

<<零点起步>>

图书基本信息

书名：<<零点起步>>

13位ISBN编号：9787111372455

10位ISBN编号：711137245X

出版时间：2012-4

出版时间：机械工业出版社

作者：薛冰 等编著

页数：361

字数：577000

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>

## &lt;&lt;零点起步&gt;&gt;

## 内容概要

薛冰编著的《零点起步--Altera

CPLD\FPGA轻松入门与开发实例(附光盘)》面向FPGA初中级读者，全书共分11章，较系统地介绍了可编程逻辑器件的结构原理以及VHDL(超高速集成电路硬件描述语言)基础知识、语法结构和常用基本数字电路的例程，使读者能够轻松入门并巩固数字电路基础知识；详细分析了目前主流的可编程逻辑器件和开发工具，并介绍了逻辑电路设计过程中的一些基本问题和处理方法，以及如何在Altera公司的CPLD / FPGA芯片上运行VHDL程序，其中包含一些常用数字信号处理算法和接口控制器的设计实例；最后介绍基于SOPC的嵌入式设计的具体工作流程和设计方法，并结合实际工程设计，说明如何采用可编程逻辑器件实现典型的嵌入式系统。

本书内容翔实，讲解透彻，案例实用，能够使读者快速、全面地掌握 CPLD / FPGA的基本开发方法。

《零点起步--Altera

CPLD\FPGA轻松入门与开发实例(附光盘)》既可作为各类培训学校的教材，也可作为工程技术人员及大中专院校相关专业师生的参考书。

## 书籍目录

## 前言

## 第1章 可编程逻辑器件概述

## 1.1 EDA技术及其应用

## 1.2 可编程逻辑器件的分类及原理

## 1.2.1 EPLD和CPLD

## 1.2.2 FPGA

## 1.2.3 FPGA与CPLD的比较

## 1.3 PLD的设计开发

## 1.3.1 PLD设计方法

## 1.3.2 PLD设计流程

## 1.4 Altera可编程逻辑器件

## 1.4.1 Stratix系列

## 1.4.2 Arria系列

## 1.4.3 Cyclone系列

## 1.4.4 MAX系列

## 1.5 思考与练习

## 第2章 VHDL基础

## 2.1 VHDL概述

## 2.1.1 硬件描述语言概述

## 2.1.2 VHDL的特点及设计流程

## 2.2 VHDL程序基本结构

## 2.2.1 实体

## 2.2.2 结构体

## 2.2.3 配置

## 2.2.4 程序包

## 2.3 VHDL中的数据

## 2.3.1 标识符

## 2.3.2 数据对象

## 2.3.3 数据类型

## 2.4 VHDL中的表达式

## 2.5 VHDL描述语句

## 2.5.1 顺序描述语句

## 2.5.2 并行描述语句

## 2.6 思考与练习

## 第3章 VHDL程序设计

## 3.1 基本数字电路的VHDL描述

## 3.1.1 基本逻辑门电路

## 3.1.2 编码器和译码器电路

## 3.1.3 数据选择器

## 3.1.4 加法器

## 3.1.5 乘法器

## 3.1.6 触发器电路

## 3.1.7 寄存器电路

## 3.1.8 计数器电路

## 3.1.9 存储器

## &lt;&lt;零点起步&gt;&gt;

## 3.2 有限状态机

## 3.2.1 有限状态机的基本描述

## 3.2.2 有限状态机状态编码方式

## 3.2.3 有限状态机的输出和复位

## 3.3 应用实例——表决器、双口RAM的读取

## 3.4 思考与练习

## 第4章 Altera开发工具的使用

## 4.1 Quartus 集成开发环境

## 4.1.1 概述

## 4.1.2 Quartus 的安装

## 4.1.3 Quartus 