagram (b)

的电平来判断 FX2 的 FIFO 数据状态, 只要 FIFO 非空, 则读数据。FPGA 中 FIFO 接口的工作状态如图 3(b) 所示。

帧频控制模块控制 FIFO 接口模块读取视频流数据的速率, 使其读取每帧视频数据的频率与视频的帧频相等, 从而使 SDRAM 中帧缓存的数据刷新率与此一致, 最终达到微投影仪以同样帧频播放视频的效果。上位机在每个视频数据传输开始前都将包含该视频的帧频及分辨率等信息传送给 FPGA, 并在设计中约定帧结束标志为 16'haaaa, 该标志也为新一帧的开始标志。帧频控制模块通过将视频传输过程中实际读完一帧视频数据所用时间与该视频的帧周期比较, 判断视频流数据读取是否过快, 若过快则插入等待周期, 直到本帧周期结束, 新一帧周期开始。帧频控制模块的 Verilog HDL 描述代码如下:

```

reg sign_f;
//实际读取数据过快的标志
reg sign_d;
//每帧完成的标志
reg [28:0] clk_fact;
//每帧图像读取实际花费的时钟周期数
reg [28:0] clk_standard;
//所传输视频的帧周期
assign empty1 = (sign_f == 0)? empty:
1'b0; //
//判断数据读取是否过快, empty1 为 0 时
暂停读数据
always @(negedge CLK) begin
    clk_standard <= 5000000/frame;
    //CLK 频率:50MHz, frame: 视频帧率
    if (data_from_usb == 16'haaaa)
    //本帧数据读完
        sign_d <= 1;
    else if (clk_num < clk_standard && sign_d
== 1) begin //读取数据过快
        sign_f <= 1; //过快标志位有效
        clk_fact <= clk_fact + 1;
    end
    else if (clk_fact >= clk_standard &&
sign_d == 1) begin
        //本帧结束, 新一帧开始
        sign_f <= 0;

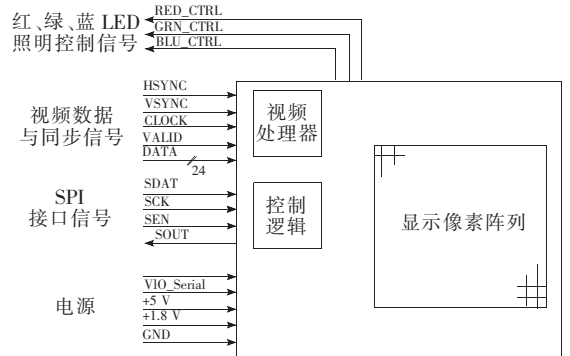
```

```

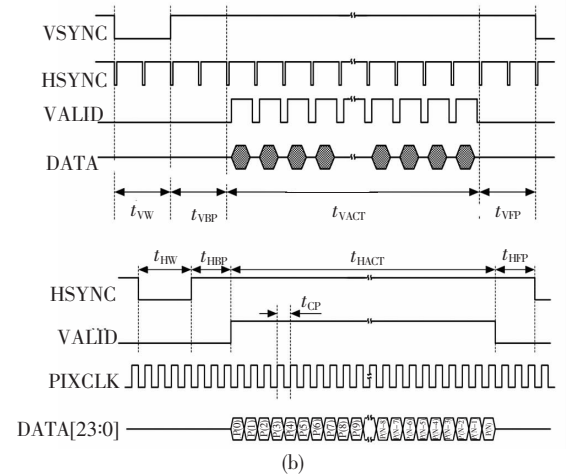
sign_d <= 0;
clk_fact <= 0;
end
else
    clk_fact <= clk_fact + 1;
end

```

SDRAM 中帧缓存的读写主要是通过 SDRAM 控制器来实现的^[6]。该控制器使用方便, 只需给出读或写的起始和结束地址即可在读或写时钟的配合下实现数据读写。SDRAM 控制器接口中的 rd_load、wr_load 分别为读、写刷新信号, 当信号有效时, 相应的读或写地址将回到初始位置。为防止投影图像的漂移, 需保证微投影仪投影视频时每场扫描的第一个数据为该帧图像的的第一个像素点。为此, 要求每帧视频图像的的第一个像素点都能存储在帧缓存的起始位置上, 即在存每帧视频图像的的第一个像素点时刷新 SDRAM 缓存, 以使写地址回到起始位置。同理,



