

文章编号:1007-2780(2010)01-0079-06

基于 SOPC 的彩色液晶显示控制器的设计

高恭娴, 胡国兵

(南京信息职业技术学院, 江苏 南京 210036, E-mail:gao_gx@hotmail.com)

摘 要: 使用片上系统(SOC)和可编程片上系统(SOPC)的设计方法, 利用嵌入 Nios 软核的 FPGA 产品, 根据 TFT 彩色液晶显示要求, 提出彩色液晶显示器控制器软核设计的完整方案, 成功设计出数字信号综合测试仪。给出的显示点、图、汉字程序对各种需液晶显示的仪器仪表设备的设计有一定的参考作用。

关 键 词: 彩色液晶; SOPC 系统; DMA 传输; Nios 软核; Avalon 总线

中图分类号: TN27; TN334.1 **文献标识码:** A

Design of Color LCD Display Controller Based on SOPC

GAO Gong-xian, HU Guo-bing

(College of Electric Information, Nanjing College of Information Technology,
Nanjing 210036, China, E-mail:gao_gx@hotmail.com)

Abstract: According to the requirements of the TFT color LCD display system, an overall soft-core design of the controller in the LCD display combined of system on a chip (SOC) and programmable system on chip (SOPC) technique is proposed to successfully develop the digital signal tester, which is based on an embedded Nios soft-core FPGA product. The programs to display points, maps and Chinese characters have a certain reference value to the designs of a variety of LCD instrumentation equipment.

Key words: color LCD; SoPC system; direct memory access transmit; Nios soft core; Avalon bus

1 引 言

液晶显示器件具有工作电压低、功耗小、画面稳定、重量轻、屏幕调节方便等优点, 特别适合应用在便携式仪器系统中。随着 TFT-LCD 的价格日益走低, 采用彩色 TFT-LCD 取代单色 STN-LCD 作为仪器仪表的显示界面成为发展的趋势。但是, 由于便携式仪器中彩色液晶显示器既需要显示大量高速实时数据, 同时还需兼顾高速数据采集, 所以给系统设计和运行速度提出了很高的要求。在系统需要处理大量高速实时数据的情况下, 单片机由于受处理速度的限制就显得力不从

心, 而基于可编程片上系统(System on Programmable Chip, SOPC)的 FPGA 的设计是解决这个问题的最好办法。

本文使用片上系统(SOC)和 SOPC 的设计方法, 利用嵌入 Nios 软核的 FPGA 产品, 提出彩色液晶显示器控制器软核设计的完整方案, 成功设计出数字信号综合测试仪。给出的显示点、图、汉字程序对各种需液晶显示的仪器仪表设备的设计有一定的参考作用。

2 SOPC 系统设计

ALTERA 公司提出的 SOPC 解决方案, 使

收稿日期: 2009-08-14; 修订日期: 2009-09-01

作者简介: 高恭娴(1961—), 女, 江苏泰州人, 副教授, 从事电力电子产品设计工作。

得现场可编程门阵列(Field Programmable Gate Array,FPGA)在嵌入式系统领域的地位越来越重要。SOPC 的设计以 IP 核为基础,以分层次的硬件描述语言为系统功能和结构的主要描述手段,借助于以计算机为平台的 EDA 工具进行。SOPC 从整个系统的角度出发,把处理机制、模型算法、芯片结构、各层次电路、直至器件的设计精密地结合起来,将多个模块集成到一个可编程片上构成可编程的片上系统。由于 SOPC 设计能够综合并全盘考虑整个系统的各种情况,因此可以在同样的工艺技术条件下实现更高性能的系统指标,即利用 SOPC 可以成功实现软硬件的无缝结合,直接在处理器芯片内嵌入操作系统的代码模块^[1]。

Altera 的 Nois 是用户可随意配置的 32/16 位总线(用户可选)指令集和数据通道的嵌入式系统 IP 软核,采用 Avalon 总线结构通信接口,带有增强的内存、调试和软件功能(C 或汇编程序优化功能),含有 First Silicon Solutions(FS2)开发的具有 JIAG 的片内设备(OCI)内核。另外,基于 Quartus II 平台的用户可编辑的 Nois 核含有很多可配置的接口模块核,包括可配置高速缓存模块,可配置 RS232 通信口、SDRAM 控制器、标准以太网协议接口、DMA、定时器、协处理器等。在植入(配置进)FPGA 前,用户可根据设计要求,利用 Quartus II 和 SOPCbuilder,对 Nois 及外围系统进行构建,使该嵌入式系统在硬件结构、功能特点、资源占用等方面全面满足用户设计系统要求。Nois 核在同一 FPGA 中植入的数量没有限制^[2]。

