

·光电系统·

基于SOPC的实时图像处理系统设计

许 峥, 李 明, 李成金, 赵勋杰

(苏州大学物理科学与技术学院, 江苏 苏州 215006)

摘 要:介绍了基于可编程片上系统(system on programmable chip, SOPC)技术的图像处理系统的软硬件设计, 创建了CMOS图像传感器控制器IP核和VGA显示控制器IP核, 实现了图像的采集和显示, 并由Nios II软核实现了基于Sobel算子的图像边缘检测算法。采用此系统不仅可以获得良好的实时性, 还能大大简化图像处理系统的软硬件设计。

关键词:可编程片上系统(SOPC); Nios II; Sobel算子; 图像处理

中图分类号: TP391

文献标识码: A

文章编号: 1673-1255(2011)04-0004-04

Design of Real-time Image Processing System Based on SOPC

XU Zheng, LI Ming, LI Cheng-jin, ZHAO Xun-jie

(School of Physical Science and Technology, Soochow University, Suzhou 215006, China)

Abstract: Based on SOPC (system on programmable chip), the hardware and software designs of the image processing system are introduced. The controller IP cores of the CMOS image sensor and VGA display are designed to acquire and display the images respectively. The algorithm of the image edge detection based on the Sobel arithmetic operators is used by the Nios II soft core. Using this system a good real-time performance can be obtained, and the software and hardware designs of image processing system are greatly simplified.

Key words: system on programmable chip(SOPC); Nios II; Sobel arithmetic operators; image processing

SOPC(system on programmable chip, 可编程片上系统)技术是一种用于嵌入式开发的片上系统可编程技术, SOPC基于FPGA芯片, 将处理器、存储器、I/O口等模块集成在一起, 完成整个系统的主要逻辑功能, 具有设计灵活、可剪裁、可扩充、可升级及软、硬件在系统可编程的功能, 运用该技术使系统软硬件设计变得轻松, 移植性更强, 因而被广泛应用于图像处理系统的设计中^[1-4]。此设计首先创建CMOS图像传感器控制器IP核和VGA显示控制器IP核, 分别用于控制CMOS图像传感器实现图像采集以及完成图像显示的功能。然后基于SOPC技术, 在SOPC Builder平台下完成系统的硬件设计。在完成图像处理系统的硬件设计后, 在Nios II IDE

平台下开发系统软件, 由Nios II软核控制图像的采集, 并以Sobel算子边缘检测算法为例, 实现了实时图像边缘检测和显示。

1 硬件系统设计

基于SOPC技术的系统设计, 首先要完成系统的硬件设计^[5]。所设计的硬件系统框图如图1所示。图1中FPGA部分建立在FPGA芯片内, 由SOPC Builder软件完成该部分的设计, 其核心是Nios II处理器。

设计选用的FPGA芯片为Altera公司生产的EP2C35F672C6N芯片, 该芯片具有性能强功耗低的

收稿日期: 2011-06-13

基金项目: 江苏省高校自然科学基金(BK2009116)

作者简介: 许峥(1987-), 男, 江苏丹阳人, 硕士研究生, 主要从事基于FPGA的图像处理方向的研究; 导师简介: 李成金(1960-), 男, 辽宁锦州人, 教授, 研究方向图像处理技术。

