

<<基于NiosII内核的FPGA电路系统>>

图书基本信息

书名：<<基于NiosII内核的FPGA电路系统设计>>

13位ISBN编号：9787121106477

10位ISBN编号：7121106477

出版时间：2010-4

出版时间：电子工业

作者：赫建国//倪德克//郑燕

页数：273

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>

## <<基于NiosII内核的FPGA电路系统>>

### 前言

2004年春天,作为指导教师,作者首次参加了由Altera公司组织的EDA / SOPC电子设计竞赛。Altera公司提出的数字系统SOPC ( System On Programmable Chip ) 解决方案深深地吸引了我。

在此之前,作者虽然从事数字系统的教学和开发设计工作,并对用来实现数字系统的微控制器和可编程逻辑器件的使用也已多年,但是一直把它们看做是两种各具特点的目标器件。

如果需要对信号进行较复杂的处理,首选的目标器件为微控制器;如果需要进行高速信号传送,首选的目标器件为可编程逻辑器件;如果希望同时获得复杂的信号处理和高速的信号传送,那么将联合使用微控制器和可编程逻辑器件。

Altera公司提出的数字系统SOPC解决方案能够在可编程逻辑器件中形成一个用户可配置的软核处理器——Nios 软核处理器,这样就在一个芯片上同时实现了信号的复杂处理和信号的高速传送。

利用数字系统SOPC解决方案实现应用系统设计需要掌握Quartus 可编程逻辑器件开发软件和Nios 集成开发环境。

利用开发软件Quartus ,使用者能够创建以FPGA芯片为目标器件的Nios 软核处理器系统及其他外围模块;利用Nios 集成开发环境,使用者能够完成Nios 软核处理器系统的c语言设计文件的编辑、编译,并下载设计到Nios 软核处理器系统中进行调试和运行。

掌握一种技术需要大量的理论和实践学习,同时也需要一个好的学习方法,如果有一个好的学习指导是可以获得事半功倍的效果的。

通过这些年来对参加Altera公司的EDA / SOPC电子设计竞赛和全国大学生电子设计竞赛的学生的赛前训练的总结,作者编写了本书。

本书在撰写内容上力求实用,并尽量做到理论和实践相结合;在内容编排上,尽量做到由易到难。

对书中列举的基础实验和应用设计,作者努力做到不仅描述如何进行,而且分析这样做的原因,使读者能够比较容易地举一反三。

本书的内容包括3个部分。

第一部分包括第1-4章,这部分描述的是可编程逻辑器件开发的基础知识。

如果读者具有使用可编程逻辑器件的经历,这部分可以跳过去,直接从第二部分开始。

第1章介绍数字集成电路的基础知识及Nios 软核处理器的特点。

第2章描述FPGA器件的基础知识。

包括FPGA器件和标准逻辑器件在功能上的对应关系, FPGA器件的内部结构, 以及Altera公司生产的FPGA器件的性能介绍。

第3章介绍可编程逻辑器件开发软件Quartus 的使用。

内容包括设计项目的创建,设计的输入、编译、模拟, 以及向目标器件的下载。

## <<基于NiosII内核的FPGA电路系统>>

### 内容概要

本书介绍了一种能够在FPGA芯片中同时获得数据传送速度快和数据处理能力强两个优点的设计方案——在片可编程系统(System On Programmable Chip, SOPC)解决方案。该方案通过在FPGA芯片中置入一个软核处理器系统(Nios 软核处理器系统)来增强对信号的处理能力。

本书系统地描述了Nios 软核处理器系统的开发知识。内容包括Altera公司FPGA芯片的介绍、可编程逻辑器件开发软件Quartus 的使用、硬件描述语言VHDL的简介、Nios 软核处理器系统创建工具SOPCBuilder和Nios 集成开发环境(Nios IDE)的使用。书中还包括了大量的基础实验和应用系统的设计实例，能够帮助读者更快、更容易地掌握及应用这门技术。

