第一图书网, tushu007.com

<<爱上FPGA开发>>

图书基本信息

书名：<<爱上FPGA开发>>

13位ISBN编号：9787512405424

10位ISBN编号：7512405421

出版时间：2011-10

出版时间：北京航空航天大学出版社

作者：吴厚航

页数：315

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：http://www.tushu007.com

内容概要

《爱上FPCA开发——特权和你一起学NIOS II》结合一款基于Altera公司Cyclone II系列FPGA的开发板，从一些嵌入式开发的基本术语和概念人手，到手把手第一个工程的构建；再从一个稳定的SOPC平台设计，到NIOS II软件编程的入门；最后软硬件结合，像模像样地搭建了一些有实用价值的IT程。
本书内容可谓由浅人深，为在其他嵌入式平台上已有一定开发基础的NIOS II初学者量身打造。
字里行间，不仅透露出一个年轻工程师对技术的执着和认真，而且很多诙谐幽默的文字和真实的感悟伴随着知识也传递给读者，相信一定能够带给读者更多耳目一新的感觉。
本书配套DVD光盘，内含程序源码和20课时视频教程，方便读者学习。

《爱上FPCA开发——特权和你一起学NIOS II》的主要读者对象为电子、计算机、控制及信息等相关专业的在校学生，从事FPGA开发设计的电子工程师以及所有电子设计制作的爱好者们。

书籍目录

第1章 海阔天空聊概念
1.1 CPU之软核与硬核
1.2 SOPC是什么
1.3 NIOS II的优势
第2章 开发流程
2.1 流程
2.2 执行流程的必要性
2.3 SOPC开发的流程
第3章 流程实践案例——手把手第一个工程
3.1 硬件平台
3.2 软件平台
3.3 手把手硬件工程
3.3.1 新建Quartus II工程
3.3.2 SOPC Builder配置
3.3.3 例化NIOS II工程
3.3.4 分配引脚与编译下载
3.4 手把手软件工程
3.4.1 新建软件模板工程
3.4.2 设置软件编译属性
3.4.3 软件编译和下载
第4章 实战演练之存储控制器
4.1 嵌入式存储系统
4.2 基于FPGA的嵌入式存储解决方案
4.2.1 片内存储器
4.2.2 外部SRAM
4.2.3 Flash
4.2.4 SDRAM
4.3 第二个系统准备工作
4.4 SDRAM控制器组件添加与配置
4.5 EPCS控制器组件添加与配置
4.6 PLL组件添加与配置
4.7 其他SOPC Builder配置
4.8 编辑顶层文件与引脚分配
4.9 软件工程调试与下载
第5章 实战演练之时序收敛
5.1 时序设计四部曲
5.2 一部曲——时序分析
5.2.1 系统内部时钟时序分析
5.2.2 SDRAM接口时序分析
5.3 二部曲——时序约束
5.3.1 全局时钟约束
5.3.2 I／O接口约束之virtual clock生成与约束
5.3.3 I／O接口约束之理论推导
5.3.4 I／O接口约束之output delay约束
5.3.5 I／O接口约束之input delay约束

5.3.6 SDRAM时钟频率与相移更改
5.4 三部曲——时序报告
5.4.1 寻找PLL相移值
5.4.2 查看关键路径
5.5 四部曲——时序收敛
5.6 时序最优化
第6章 实战演练之玩转NIOS II
6.1 第三个SOPC系统硬件架构
6.1.1 添加新组件
6.1.2 例化新系统与引脚分配
6.1.3 时序约束与收敛
6.2 熟悉NIOS EDS软件开发平台
6.2.1 加载和关闭工程
6.2.2 新建工程
6.3 软件例程1——蜂鸣器实验
6.4 软件例程2——流水灯实验
6.5 软件例程3——数码管定时器实验
6.6 软件例程4——串口收发实验
6.7 软件例程5——看门狗定时器实验
6.8 软件例程6——按键中断实验
6.9 软件例程7 SD卡SPI通信实验
第7章 实战演练之自定义外设组件
7.1 总线
7.1.1 Avalon—MM总线
7.1.2 Avalon—ST总线
7.2 Avalon组件集成之TLC549
7.2.1 准备工作
7.2.2 模块源码设计
7.2.3 组件封装
7.2.4 集成新组件到系统中
7.2.5 工程例化与编译
7.2.6 软件调试
7.3 Avalon组件集成之DAC5571
7.3.1 模块源码设计
7.3.2 组件封装
7.3.3 集成新组件到系统中
7.3.4 工程例化与编译
7.3.5 软件调试
第8章 实战演练之USB通信
8.1 硬件系统架构
8.1.1 CH376芯片概述
8.1.2 准备工作
8.1.3 集成组件
8.1.4 工程例化与引脚分配
8.1.5 编译与时序收敛
8.2 软件编程
8.2.1 数据／指令读写