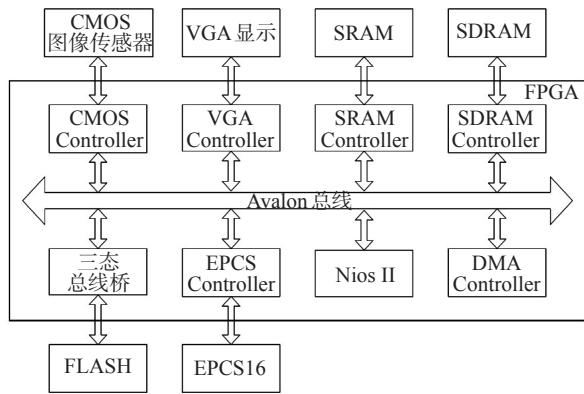


图1 图像处理系统框图

特性。设计采用的FPGA配置芯片为EPCS16,用于存储FPGA硬件配置数据,在系统上电时对FPGA进行硬件配置。FLASH存储器用于存放应用程序,SRAM作为显示缓冲区保存图像边缘检测的结果,SDRAM则用于暂存CMOS采集到的图像数据。图1中的三态总线桥、SRAM Controller和SDRAM Controller分别为上述三者与Avalon总线相连提供相应的Avalon总线接口。CMOS Controller为自定义的CMOS图像传感器控制器IP核,它在Nios II软核的控制下将CMOS图像传感器采集到的图像数据通过Avalon总线经SDRAM Controller暂存于外部SDRAM存储器中,VGA显示控制器IP核VGA Controller则按照分辨率为640×480×60 Hz的VGA显示时序输出图像分割结果,图像数据经数模转换后通过VGA接口在具有VGA接口的LCD显示器上显示输出。

1.1 CMOS Controller的设计

CMOS Controller为用户自定义的Avalon总线外设,其内部结构如图2所示。图2中,左侧的信号线与CMOS图像传感器相连;右侧信号线与Avalon总线相连。CMOS图像传感器配置模块通过时钟信号线I2C_SCLK和数据信号线I2C_SDA以I2C总线的通信方式配置CMOS图像传感器的工作方式。图像采集模块在像素时钟信号PCLK的上升沿捕获帧有效信号FVAL和行有效信号LVAL,并在FVAL和LVAL同时为高电平时采集图像数据DATA[11:0],图像数据经双时钟先入先出缓冲器DCFIFO后,由Avalon总线接口模块读出并通过Avalon总线写入SDRAM存储器中。Nios II可以通过Avalon总线接口模块访问CMOS Controller的内部控制

寄存器,通过控制寄存器控制图像采集模块开始或停止采集图像。

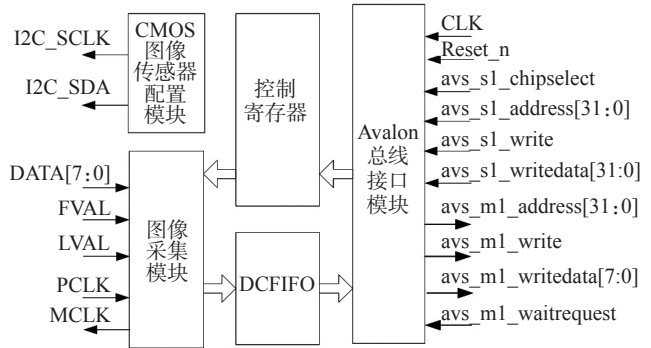


图2 CMOS Controller结构图

1.2 VGA Controller的设计

在显示图像边缘检测结果时,图像数据量大、实时性要求高,如果每个像素点的传输都由Nios II去实现,需要占用大量的CPU资源。为了将Nios II从繁重的数据传输任务中解脱出来,设计使用DMA控制器(DMA Controller)完成图像数据的传输。同时,为了实现图像数据的高速传输,VGA Controller的设计采用了流模式传输方式,其内部结构如图3所示。

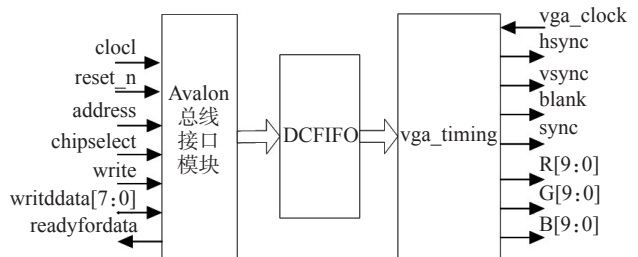


图3 VGA Controller结构图

在图3中,左侧的信号线与Avalon总线相连,右侧信号为VGA显示所需的数据信号和控制信号。Avalon总线接口模块以流模式传输方式与Avalon总线相连。由于系统时钟与VGA显示的像素时钟不同,所以采用双时钟先入先出缓冲器DCFIFO来缓冲图像数据。当DCFIFO中存储的数据达到设定的容量阈值时,readyfordata端口输出高电平以通知主端口停止对VGA Controller的写操作。反之,当DCFIFO中存储的数据小于设定的容量阈值时,readyfordata端口输出低电平以通知主端口VGA Controller已经做好了接收数据的准备。vga_timing功能模块从DCFIFO中读取图像数据并按照VGA