10.0图形用户界面

## 4.1.4 设计输入

## 4.1.5 对设计工程进行设置和约束

## 4.1.6 设计综合

## 4.1.7 布局布线

## 4.1.8 仿真

## 4.1.9 时序分析

## 4.1.10 编程与配置

## 4.2 在Altera CPLD/FPGA中运行第一个程序

## 4.2.1 实验目的、环境及实验原理

## 4.2.2 实验步骤

## 4.3 应用实例——MAX 内嵌UFM模块的使用

## 4.3.1 实验目的、环境及实验原理

## 4.3.2 实验步骤

## 4.4 思考与练习

## 第5章 可编程逻辑设计的基本问题

## 5.1 可编程逻辑设计基本原则

## 5.1.1 面向硬件原则

## 5.1.2 系统性原则

## 5.1.3 面积与速度之间的互换关系

## 5.1.4 同步设计原则

## 5.2 常用设计思想与技巧

## 5.2.1 串并转换

## 5.2.2 应用实例——RS232数据串并转换

## 5.2.3 乒乓操作

## 5.2.4 流水线操作

## 5.3 常见问题及处理方法

## 5.3.1 竞争冒险产生的原因和解决方法

## 5.3.2 时钟设计常见问题和解决方法

## 5.3.3 建立和保持时间的处理方法

## 5.4 应用实例——利用FIFO缓存器做高速缓存

## 5.5 思考与练习

## 第6章 典型数字信号处理算法FPGA设计

## 6.1 坐标旋转数字计算机

## 6.1.1 CORDIC算法原理

## 6.1.2 CORDIC算法结构

## 6.1.3 CORDIC算法设计

## &lt;&lt;零点起步&gt;&gt;

- 6.1.4 程序分析及设计
- 6.2 有限脉冲响应数字滤波器
  - 6.2.1 FIR数字滤波器原理
  - 6.2.2 FIR数字滤波器程序设计
  - 6.2.3 程序分析及设计
- 6.3 无限脉冲响应数字滤波器
  - 6.3.1 IIR数字滤波器原理
  - 6.3.2 IIR数字滤波器程序设计
  - 6.3.3 程序分析及设计
- 6.4 快速傅里叶变换
  - 6.4.1 FFT算法原理
  - 6.4.2 FFT程序设计
  - 6.4.3 程序分析及设计
- 6.5 思考与练习
- 第7章 通用接口的FPGA设计
  - 7.1 通用异步收发器
    - 7.1.1 UART功能简介
    - 7.1.2 UART实现结构
    - 7.1.3 UART程序设计
  - 7.2 CAN总线控制器
    - 7.2.1 CAN总线协议概述
    - 7.2.2 CAN通信控制器实现框架
    - 7.2.3 CAN通信控制器程序设计
  - 7.3 以太网控制器
    - 7.3.1 以太网基本原理
    - 7.3.2 以太网控制器实现框架
    - 7.3.3 以太网控制器程序设计
  - 7.4 应用实例——双UART设计
  - 7.5 思考与练习
- 第8章 基于SOPC的嵌入式系统设计
  - 8.1 Altera SOPC技术及其设计开发流程
    - 8.1.1 SOPC技术简介
    - 8.1.2 SOPC Builder使用
    - 8.1.3 SOPC设计开发流程
  - 8.2 Nios 处理器
  - 8.3 Avalon总线
    - 8.3.1 Avalon总线的特点
    - 8.3.2 Avalon总线信号
  - 8.4 Nios 外围标准设备
    - 8.4.1 SDRAM控制器
    - 8.4.2 CFI控制器
    - 8.4.3 EPCS控制器
    - 8.4.4 并行输入/输出控制器
    - 8.4.5 定时器控制器
    - 8.4.6 UART核
    - 8.4.7 JTAG UART核
    - 8.4.8 SPI核