3 液晶显示控制器硬件设计

本系统选择台湾晶光电公司生产的 26.4 cm(10.4 in)、640×480 分辨率、26 万色的 AM640480E 型 TFT-LCD,它有 4 个控制信号:行信号、场信号、使能信号和扫描时钟信号。信号通过差分接口存送。

液晶显示器模块采用逐行扫描,当第一个扫描时钟到来时,把第一行第一个点的 RGB 彩色信息传送给 LCD;每当发送完 640 点时,发送一个行脉冲信号;每当发送完 480 行时(也就是发送完一帧数据)时,发送一个场信号(也叫帧信号)。

扫描时钟一般采用 60 帧/s 的扫描速度;当采用 640×480 的液晶时,扫描时钟频率为 640×

480×60=18 432 kHz。在确定了点的扫描信号后,根据扫描原理就很容易得到行扫描和场扫描信号。

FPGA 芯片采用 EP1C6QC240C8 cyclone,此芯片支持在线配置 Nios 软核。系统利用 SOPC Builder 及其芯片所提供的 Avalon 总线,配置了 32 位 CPU 软核,ROM、RAM 和 FIFO 等片内资源,SDRAM、DMA 等外设,用以存储和快速传输数据。

液晶显示控制系统由 Avalon 总线接口、控制模块、DMA、FIFO 等组成。处理器通过 Avalon 总线读写寄存器从而完成对 LCD 控制器工作状态的设定与控制^[3]。其结构如图 1 所示。

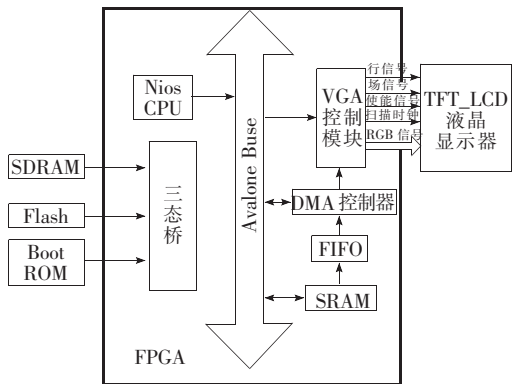


图 1 FPGA 系统结构图

Fig.1 Block diagram of the FPGA system

4 液晶显示控制器软核设计

Nios II 处理器对 LCD 模块的接口可以有两种方案实现:第一种是将 LCD 接口当作普通的外部设备 PIO 进行操作,LCD 的数据以及读、写使能,片选信号等都包括在 PIO 总线内;第二种是将 LCD 看成 SOPC Builder 中的一个 component,即外部存储器,直接做成 Avalon 总线的 Slave 设备^[4]。由于针对 SOPC 已开发了相应的模块,本文选择第二种方式。在 SOPC Builder 中打开 Component Editor,在 HDL Files 标签下添加硬件描述语言编写的文件并将其设定为顶层模块,该文件描述了组件与 Avalon 总线的接口以及组件与液晶屏的接口^[5]。在 Opencore 网站上下载 VGA 控制器将其添加到 Avalon 总线上,首先必须对 VGA 控制器进行设置:

(1)彩色数据位宽:在 Color Depth 选项区域

内选择 LCD 一个点的色彩,如果设定 18 位宽度也就是 26 万色。但是为了节省显存的存储空间,本文采用 16 位宽度,其中 R 为 5 位,B 为 6 位,G 为 5 位,颜色为 65 536 色。