8.2.2 指令功能描述
8.2.3 芯片寄存器读／写测试
8.2.4 作为USB从机与PC连调
8.2.5 U盘扇区读／写操作
8.2.6 基于U盘的TXT文本创建
8.2.7 基于U盘的A／D采集数据存储
8.3 改进的CH376并口控制方式
第9章 实战演练之显示控制器DIY
9.1 液晶驱动组件设计
9.1.1 LCD显示驱动模块
9.1.2 自定义外设模块
9.2 SOPC系统硬件架构
9.2.1 准备工作
9.2.2 集成组件
9.2.3 工程例化与引脚分配
9.2.4 编译与时序收敛
9.3 图片取模配置
9.4 软件程序
9.5 串口下发板级测试
第10章 网络杂文
10.1 设计资源最大化
10.1.1 何谓设计资源
10.1.2 设计资源就在您身边
10.1.3 设计资源的案例
10.1.4 设计资源，无处不在
10.1.5 总结——积攒设计资源
10.1.6 设计思想，来源于生活
10.1.7 结束语
10.2 2010——成长，在路上
10.3 好书从比喻开始
参考文献

章节摘录

版权页：片内存储器的最佳应用场合包括作为常见的缓存、点到点的缓存、查找表以及FIFO等。
缓存：由于其具有低反应延时，片内存储器在微处理器中作为缓存表现良好。
NIOSII处理器使用片内存储器作为引导和数据缓存。
因为缓存本身都相对较小，所以片内存储器有限的容量作为缓存通常不是一个问题。
点到点的缓存：低延时的存取也使得片内存储器适用于作为器件间的缓存，即点到点的缓存。
它是指处于正常的寻址空间，但与微处理器有专用接口的存储器。
这些存储器主要用来实现缓存存储器的高速、低反应延时特性。
查找表：针对某些软件编程功能，尤其是算法上的功能。
与在软件中进行计算相比，使用查找表储存所有可能的功能结果通常是最快的方法。
片内存储器在这方面表现良好，前提是片内存储器的可用容量能够容纳可能的功能结果。
FIFO：嵌入式系统经常需要管理从一个模块到另一个模块的数据流。
FIF0可以在以不同的高速运行着的模块间作为数据缓冲存储器。
根据应用程序所需FIFO的大小，片内存储器可以作为高速和便利的FIFO存储。
但片内存储器不适用于需要大容量存储的应用中。
因为片内存储器容量相对受限，所以应避免使用其储存大量的数据。
但是，有些工作可以更好地利用片内存储器完成。
如果应用程序使用多个小块数据。
并且不是所有的数据块都适合使用片内存储器，那么设计者应当仔细考虑某些应用可以使用片内存储器。
如果用户的目标是系统的高速性能，那么可以将最经常存取的数据放在片内存储器中。
在系统中使用片内存储器需要遵从下面一些规则：①设置片内存储器的数据带宽与主控系统的数据带宽相匹配。
例如，如果要连接片内存储器到NIOSII处理器的数据控制器上，需要设置片内存储器的数据带宽为32bit，与NIOSII处理器的数据控制器的数据带宽相同；否则存取的延时将会大于一个时钟周期，这是因为系统相互连接需要进行带宽转换。
当然了，如果系统中允许这种存取延时（即多次存取以完成一次总线带宽访问），那么不匹配带宽也是没有问题的。
②如果片内存储器元件连接了多个主控制器，应考虑使用具有双通道的片内存储器。
双通道特性使得在两个主控制器连接到同一个片内存储器时无需设置判断逻辑。
另外，双通道存储器允许在两个接口同时进行存取，这样在两个主控制器连接到同一个片内存储器时，可以极大地提升效率与性能。
然而，如果没有在主控制器间进行协调，那么两个RAM的从接口同时执行写入操作将会导致数据冲突。

第一图书网, tushu007.com

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:http://www.tushu007.com