显示时序,输出图像数据和相应的控制信号。

2 SOPC 软件系统设计

完成 SOPC 硬件系统设计后,还需要进行 SOPC 软件系统设计,通过软件来协调各个模块统一工作实现设计要求。此系统的软件设计是在 Nios II IDE 平台下,使用 C 语言编写而成。图 4 给出了单帧 CMOS 采集图像经过图像处理并显示图像处理结果的程序流程图。

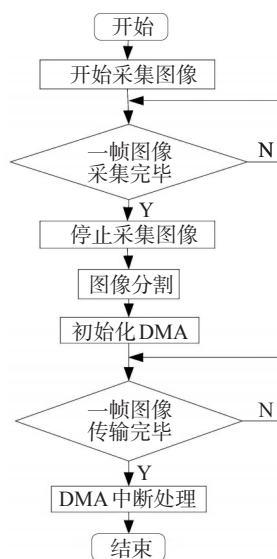


图4 图像采集、处理和显示程序流程图

在图像采集、处理和显示边缘检测结果时, Nios II 首先完成对 CMOS Controller 的初始化,由 CMOS Controller 将 CMOS 采集到的图像数据暂存于 SDRAM 存储器中。一帧图像采集完毕后, Nios II 通过复位 CMOS Controller 停止采集图像,并对 SDRAM 中的图像进行处理同时将处理结果写入 SRAM 中。图像处理结束后, Nios II 启动 DMA Controller,由 DMA Controller 将 SRAM 中的处理后的图像以流模式的传输方式送给 VGA Controller 完成图像处理结果的显示。

3 系统应用

为检验所设计的图像处理系统的性能,以 Sobel 边缘检测算法为例,应用该系统对图像进行实时的边缘检测和显示。

基于 Sobel 算法进行边缘检测,首先用 Sobel 算子计算出水平梯度和垂直梯度,然后把 2 个方向的

梯度结合起来,最后应用门限处理方法判断图像边缘。其算法流程如图 5。

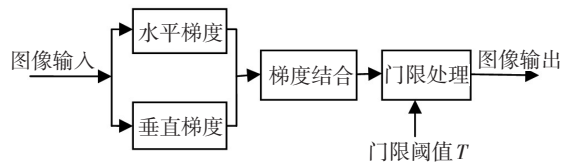


图5 基于 Sobel 算子的边缘检测算法框图

Sobel 算子对图像的每个像素以检测像素点为中心,使用如图 6 所示的卷积模板来计算水平方向和垂直方向的梯度值。在门限处理时,采用全局门限。当某像素点 (x, y) 的梯度值大于或等于设定的门限时,该点的值设为 255,反之则为 0。

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1
x 方向			y 方向		

图6 Sobel 算子卷积模板

基于 Sobel 算子的边缘检测算法的软件实现可分为以下 4 个步骤:

(1) 采用 Sobel 算子卷积模板分别与预处理后的图像 $g(x, y)$ 进行卷积,卷积结果做归一化处理,取二者中最大的结果作为边缘响应。

(2) 对边缘响应图像 $h(x, y)$ 做直方图统计,结果存入直方图数组 Count(256) 中。

(3) 对直方图数组 Count(256) 中的所有非 0 像素进行平均,得到均值 $TH0$ 。由于图像 $h(x, y)$ 中大部分非 0 像素为非边缘响应,所以低于均值 $TH0$ 的像素都可视为非边缘像素。

(4) 对边缘响应图像做阈值分割,高于阈值 T 的点赋值 255(视为真正边缘点),低于阈值 T 的点赋值 0。

图像边缘检测算法流程图如图 7 所示。

4 实验结果

在 Altera 公司的 DE2 开发板上搭配 TRDB-D5M 运行该系统对图像进行图像边缘检测,TRDB-D5M 是一个 500 万像素的 CMOS,显示输出时通过 VGA 接口外接 1 280×1 024 的 LCD 显示器,分割阈值 $T=24$ 。本系统为实时处理系统,图 8 是其中一帧图像处理前、后的结果。