(2)DMA 工作方式:第二选项区域用以选择是管道读方式或者突发读方式。

(3)FIFO 的深度:对于 LCD 显示来说,FIFO 越大越好。

(4)液晶点阵大小:本设计选择 640×480 的 TFT-LCD。

(5)单显存或者双显存:若选择单显存指针就指向一个显存块,双显存指针就同时指向两个显存块,本文选择双显存。

将设置好的 VGA _Control 及其他组件通过 SOPC Builder generate Nois CPU 软核即可完成初始化设计。

在系统设计中,主要利用 CPU 完成所有的驱动任务。CPU 利用 SOPC Builder 产生,VGA 显示的分辨率为 640×480 ,每秒扫描 60 帧,每一个扫描周期只有 50 ns 左右,如果每个点都让 Nios 软件扫描产生,在 50 ns 的时间间隔内最多只能让工作在 50 MHz 的 Nios CPU 执行两条指令。由 VGA 扫描的原理可知下一个扫描点是可以预知的,这样就可以将要送出的像素信息排成一行顺序送出,称为一个数据流(Streaming),借助于 Avalon 流模式外设的设计方法,实现一个 Avalon 流模式的 VGA 控制器。利用 DMA 控制器在流模式控制器和 SRAM 之间建立一条 DMA 传送通道,让硬件来完成像素信息的自动读取,而 Nios CPU 仅仅通过操作 SRAM 中对应的区块来完成 VGA 图像的更新。

DMA 外设模块如图 2 所示。从图中可以看

到一个 DMA 控制器具有 3 个 Avalon 总线接口: Avalon 读主端口 (Avalon Read Master Port)、Avalon 写主端口 (Avalon Write Master Port) 和读写均可的 Avalon 从端口 (Avalon Slave Port)。两个主端口都是用于 DMA 通道的,而从端口用于 DMA 控制相关寄存器的读写。

一个典型的 DMA 传输过程如下:

(1)CPU 通过控制端口 (Control Port) 配置 DMA 外设,准备启动 DMA 传输;

(2)CPU 启动 DMA 外设进行 DMA 传输,DMA 外设开始在没有 CPU 干预的情况下传送数据;

(3)DMA 读端口从读地址读取数据,然后向 FIFO 依次写入数据,DMA 写端口从 FIFO 中读取数据并向目标地址写入数据,在传输过程中可以不需要 CPU 进行干预,但 CPU 可以终止当前的 DMA 传输过程。

(4)当传输数据达到指定长度或者遇到包结束符(EOP)时,DMA 传输结束,DMA 外设将产生一个 DMA 传输结束中断。

5 软件设计

VGA 驱动的底层建立在 Avalon 总线流模式硬件上。每帧刷新采用 DMA 中断实现,数据刷新采用 SRAM 对应的区域就可实现。上层图形显示直接对驱动的图形函数调用,显示出各种图形和波形。程序设计结构如图 3 所示。程序调试仿真如图 4 所示。

下面定义了结构体 vga_frame_buffer_struct,用于定义结构体变量,对 LCD 的操作就是通过此结构体来完成的^[3]。显示软件驱动部分程序如下:

```
typedef struct{
    vga_frame_array * vga_frame0;
    //显存 0 的指针
    vga_frame_array * vga_frame1;
    //显存 1 的指针
    int width;
    //LCD 控制核支持的宽度
    int height;
    //LCD 控制核支持的高度
    int color_depth;
    //LCD 控制核支持的彩色数据宽度
```