本书适合从事Altera公司FPGA芯片开发设计的研究生和本科高年级学生使用，也适合从事该方面工作的工程师使用。

## &lt;&lt;基于NiosII内核的FPGA电路系统&gt;&gt;

## 书籍目录

第1章 引言 1.1 数字集成电路的分类 1.1.1 标准逻辑器件 1.1.2 微处理器 1.1.3 可编程逻辑器件 1.2 Nios 软核处理器 1.2.1 Nios 软核处理器系统简介 1.2.2 可配置软核处理器的优点第2章 现场可编程门阵列器件 2.1 可编程逻辑器件概述 2.2 可编程逻辑器件的发展历程 2.2.1 简单PLD的基本结构 2.2.2 FPGA的基本结构 2.3 Altera公司Cyclone 器件的工作原理 2.4 Cyclone 系列器件的主要技术指标 2.5 小结第3章 Quartus 开发软件的使用 3.1 简介 3.2 创建工程 3.3 设计输入 3.3.1 建立文本设计文件 3.3.2 建立图形设计文件 3.3.3 层次化设计 3.4 设计的编译 3.5 设计的仿真验证 3.5.1 创建仿真波形文件 3.5.2 设计仿真 3.6 引脚分配 3.7 器件配置 3.8 小结第4章 VHDL语言基础 4.1 VHDL的历史 4.2 VHDL的程序结构 4.2.1 VHDL程序的基本结构 4.2.2 实体 4.2.3 结构体 4.2.4 包集 4.2.5 库 4.3 VHDL的语言元素 4.3.1 标识符 4.3.2 对象类别与定义 4.3.3 数据类型 4.3.4 运算符 4.4 并行语句 4.4.1 并行信号赋值语句 4.4.2 进程语句 4.5 顺序语句 4.5.1 顺序信号赋值语句 4.5.2 条件(IF)语句 4.5.3 选择(CASE)语句 4.5.4 循环(LOOP)语句 4.5.5 空操作(NULL)语句 4.6 小结第5章 Nios 软核处理器系统的开发过程 5.1 概述 5.2 配置Nios 软核处理器系统 5.2.1 创建Nios 软核处理器系统 5.2.2 配置Nios 软核处理器系统 5.3 产生Nios 软核处理器系统 5.3.1 产生Nios 软核处理器系统模块 5.3.2 Nios 软核处理器系统的产生 5.4 创建Nios IDE环境下的应用工程 5.4.1 Nios IDE工程创建 5.4.2 C语言源文件的编辑 5.4.3 C语言源文件的编译 5.5 C语言源程序的调试 5.5.1 在目标电路板上运行程序 5.5.2 在目标电路板上调试程序 5.5.3 配置目标FPGA器件 5.6 小结第6章 Nios 软核处理器——程序运行时间的测量 6.1 Nios 软核处理器的结构 6.2 Nios 软核处理器 6.2.1 “Core Nios”选项卡 6.2.2 “Caches and Memory Interfaces”选项卡 6.2.3 “Advanced Features”选项卡 6.2.4 “JTAG Debug Module”选项卡 6.2.5 “Custom Instructions”选项卡 6.3 程序运行时间的测量 6.3.1 程序运行时间的测量方法 6.3.2 Nios 软核处理器对程序运行时间的影响 6.4 浮点专用指令的使用 6.4.1 浮点专用指令的添加 6.4.2 浮点专用指令测试程序 6.4.3 浮点专用指令测试结果 6.5 小结第7章 8段数码管显示电路——并行输入/输出(PIO)内核的使用 7.1 8段数码管 7.2 并行输入/输出(PIO)内核 7.2.1 并行输入/输出(PIO)内核的寄存器 7.2.2 并行输入/输出(PIO)内核的配置 7.2.3 C语言编程 7.3 1位数码管的显示实验 7.3.1 产生数码管的显示控制电路 7.3.2 数码管显示控制程序 7.3.3 目标芯片的配置 7.4 多位数码管显示实验 7.4.1 多位数码管显示控制电路 7.4.2 4位数码管显示驱动函数 7.4.3 4位数据的分离 7.5 小结第8章 按键电路——中断的应用 8.1 按键电路 8.2 并行输入/输出(PIO)内核的中断 8.2.1 并行输入/输出(PIO)内核涉及中断的相关寄存器 8.2.2 并行输入/输出(PIO)内核中断的配置 8.2.3 C语言编程 8.3 Nios 处理器的中断 8.3.1 异常 8.3.2 C语言编程 8.4 1位按键电路的实验 8.4.1 产生按键的控制电路 8.4.2 1位按键控制程序 8.4.3 数据类型 8.4.4 alt\_main()和main()的区别 8.5 4位按键电路的实验 8.5.1 4位按键控制电路 8.5.2 4位按键控制程序 8.6 小结第9章 Flash的编程——EPCS控制器、CFI控制器的使用 9.1 EPCS控制器 9.1.1 EPCS控制器概述 9.1.2 EPCS控制器配置 9.2 CFI控制器 9.2.1 CFI控制器概述 9.2.2 CFI控制器配置选项 9.2.3 CFI控制器C语言编程 9.3 Flash的编程实例 9.3.1 硬件系统的SOPC设计——从EPCS引导程序方式 9.3.2 系统软件设计——从EPCS引导程序方式 9.3.3 从CFI\_FLASH引导程序方式 9.4 小结第10章 时钟信号的产生与测量——定时器(Interval Timer)内核的使用 10.1 定时器内核 10.1.1 定时器内核的组成 10.1.2 定时器内核的寄存器 10.1.3 定时器(Interval Timer)内核的配置 10.1.4 C语言编程 10.2 时钟信号产生实验 10.2.1 时钟信号产生电路 10.2.2 时钟信号产生电路控制程序 10.3 定时器(Interval Timer)内核的中断实验 10.4 信号的周期测量 10.4.1 信号周期测量电路 10.4.2 周期测量控制程序 10.5 “看门狗”电路实验 10.5.1 “看门狗”电路 10.5.2 “看门狗”电路控制程序 10.6 小结第11章 LCD12864液晶模块的驱动设计——SDRAM控制器内核的使用 11.1 LCD12864液晶模块简介 11.2 SDRAM控制器内核 11.2.1 概述 11.2.2 SDRAM控制器内核的配置选项 11.2.3 时钟、PLL和时序 11.2.4 SDRAM内核的C语言编程 11.3 LCD12864模块的驱动实例 11.3.1 LCD12864模块接口电路 11.3.2 硬件系统的SOPC设计 11.3.3 系统软件设计 11.4 小结第12章 JTAG UART通信——JTAG UART内核的使用 12.1 JTAG UART内核 12.1.1 JTAG UART内核概述 12.1.2 JTAG UART内核配置选项 12.1.3 JTAG UART内核的C语言编程 12.2 JTAG UART通信实例 12.2.1 硬件系统的SOPC设计 12.2.2 系统软件设计 12.3 小结第13章 SD卡读写

