

基于SOPC的视频数据采集系统的设计

□ 古应春 郑郁正 (成都信息工程学院 电子工程学院, 四川 成都 610225)

摘 要: 本文研究了基于可编程片上系统(SOPC)技术的实时视频数据采集系统的实现方案。使用ALTERA公司的Nios II软核处理器作为总控制器, 采用CCD 传感器和ADV7181B视频解码芯片进行图像数据的采集。系统采用软硬结合的设计方法, 实现了硬件平台的构建和软件的调试。

关键词: 数据采集 SOPC技术 CCD

1 引言

传统的视频处理系统为了满足实时性和灵活的接口需要, 多采用FPGA+DSP的硬件架构。FPGA可编程逻辑器件功能强大, 开发投资小, 周期较短, 可以反复编程修改。随着FPGA的应用越来越广泛, 大容量高速的FPGA以及SOPC技术的出现, 为图像采集和处理提供了一种新的解决方案, 使用FPGA来构建基于可编程系统(System On a Programmable Chip, 简称SOPC)的图像处理系统, 充分利用SOPC Builder中自带的各种IP核控制器, 减少底层硬件的开发^[1-2], 已成为一种趋势。

2 视频数据格式

CCD (Charge coupled Device, 电荷耦合原件) 是一种半导体器件, 能够将光学影像转换为数字信号。CCD摄像头与传统的摄像头相比较而言, 具有不失真, 成像效果好, 噪点少, 在强光下不易曝光过度等优点而广泛应用于各种环境下的监控和视频图像采集系统。CCD摄像头分为PAL格式和NTSC格式^[3]。在本设计中采用PAL (720*576) 制式的摄像头, 采用隔行扫描方式进行扫描图像。CCD摄像头输出的信号为模拟信号, 为了实现数字化处理, 必须进行A/D转换, 因此采用

ADV7181B芯片将模拟信号转换为数字信号, 但是此时的视频信号包括同步信号和消隐信号, 还无法处理, 为了真正实现可处理的有效视频数据, 要对ITU-RBT.656视频流进行解码。

2.1 ITU-RBT.656视频数据传输格式

ITU-RBT.656视频标准接口PAL制式, 每场包含有效视频数据、水平消隐、垂直消隐三个部分, PAL制式一场的数据行格式如表1所示^[4-5]:

表1 一场数据行格式

Blanking	Line1--Line23
ODD/active video	Line24--Line311
Blanking	Line312--Line336
EVEN/active video	Line337--Line624
Blanking	Line625

从表1中可以看出, 有效视频数据(active video), 分为奇数场和偶数场, 均由288行组成, 所以有效数据行为576行, Blanking为水平消隐区, 通常由80H/10H来填充, 消隐行为49行, 所以每场总共625行。而每一行数据如表2所示。

表2 一行数据结构

EAV Code				Blanking Video					SAV Code				Active Video				
FF	00	00	EAV	80	10	80	10	...	FF	00	00	SAV	Cb	Y	Cr	Y	...
4Bytes				280Bytes					4Bytes				1440Bytes				
1728 Bytes																	

从表2中可以看出,每行数据有1728 Bytes,有效视频数据 (Active Video) 有1440个字节,其中720个字节为Y分量,360个字节为Cb分量,360个字节为Cr分量,有效视频数据开始(SAV)和有效视频数据结束 (EAV) 是两个定时基准信号^[4],每个信号由4个字节构成,其格式如下: FF 00 00 XY, 前三个是固定头缀,XY 为控制

表3 定时基准信号SAV和EAV定义

数据比特号	第一字 (FF)	第二字 (00)	第三字 (00)	第四字 (XY)
9 (MSB)	1	0	0	1
8	1	0	0	F
7	1	0	0	V
6	1	0	0	H
5	1	0	0	P3
4	1	0	0	P2
3	1	0	0	P1
2	1	0	0	P0
1	1	0	0	0
0	1	0	0	0

字,其含义如表3。

表3中给出的是10比特接口的建议值,但我们用8比特的接口,所以D0和D1未作规定。

最高位 (MSB): 始终保持为1;

F: 场同步信号,表示该行数据处于奇数场还是偶数场, F=0或者1;

V: 垂直同步信号,表示处于场消隐区间还是正程区间 (有效数据行), V=0是表示处于有效数据行, V=1 表示处于场消隐区间;

H: 水平同步信号,表示是“SAV”还是“EAV”, H=0 表示处于有效数据开始,即SAV, H=1表示处于有效数据结束,即EAV;