## &lt;&lt;零点起步&gt;&gt;

## 8.4.9 DMA控制器

## 8.4.1 0PLL核

## 8.5 自定制用户外设

## 8.6 基于Nios 系统开发实例

## 8.7 思考与练习

## 第9章 SOPC硬件开发

## 9.1 基于 Nios 的SOPC硬件系统开发流程

## 9.2 SOPC Builder硬件开发环境介绍

## 9.2.1 SOPC Builder的功能

## 9.2.2 SOPC Builder的组成

## 9.3 应用实例

## 9.3.1 创建一个Quartus 工程

## 9.3.2 使用SOPC Builder创建Nios 系统模块

## 9.3.3 添加CPU及外设IP模块

## 9.3.4 集成Nios 系统到Quartus 工程

## 9.3.5 设置编译选项并进行工程编译

## 9.3.6 将设计下载到目标FPGA中

## 9.4 思考与练习

## 第10章 SOPC软件开发

## 10.1 SOPC软件开发环境概述

## 10.1.1 Nios 指令系统介绍

## 10.1.2 Nios 处理器运行模式

## 10.2 Nios IDE

## 10.3 硬件抽象层系统库

## 10.3.1 HAL系统库的功能

## 10.3.2 使用HAL开发应用程序

## 10.4 应用实例

## 10.4.1 使用Nios IDE建立用户程序

## 10.4.2 创建一个新的C/C++应用工程

## 10.4.3 设置工程系统库属性和编译选项

## 10.4.4 编译连接工程

## 10.4.5 对程序进行运行调试

## 10.4.6 使用Flash Programmer下载

## 10.5 思考与练习

第11章 基于Nios 的 $\mu$ C/OS- 应用11.1  $\mu$ C/OS- 操作系统简介11.2 基于Nios 的 $\mu$ C/OS- 实验

## 11.2.1 系统结构图

## 11.2.2 SOPC系统的创建

## 11.2.3 基于Nios 软核处理器的软件

## 11.2.4 测试硬件设计

11.2.5 编写 $\mu$ C/OS- 的多任务控制程序

## 11.3 总结

## 11.4 思考与练习

## 参考文献

## 章节摘录

版权页：第1章 可编程逻辑器件概述可编程逻辑器件（Programmable Logic Device,PLD）起源于20世纪70年代，是在专用集成电路（Application Specific Integrated Circuit,ASIC）的基础上发展起来的一种新型逻辑器件，是当今数字系统设计的主要平台。

其主要特点是完全由用户通过软件进行配置和编程，从而完成特定的功能，并且可以反复擦写。

本章首先对电子设计自动化（Electronic Design Automation,EDA）技术及其应用进行介绍，然后给出可编程逻辑器件的分类及原理，进而简要介绍可编程逻辑器件的设计方法及流程，最后详细介绍Altera可编程逻辑器件。

本章要点： EDA技术及其应用。

可编程逻辑器件分类及原理。

PLD的设计方法及流程。

Altera可编程逻辑器件介绍1.1 EDA技术及其应用微电子技术的进步主要表现在大规模集成电路加工技术即半导体工艺技术的发展上，表征半导体工艺水平的线宽已经达到了45mm，并且还在不断地缩小；而在硅片单位面积上，则集成了更多的晶体管。

集成电路设计正在不断地向超大规模、极低功耗和超高速的方向发展，ASIC的设计成本不断降低，在功能上，现代的集成电路已能够实现单片电子系统（System On Chip,SOC）。

## <<零点起步>>

### 编辑推荐

《零点起步:Altera CPLD/FPGA轻松入门与开发实例》编辑推荐：从零开始，轻松入门，图解案例，清晰直观，图文并茂，操作简单，实例引导，专业经典，学以致用，注重实践。

《零点起步:Altera CPLD/FPGA轻松入门与开发实例》内容翔实，讲解透彻，案例实用，能够使读者快速、全面地掌握CPLD / FPGA的基本开发方法。

《零点起步:Altera CPLD/FPGA轻松入门与开发实例》既可作为各类培训学校的教材，也可作为工程技术人员及大中专院校相关专业师生的参考书。



#### 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>