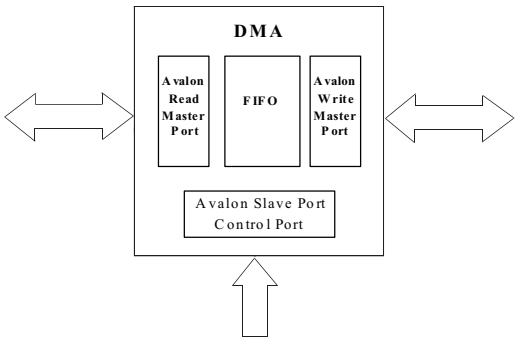


图 2 DMA 外设模块
Fig.2 Module of DMA

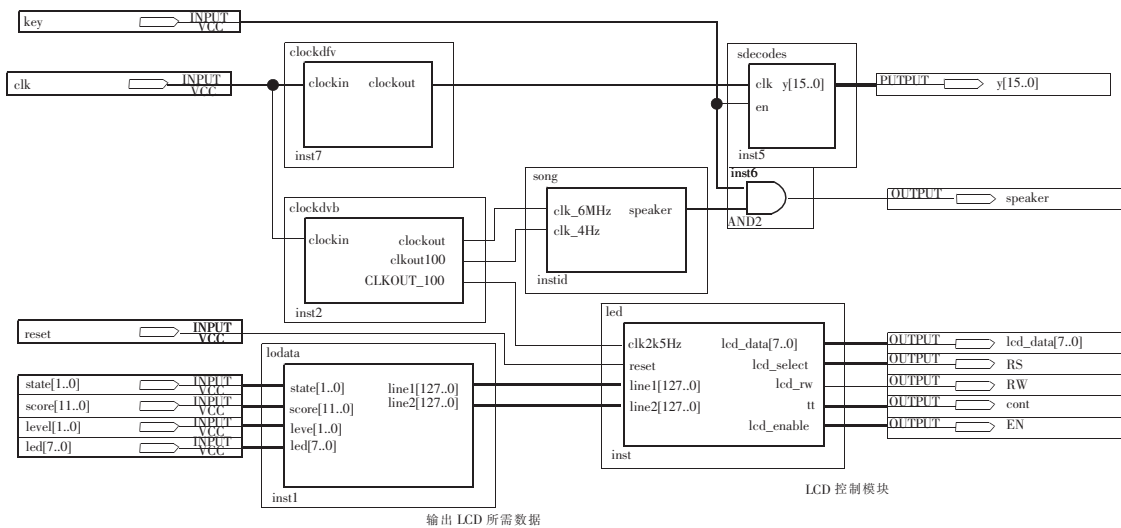


图 3 程序设计结构图

Fig. 3 Structure of FPGA program

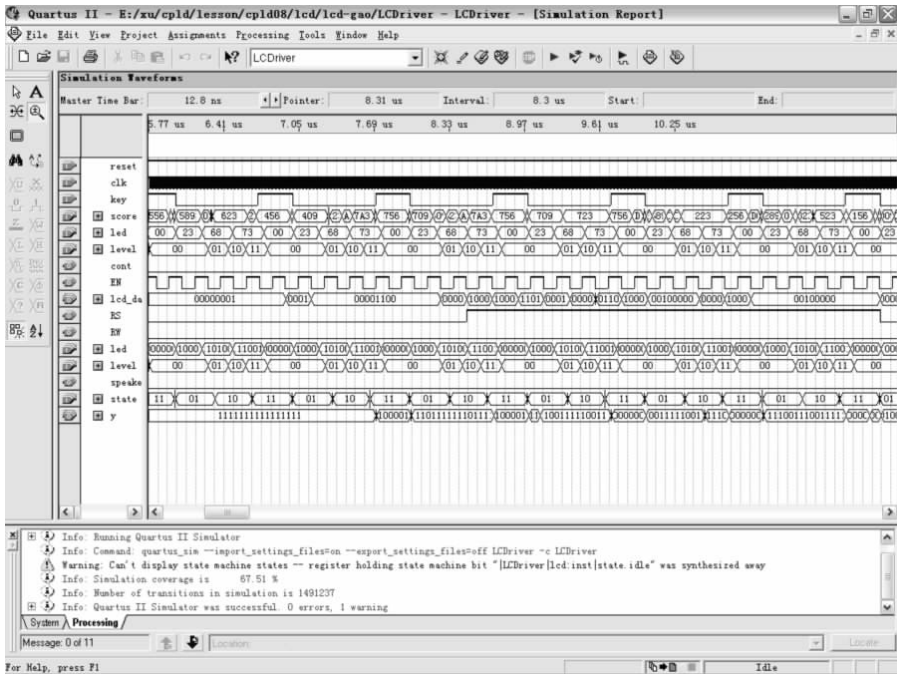


图 4 计算机仿真波形图

Fig. 4 Computer emulation oscillograph

```
int byte_per_pixel;
//每个点彩色数据所占的字节数
int byte_per_frame;
//LCD 核支持的缓存数

int vga_controller_base;
//LCD 控制寄存器的基地址
} vga_frame_buffer_struct;
打点函数定义为:
```

```
void vga_print_pixel(int row,int line,char r,char g,char b,
vga_frame_buffer_struct * vga_frame_buffer, int buffer_number)
{
int addr1,addr2,addr3;
```

```
addr1 = row * vga_frame_buffer->width * 3 + line * 3;           //计算出该点的 R 的位置
addr2 = addr1 + 1;                                               //计算出该点的 G 的位置
addr3 = addr2 + 2;                                               //计算出该点的 B 的位置
if( buffer_number == 0)
{
    IOWR_BDIRECT(vga_frame_buffer->vga_frame0, addr1, r);
    IOWR_BDIRECT(vga_frame_buffer->vga_frame0, addr2, g);
    IOWR_BDIRECT(vga_frame_buffer->vga_frame0, addr3, b);
}
else if( buffer_number == 1)
{
    IOWR_BDIRECT(vga_frame_buffer->vga_frame1, addr1, r);
    IOWR_BDIRECT(vga_frame_buffer->vga_frame1, addr2, g);
    IOWR_BDIRECT(vga_frame_buffer->vga_frame1, addr3, b);
}
}
```