P0, P1, P2, P3 是保护比特位。

3 CCD 视频信号数据采集系统设计

3.1 硬件设计框架结构图

整体设计的一个框架结构图如图1所示。

本系统主要包括CCD摄像头, 视频解码芯片 ADV7181B^[3], I2C总线配置视频解码芯片, 视频采集控制电路, FIFO, DMA控制器, SDRAM, NiosII处理器, 由 SOPC Builder生成的IP核, 发送到PC上位机。

3.2数据采集体IP核的设计

将图像数据采集器制作成IP核, 挂载到 SOPC Builder^[4] 生成一个NiosII系统。此硬件电路包括了数据采集器和FIFO两个单元, 而NiosII处理器通过AVALON总线连接外设进行读写操作, 所以需要配置硬件电路中的读写等AVALON总线信号, 实现总线控制外设的功能, 也就实现了NiosII 处理器通过AVALON总线对硬件电路的控制。

在本系统中需要的IP核有: NiosII 处理器, onchip_RAM, jtag_uart, uart, sdrum, dma,DATAEXT。用 SOPC Builder构造的系统如图2所示。

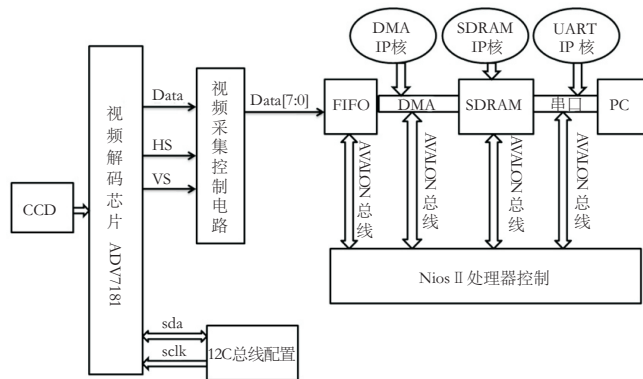


图1 设计框架图

Use	Connections	Module Name	Description	Clock	Base	End
<input checked="" type="checkbox"/>		MCU	Nios II Processor	clk_0		
		instruction_master	Avalon Memory Mapped Master			
		data_master	Avalon Memory Mapped Master			
		jtag_debug_module	Avalon Memory Mapped Slave		IRQ 0	IRQ 31
<input checked="" type="checkbox"/>		onchip_RAM	On-Chip Memory (RAM or ROM)	clk_0	0x01020800	0x01020fff
		s1	Avalon Memory Mapped Slave			
<input checked="" type="checkbox"/>		dma	DMA Controller	clk_0	0x01010000	0x01019fff
		control_port_slave	Avalon Memory Mapped Slave			
		read_master	Avalon Memory Mapped Master		0x01021000	0x0102101f
		write_master	Avalon Memory Mapped Master			
<input checked="" type="checkbox"/>		jtag_uart	JTAG UART	clk_0	0x01021040	0x01021047
		avalon_jtag_slave	Avalon Memory Mapped Slave			
<input checked="" type="checkbox"/>		uart	UART (RS-232 Serial Port)	clk_0	0x01021020	0x0102103f
		s1	Avalon Memory Mapped Slave			
<input checked="" type="checkbox"/>		sdrum	SDRAM Controller	clk_0	0x00800000	0x00ffff
		s1	Avalon Memory Mapped Slave			
<input checked="" type="checkbox"/>		DATAEXT	extdata	clk_0	0x01021048	0x01021049
		avalon_slave_0	Avalon Memory Mapped Slave			