(a)



(b)

图 4 FLCOS 面板接口信号及显示扫描时序
 Fig. 4 Interface signals and display timing of the FLCOS panel

在读帧缓冲区时,要求微投影仪投影视频时每开始新的一帧扫描时,读地址能回到起始位置,这可对 rd_load 信号进行类似处理来实现。

FLCOS 微投影系统由光源、光学引擎和系统电路组成,光学引擎的功能组件包含 FLCOS 微显示面板和光学系统,其中光学系统的主要作用是对光源光束进行整形(扩束、准直、匀光等)及微显示图像的投影放大。本设计应用的微投影系统采用 Micron 公司的 FLCOS 微显示面板^[7],其接口信号和显示扫描时序分别如图 4(a)和图 4(b)所示。FLCOS 面板接口电路设计包括符合图 4(b)定时要求的帧同步(VSYNC)、行同步(HSYNC)、数据有效(VALID)等同步信号的产生,同时按照帧扫描时序和行扫描时序要求,从 SDRAM 帧缓存中读取 USB 传输的视频像素数据并调整为 FLCOS 面板输入规定的 R、G、B 三色 24 位数据(DATA)格式。除视频数据与同步信号外,该 FLCOS 面板接口信号还包括输出的光源照明控制信号和进行器件初始化工作配置的 SPI 串行总线信号,后者由微投影系统电路中嵌入式 MCU 的 SPI 接口实现。

3 系统软件设计

本 USB 传输系统的软件设计主要包括上位机应用程序和 FX2 固件程序的设计,前者实现视频帧的获取、帧图像的处理以及上位机与 USB 设备的通信;后者控制 FX2 响应 PC 机的请求,完成 USB 设备重枚举,实现对 USB 设备的设置。

3.1 上位机程序

在上位机中采用现有主流的媒体播放器播放时所需传输的视频,上位机应用程序仅需实时抓取播放器画面图像,并将其转化为 bmp 位图,从而获取 bmp 位图的 RGB 三色数据,经过进一步处理后,按 USB 设定的管道号以及端点号通过块传输方式发送至 FX2 上。采用抓屏方式,避免了上位机中不同格式视频文件软件解码处理的复杂性。

上位机程序通过 VC 编程实现,程序中每帧图像数据的获取处理和数据传输是占用时间较大的两个进程。为了保证 USB 的高速传输及连续性,本设计采用多线程技术,分别负责数据的获取处理和数据的传输,上位机程序流程如图 5 所示。

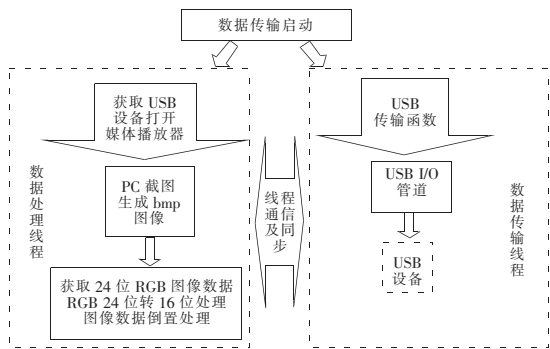


图 5 上位机程序流程图

Fig. 5 Program flowchart on host computer

bmp 位图文件结构包括位图文件头、位图信息头、颜色表以及位图数据 4 部分^[8],其中位图信息头偏离位图数据 40 字节,数据传输时需从数据块的第 41 字节开始,本设计为真彩色传输,不需要颜色表部分。在图 5 的视频处理中,PC 截图需获取屏幕的分辨率以及内存设备描述表,由 API 函数 CreateDC 为屏幕创建设备描述表,CreateCompatibleDC 为屏幕设备描述表创建兼容的内存设备描述表,CreateCompatibleBitmap 创建一个与屏幕设备描述表兼容的位图,从而提取 24 位真彩色图像的 RGB 三颜色数据。由于数据的储存方式为 BGR,且帧图像数据由下到上,由左到右,所以必须进行数据倒置处理后发送。考虑到 USB2.0 的传输带宽,将每个像素 24 位的 RGB 数据处理为 16 位的 RGB 数据,其中 RGB 有效数据为 5:5:5 格式,最后一位数据为标志位,用于区别是否为有效像素点数据。另外,图像数据存储是以 4 个字节的整数倍存储的,所以在传输时需判断图像宽度是否为 4 个字节的整数倍,如果不是则要将其处理为 4 个字节的整数倍。