由这个打点函数很容易设计出画图函数,下面是显示一幅画面的代码:

```
Void LCD_print_picture( vga_frame_buffer_struct * vga_frame_buffer, char * Picture_data)
{
    int row, line;
    int addr;
    for( row = 0; row < vga_frame_buffer->height; row++)
    {
        for( line = 0; line < vga_frame_buffer->width; line++)
        {
            addr = row * vga_frame_buffer->width * 3 + line * 3;
            vga_print_pixel( row, line, Picture_data(addr), Picture_data(addr+1),
            Picture_data(addr+2), vga_frame_buffer, 0);
        }
    }
}
```

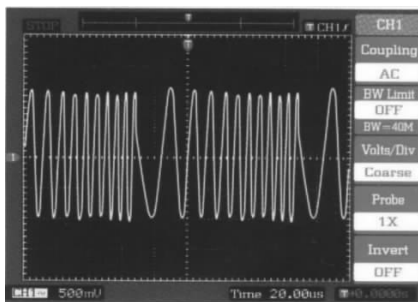


图 4 数字信号综合测试仪 LCD 显示

Fig. 4 LCD display in digital signal integrated test instrument

本设计完成后成功地应用在数字信号综合测试仪的显示中,显示画面如图 4 所示。

6 结 论

使用 SOC 和 SOPC 的设计方法,利用嵌入 Nios 软核的 FPGA 产品,根据 TFT 彩色液晶显示要求,提出彩色液晶显示器控制器软核设计的完整方案,通过调试,在液晶屏幕上实现了显示各种颜色的汉字、英文、数字及图形等功能。本设计已应用于所设计的数字信号综合测试仪上,达到了显示稳定、色彩丰富、人机界面友好的显示效果。

参 考 文 献:

[1] 潘松,黄伟业,曾毓. SOPC 技术实用教程 [M]. 北京:清华大学出版社,2005.

[2] 王晓迪,张景秀. SOPC 系统设计与实践 [M]. 北京:北京航空航天大学出版社,2008.

[3] 杨亮亮,凌朝东,李国刚,等. 基于 NiosII 的 TFT- LCD 控制器设计与实现 [J]. 现代显示,2008,(7):28-32.

[4] 侯长宏,袁慧梅. 基于 SOPC 的 Nios II 与液晶模块接口及程序设计 [J]. 液晶与显示, 2008, 23(6):307-310.

[5] 郭书军,候文杰,贾克斌. Nios II 液晶屏接口组件设计 [J]. 液晶与显示,2007,22(2): 213-216.



包括 SI 辅助单位在内的具有专门名称的 SI 导出单位

量的名称	名称	符号	SI 导出单位
			用 SI 基本单位和 SI 导出单位表示
[平面]角	弧度	rad	1 rad=1 m/m=1
立体角	球面度	sr	1 sr=1 m ² /m ² =1
频率	赫[兹]	Hz	1 Hz=1 S ⁻¹
力	牛[顿]	N	1 N=1 kg·m/S ²
压力,压强,应力	帕[斯卡]	Pa	1 Pa=1 N/m ²
能[量],功,热量	焦[耳]	J	1 J=1 N·m
功率,幅[射能]通量	瓦[特]	W	1 W=1 J/S
电荷[量]	库[仑]	C	1 C=1 A·S
电压,电动势,电位	伏[特]	V	1 V=1 W/A
电容	法[拉]	F	1 F=1 C/V
电阻	欧[姆]	Ω	1 Ω=1 V/A
电导	西[门子]	S	1 S=1 Ω ⁻¹
磁通[量]	韦[伯]	Wb	1 Wb=1 V·S
磁通[量]密度,磁感应强度	特[斯拉]	T	1 T=1 Wb/m ²
电感	亨[利]	H	1 H=1 Wb/A
摄氏温度	摄氏度	℃	1 ℃=1 K
[光]通量	流[明]	lm	1 lm=1 cd·Sr
光照度	勒[克斯]	lx	1 lx=1 lm/m ²
[放射性]活度	贝可[勒尔]	Bq	1 Bq=1 S ⁻¹
吸收剂量	戈[瑞]	Gy	1 Gy=1 J/kg
剂量当量	希[沃特]	Sv	1 Sv=1 J/kg