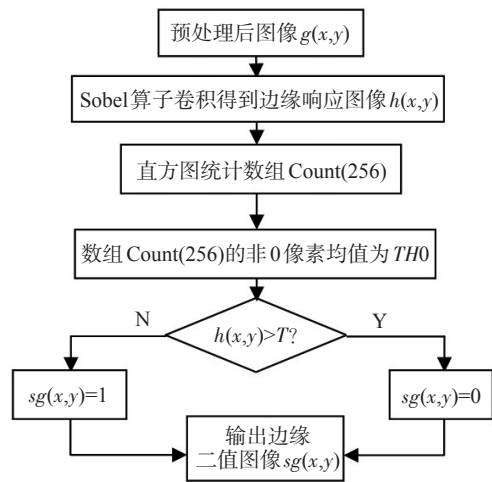


图7 基于Sobel算子的边缘检测流程图



(a) CMOS采集到的原始图像 (b) 边缘检测后得到的结果

图8 LCD显示器的显示结果

图8a为CMOS采集到的原始图像,图8b为经过该实时图像处理系统进行边缘检测后得到的结果,给出的边缘检测结果表明了该系统很好地完成了图像边缘检测任务,且系统的实时性很好。

由于该系统自定义了CMOS的IP核和VGA显示的IP核,同时因为SOPC技术的可重构性,在对图

(上接第3页)

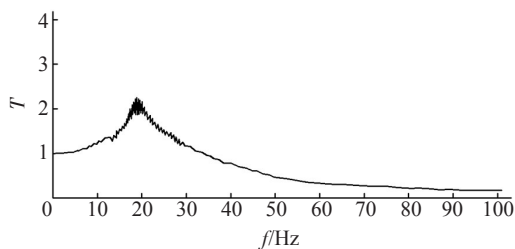


图6 传递率-频率关系曲线

递率在优势频率附近不超过2.5,且隔振频带非常宽,为30~∞ Hz。当频率大于30 Hz,放大系数小于1,并且随着频率的增加,传递率显著减小。这说明此种减振器减振效果明显,且优势频带宽,充分体现了减振的非线性及振动过程中耗能的特性。

5 结束语

上述三维减振系统能在空间任意方向动载荷

像处理算法的适当变换能获得与之相对应的图像处理结果,使用该系统的设计思路可以大大简化图像处理系统的开发工作,为其他各种图像处理系统开发提供借鉴。

5 结束语

基于SOPC技术,给出了一种在单片FPGA上实现软硬件协同完成图像采集、处理和显示的实时系统设计方法。实验结果表明,该系统具有很好的实时性和图像边缘检测效果。由于SOPC技术具有可重构的特点,可以根据图像处理算法的要求,对系统进行灵活的裁减、扩充、重新开发,大大简化图像处理系统的软硬件设计工作。

参考文献

- [1] 詹毕旺. 基于FPGA的SOPC的图像采集模块设计[J]. 微计算机信息, 2010, 26(6-2): 134-136.
- [2] 陶宝泉, 付永庆, 梁盼. 基于SOPC的视频采集及传输系统的设计[J]. 应用科技, 2010, 37(4): 31-34.
- [3] 沈永格, 李范鸣, 李争. 基于SOPC的红外图像采集系统[J]. 红外技术, 2010, 32(6): 311-314.
- [4] 邱宏, 赵馨瑶, 刘纪洲. 基于片上系统的红外直方图实时处理架构[J]. 激光与红外, 2009, 39(10): 1123-1125.
- [5] 李英兰. SOPC设计原理及应用[M]. 北京: 北京航空航天大学出版社, 2006.

作用下产生弹性动变形,大量吸收和消耗系统的振动能量,可用于大冲击振动条件下光电设备减振结构设计,实现了光电设备的线位移振动,可为动载体行进中工作的光电设备的减振设计提供参考。

参考文献

- [1] 张褒, 贾平, 黄猛. 动载体成像模糊的振动被动控制技术[J]. 光学技术, 2003, 29(3): 281-283.
- [2] 王东升, 周桐, 李健, 等. 振动、冲击环境下支架减振器刚度优化设计[J]. 航天器环境工程, 2006(2): 86-89.
- [3] 机械设计手册编委会. 机械设计手册(新版)(第五卷)[M]. 北京: 机械工业出版社, 2005.
- [4] 李德葆, 陆秋海. 工程振动试验分析[M]. 北京: 清华大学出版社, 2004.
- [5] 丁文镜. 减振理论[M]. 北京: 清华大学出版社, 1998.