上位机应用程序通过 I/O 控制来访问设备的驱动程序。应用程序首先通过调用 Win32 API 函数 CreateFile() 来取得访问设备驱动程序的句柄;进一步通过 Win32 API 函数 DeviceIoControl 来提交 I/O 控制码,并且为 CreateFile() 函数返回的设备句柄设置 I/O 缓冲区。该系统中的缓冲区 FIFO 为 1 024 字节,端口非空即读取。这样就保持了一帧图像传输的连续性。图 6 为上位机与 USB 设备通信示意图,上位机通过匹配设备名是否正确,并匹配 USB 的传输方式(块传输)以及管道号,返回成功则可进一步操作。在上位机

与设备数据传输时, 上位机需向设备发送视频帧率以及视频的长宽, 然后发送视频图像数据。

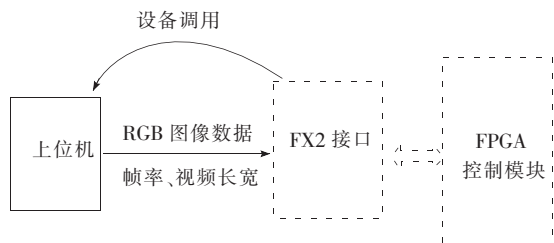


图 6 上位机与 USB 设备的通信

Fig. 6 Information transmission between host computer and USB equipment

3.2 FX2 固件程序

在现有 FX2 固件程序架构的基础上^[5], 固件程序的编程主要为 FX2 端点、传输方式和接口方式的配置。

FX2 芯片有小端点 (EP0、EP1IN 和 EP1OUT) 和大端点 (EP2、EP4、EP6 和 EP8) 两组端点。大端点 2、4、6、8 是大容量高带宽的数据传输端点, 可以配置为各种带宽以满足实际需要, 端点 2、4 是输出端点, 其中端点 2 可配置为 512 byte/s 或 1 024 byte/s, 并可配置为 2、3、4 倍的缓冲深度。本设计为单向传输, 只需一个端点来向外传输数据。考虑到传输视频流数据需要较高的带宽, 故选用大端点 EP2, 并将 EP2 配置为 512 byte/s, 4 倍缓冲深度。FX2 端点的数据传输到 FIFO 中时

有手动和自动两种方式。自动传输方式允许端点数据连续填充 FIFO, 而不需 FX2 的 8051 内核的参与。本设计选用自动传输方式, 同时选用同步 Slave FIFO 接口方式。

FX2 端点、传输方式和接口方式的配置主要通过固件程序的初始化函数 TD_Init() 对 FX2 的相关寄存器的设置来完成。

3.3 USB 驱动程序

当设备的 USB 线连接到计算机时, 上位机端的 USB 驱动程序自动地将固件下载到 FX2 上, 实现 USB 设备的热枚举。本驱动程序是在 EZ-USB FX2 软件开发包提供的 EZ-Loader 驱动程序基础上设计完成^[9]。驱动程序中主要用到 2 个 API 函数: CreatFile() 和 DeviceIoControl()。CreatFile() 的功能是取得设备句柄; DeviceIoControl() 的功能是向设备驱动程序发送请求, 它通过调用 ezusb.sys 来实现。ezusb.sys 是基于 EZ-USB 外设接口的通用设备驱动程序。为使驱动程序正确工作, 用户还须编写一个安装信息文件 (inf 文件)。该文件会将用户的 USB 设备绑定到相应的设备驱动程序上。

4 设计验证与应用

本文描述的微投影视频信号 USB 传输系统通过了 FPGA 设计软件 Quartus II^[10] 的设计验证。作为一个示例, 将数字 0~16 从上位计算机

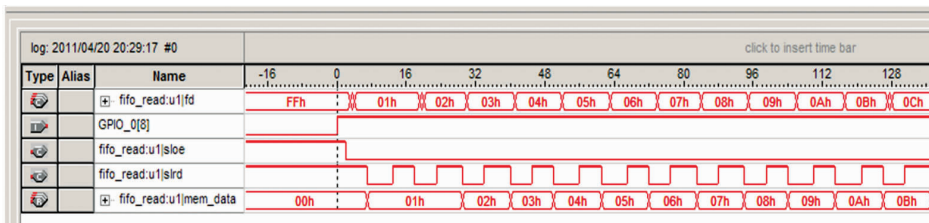


图 7 设计验证的相关信号时序示例

Fig. 7 Timing diagram of a demonstrated verification for the design

中经 USB 接口传输到 FPGA 再由微投影仪从 SDRAM 缓存中读出, 图 7 为采用 Quartus II 内嵌的逻辑分析仪工具 SignalTap II 观测到的相关信号时序, 其中第一行数据为 FPGA 从 USB 接口芯片读取的数据, 最后一行为 SDRAM 读出的数据。