## <<基于NiosII内核的FPGA电路系统>>

控制设计——SPI内核的使用 13.1 SD卡简介 13.2 SPI内核 13.2.1 SPI内核综述 13.2.2 SPI内核配置选项 13.2.3 SPI的C语言编程 13.3 SD卡读写实例 13.3.1 SD卡与FPGA接口电路 13.3.2 硬件系统的SOPC设计 13.3.3 系统软件设计 13.4 小结第14章 数字信号频率测量电路的设计 14.1 频率测量原理和电路设计要求 14.2 频率测量电路的系统设计 14.3 频率测量电路的单元电路设计 14.3.1 计数器电路 14.3.2 计数器控制电路 14.3.3 Nios 软核处理器系统 14.4 频率测量电路的系统调试 14.4.1 系统集成 14.4.2 测试方案和使用仪器 14.4.3 测量数据及数据分析 14.5 设计总结第15章 信号频谱分析电路的设计 15.1 频谱的概念 15.2 离散傅里叶变换 15.3 信号频谱分析电路的系统设计 15.4 信号频谱分析电路的单元电路设计 15.4.1 频谱分析模块 15.4.2 采样速率控制电路 15.5 信号频谱分析电路的系统调试 15.5.1 系统集成 15.5.2 信号频谱分析程序框图 15.5.3 系统测量和数据分析 15.6 设计总结参考文献

## &lt;&lt;基于NiosII内核的FPGA电路系统&gt;&gt;

## 章节摘录

微处理器的每一条汇编指令对应一个电路模块，可编程逻辑器件可以认为是对标准逻辑器件的直接升级。

它在一块芯片内部集成了非常多的逻辑门和触发器，使得一块可编程逻辑器件芯片具有了实现一个应用系统的逻辑资源，从而减少了系统中使用器件的数量，提高了系统的可靠性。

芯片内部的逻辑资源的连接不需要手工进行，与微处理器一样，用户只需要编写设计文件即可，从而使得设计灵活性得到了提高。

可编程逻辑器件在下载设计文件后，在它的内部将形成对应的硬件电路，这些电路是可以同时工作的。

例如，向两个数码管传送显示代码，这时可以同时进行，而在微处理器中，这个工作是逐个传送的。

可编程逻辑器件内部电路模块中信号传输的时间延迟只来源于硬件电路产生的时间延迟，不存在指令读取和执行产生的时间延迟。

上述工作特点使得可编程逻辑器件的工作速度比微处理器芯片快。

可编程逻辑器件的开发设计语言有许多种，其中VHDL和Verilog HDL这两种硬件描述语言已经获得较为广泛的应用，并且成为IEEE的标准。

以VHDL为例，创建VHDL的最初目标是用于标准文档的建立和系统功能的模拟，基本想法是在高层次上描述系统或元件的行为。

到了20世纪90年代初，人们发现VHDL不仅可以作为系统模拟的建模工具，而且可以作为电路和系统的设计工具，即可以利用软件工具把VHDL源代码转换为基本逻辑元件的连接关系。

vHDL没有得到可编程逻辑器件开发软件的全面支持。

例如，Altera公司的Quartus可编程逻辑器件开发软件不支持浮点类型数据，不支持乘法和除法算术运算等。

这些不足限制了可编程逻辑器件的应用。

不支持浮点类型数据减小了所处理信号的动态范围，不支持乘、除法算术运算使得一些信号处理功能不容易实现。

在这方面，微处理器开发中大量使用的C语言具有明显的优势。

## <<基于NiosII内核的FPGA电路系统>>

### 编辑推荐

掌握一种技术需要大量的理论和实践学习，同时也需要一个好的学习方法，如果有一个好的学习指导是可以获得事半功倍的效果的。

通过这些年来对参加Altera公司的EDA/SOPC电子设计竞赛和全国大学生电子设计竞赛的学生的赛前训练的总结，作者编写了本书。

本书在撰写内容上力求实用，并尽量做到理论和实践相结合；在内容编排上，尽量做到由易到难。对书中列举的基础实验和应用设计，作者努力做到不仅描述如何进行，而且分析这样做的原因，使读者能够比较容易地举一反三。

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>