图2 SOPC Builder构造的系统图 3.3 IDE软件驱动程序设计

```

00000000h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000010h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000020h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000030h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000040h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000050h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000060h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000070h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000080h: 80 10 80 10 80 10 80 10 80 10 80 10 80 10 80 10 ; e.e.e.e.e.e.e.e.
00000090h: 80 10 80 10 80 10 80 10 80 10 FF 00 00 EC 80 0D ; e.e.e.e.e. . . . .
000000a0h: 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D ; e.e.e.e.e.e.e.e.
000000b0h: 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D ; e.e.e.e.e.e.e.e.
000000c0h: 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D ; e.e.e.e.e.e.e.e.
000000d0h: 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D ; e.e.e.e.e.e.e.e.
000000e0h: 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D ; e.e.e.e.e.e.e.e.
000000f0h: 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D 80 0D ; e.e.e.e.e.e.e.e.
00000100h: 80 0D 12 80 13 80 13 80 12 80 12 80 13 80 13 80 ; e.e.e.e.e.e.e.e.
00000110h: 13 80 12 80 13 80 13 80 13 80 13 80 12 80 13 80 ; .e.e.e.e.e.e.e.e.
00000120h: 13 80 13 80 13 80 13 80 13 80 13 80 13 80 13 80 ; .e.e.e.e.e.e.e.e.
00000130h: 13 80 13 80 13 80 13 80 13 80 13 80 13 80 13 80 ; .e.e.e.e.e.e.e.e.
00000140h: 13 80 13 80 12 80 13 80 13 80 13 80 12 80 12 80 ; .e.e.e.e.e.e.e.e.
00000150h: 13 80 13 80 12 80 12 80 12 80 13 80 13 80 13 80 ; .e.e.e.e.e.e.e.e.
00000160h: 12 80 13 80 13 80 12 80 12 80 12 80 12 80 13 80 ; .e.e.e.e.e.e.e.e.
00000170h: 12 80 12 80 12 80 12 80 12 80 12 80 12 80 12 80 ; .e.e.e.e.e.e.e.e.
00000180h: 13 80 12 80 13 80 12 80 13 80 13 80 13 80 12 80 ; .e.e.e.e.e.e.e.e.
00000190h: 12 80 13 80 13 80 12 80 12 80 12 80 12 80 13 80 ; .e.e.e.e.e.e.e.e.
000001a0h: 13 80 13 80 13 80 13 80 13 80 13 80 12 80 13 80 ; .e.e.e.e.e.e.e.e.
000001b0h: 12 80 11 80 12 80 11 80 11 80 11 80 12 80 11 80 ; .e.e.e.e.e.e.e.e.
000001c0h: 12 80 12 80 12 80 12 80 12 80 12 80 12 80 12 80 ; .e.e.e.e.e.e.e.e.
000001d0h: 12 80 12 80 12 80 12 80 13 80 12 80 13 80 13 80 ; .e.e.e.e.e.e.e.e.
000001e0h: 13 80 13 80 13 80 13 80 13 80 13 80 13 80 13 80 ; .e.e.e.e.e.e.e.e.

```

图3 十六进制显示

在IDE软件里对NiosII处理器进行驱动程序设计。驱动程序主要通过NiosII对DMA的控制，SDRAM的读写和对UART的控制。包含的头文件如下^[9]：

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <altera_avalon_uart_regs.h>
#include <system.h>
#include <alt_types.h>

```

DMA有三种传输方式：存储器到存储器、硬件（固定地址）到存储器、存储器到硬件（固定地址）。本系统采用硬件（固定地址）到存储器，即是外设到存储器。只需要打开接收通道，而且源地址是固定的，目标地址是自增的。

```

rx=alt_dma_rxchan_open("/dev/dma");
alt_dma_rxchan_ioctl(rx,ALT_DMA_RX_ONLY_ON,(void*)source_addr);
dma_res = alt_dma_rxchan_prepare(rx, rx_buf, 32,

```

```

dma_done, NULL);

```

PC机采用串口接收数据，由于UART采用的是字符输出模式，转成十六进制如图3所示。

4 图像数据采集与还原

PC机接收下位机传来的数据，还原成CCD采集图像，该图像的正确还原验证了采集的数据的正确性和完整性。还原后的图像如图4。

5 结论

本文提出了一套基于SOPC的CCD视频数据采集系统，以DE2实验开发板为平台结合SOPC技术进行设计，采用硬件和软件结合，开发成了一套完整的图像采集系统。并通过上位机接收下位机传来数据，还原成原始图像，来验证数据的正确性，大大提高了CCD数据采集的灵活性。

参考文献

- [1] 李兰英等. Nios II 嵌入式软核SOPC设计原理及应用[M]. 北京: 北京航空航天大学出版社, 2006.
- [2] Analog Device Multiformat SDTV Video Decoder ADV718IB. 2004.
- [3] 祝长峰, 肖铁军. 基于FPGA的视频图像采集系统的设计[J]. 计算机工程与设计, 2008(29).
- [4] 耿洁, 陈家福, 徐宁. ITU-RBT. 656视频解码译码的FPGA实现[J]. 仪器仪表用户, 2007(02).
- [5] 杨军, 李彤. 基于FPGA的SOPC实践教程[M]. 北京: 科学出版社, 2007. 75-80.

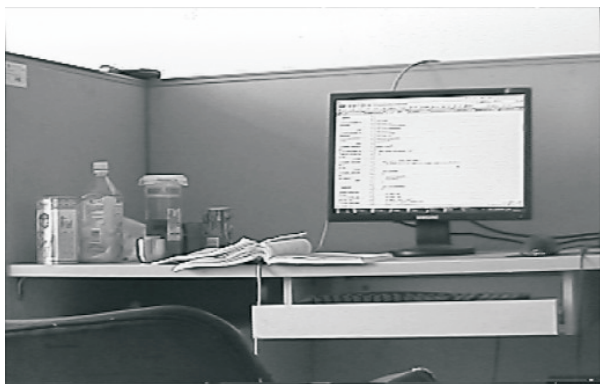


图4 采集数据还原图像