本 USB 传输系统在我们研制的 FLCOS 微投影实验平台上得到了实际应用, 图 8 为该微投影实验平台的实物图及其投影效果图。

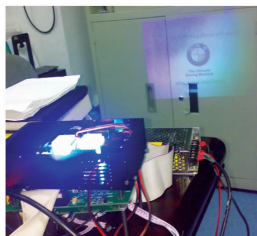


图 8 微投影实验平台实物及其投影效果

Fig. 8 Picture of the design applied on FLCOS microprojection platform

5 结 论

介绍了一种基于 USB2.0 协议的视频数据传输系统,该系统能够获取计算机上丰富的视频数据,并传送至微投影仪。系统在 FLCOS 微投影实验平台上的实际应用表明:系统传输速率可达

18.75 Mbyte/s,能够实现 16 位高彩色、640×480、30 帧/s 视频数据的实时传输。系统设计的一个主要特点是可传输现有主流媒体播放器播放画面的位图数据,从而避免了上位机对不同视频格式文件解码的复杂性。

参 考 文 献:

- [1] 梁静秋. 微显示器件的研究进展 [J]. 光机电信息,2010,27(12):21-27.
- [2] 王勇竞. 微投影技术的展望 [J]. 电视技术,2008,32(7):52-53.
- [3] 姚剑敏,勒明,宋建中. 基于 USB2.0 的实时视频图像传输 [J]. 数据采集与处理,2004,19(3):352-355.
- [4] Cypress Semiconductor Corporation. EZ-USB FX2 Manual Technical Reference [EB/OL]. [2011-06-02]. http://www.keil.com/dd/docs/datashts/cypress/fx2_trm.pdf.
- [5] 钱峰. EZ-USB FX2 单片机原理、编程及应用 [M]. 北京:北京航空航天大学出版社,2006: 133-170.
- [6] Terasic Technologies. Sdram_Control_4Port . v [CP/CD]. [2011-06-09]. <http://www.terasic.com/downloads/cd2rom/dc2/>.
- [7] Micron Technology Inc. Micron WVGA FLCOS Microdisplay Panel MT7DPWV2F [EB/OL]. [2011-06-11]. http://www.micron.com/products/flcos_microdisplays.
- [8] 宋坤,刘锐宁,马文强. Visual C++ 视频技术方案宝典 [M]. 北京:人民邮电出版社,2008: 405-420.
- [9] Samir Joshi. EZ Loader Custom USB Firmware Loader Driver. [EB/OL]. [2011-06-15]. <http://www.cypress.com/?docID=27494>.
- [10] Altera Corporation. Introduction to the Quartus II Software [EB/OL]. [2011-06-23]. http://www.altera.com.cn/literature/manual/intro_to_quartus2.pdf.

《液晶与显示》投稿指南

《液晶与显示》投稿方式为网上投稿。网上投稿便于您随时查询稿件的处理情况,方法为:登录本刊网站 <http://www.yjyxs.com>,进入“作者投稿”栏目,在线注册投稿。注册时“用户名”和“口令”由您设定,登录后按“提示”进行操作即可。稿件请用 Word 完成以便专家网上审理。

《液晶与显示》稿件发表的正常周期为 3~6 个月,缩短论文发表周期是学术论文的社会效益尽早实现的重要条件,而满足《液晶与显示》征稿简则,特别是第 3 项中(1)~(7)的要求,是稿件可以尽早编辑加工的必要条件。因此,若您希望论文能够早日发表,请您务必按“简则”写稿。

若您的稿件附有同行专家评语及单位推荐信,则您的稿件将优先发表,也欢迎您推荐 2~3 名审稿专家;同时,本刊更欢迎国家各重大科技攻关项目和基金课题产出的自主创新性文章。

《液晶与显示》稿件发表含印刷版、电子版和网络版,对版权有特殊要求者,请事先声明。