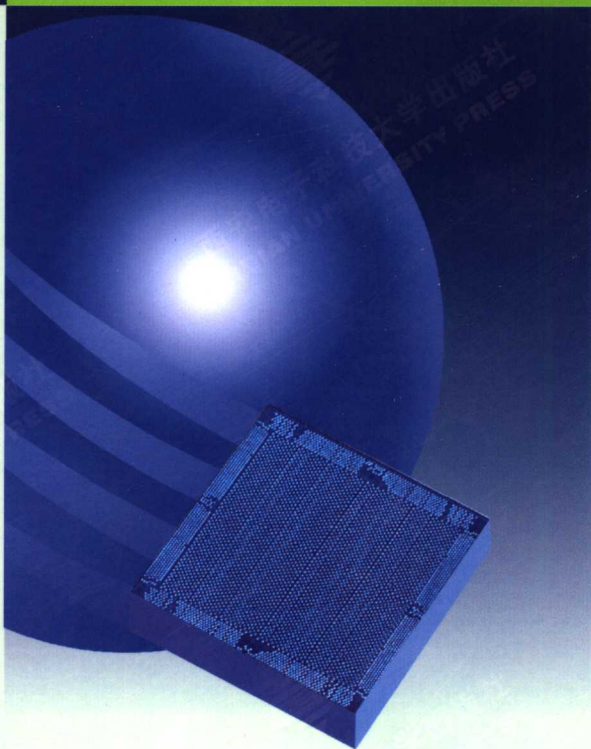


ALTERA®

The Introduction and Practice of SOPC Design

SOPC 设计基础与实践

王建校 危建国 编著



西安电子科技大学出版社
<http://www.xduph.com>

SOPC 设计基础与实践

王建校 危建国 编著

西安电子科技大学出版社

2006

内 容 简 介

本书主要介绍 Nios CPU 的结构,并结合 Quartus II 软件的基本使用方法,以大量的设计实例作为切入点,引导读者快速掌握 EDA 设计技术。本书图文并茂,举例简明易懂,适合于不同读者选用。

本书分为 4 篇,各篇内容相互独立,读者可以根据需要,选读自己所关注的内容。

本书第一篇介绍 Nios CPU 及外设,是 SOPC 的基础,所涉及的内容有 Nios CPU 的结构、Avalon 总线、外设的组织和使用。第二篇通过具体实例介绍 Quartus II 软件的使用方法,在此基础上,通过适当规模的数字系统设计(数字电子钟、数字频率计、电子抢答器、出租车计价器),分别展示了原理图、Verilog、VHDL 及混合系统设计的特点。第三、四篇向读者介绍 Nios CPU 的使用方法,并备有一定数量的实验(第三篇介绍 Nios CPU 的应用,第四篇介绍 Nios II CPU 的应用)。为方便读者学习,本书含配套光盘一张。

本书可供相关专业在校大学生、研究生使用,亦可供电子工程师和业余爱好者使用。

图书在版编目(CIP)数据

SOPC 设计基础与实践 / 王建校等编著. —西安:西安电子科技大学出版社, 2006.4

ISBN 7-5606-1653-4

I. S… II. 王… III. 微处理器—系统设计 IV. TP332

中国版本图书馆 CIP 数据核字(2006)第 016301 号

策 划 云立实 臧延新

责任编辑 邵汉平 龙 晖

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

http://www.xduph.com E-mail: xdupfbx@pub.xaonline.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2006 年 4 月第 1 版 2006 年 4 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 28.25

字 数 665 千字

印 数 1~4000 册

定 价 39.00 元(含光盘)

ISBN 7-5606-1653-4/TP·0398

XDUP 1945001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜,谨防盗版。

欢迎选购西安电子科技大学出版社工程应用类图书

程序员考试辅导	40.00	TCP/IP 协议与网络编程	30.00
网络管理员考试辅导	30.00	SDH 网络管理系统维护基础	30.00
软件设计师考试辅导	45.00	PC 计算机测控技术及应用	21.00
数据库系统工程师考试辅导	50.00	FPGA 设计及应用 (含光盘)	30.00
网络工程师考试辅导	35.00	基于 FPGA 的嵌入式系统设计	35.00
网络工程师考试试题分类详解	37.00	MPC860/850 嵌入式系统开发入门与指导	33.00
软件设计师考试试题分类详解	40.00	Cold Fire 嵌入式系统设计	27.00
多媒体应用设计师考试辅导	40.00	ARM 嵌入式系统设计及实践	25.00
程序员考试历年试题分析与解答	22.00	Linux 操作系统结构分析	40.00
网络管理员考试历年试题分析与解答	20.00	微型计算机嵌入式系统设计	34.00
软件设计师考试历年试题分析与解答	26.00	Nios 软核心嵌入式处理器设计大赛	
网络工程师考试历年试题分析与解答	23.00	优秀作品精选	30.00
数字 IC 系统设计	22.00	TMS320 5000 系列 DSP 汇编语言程序设计	30.00
系统安全工程能力成熟度模型		TMS320 C54X DSP 原理及应用	29.00
(SSE-CMM) 及其应用	18.00	USB 接口设计	25.00
C/C++ 语言硬件程序设计		USB 接口开发技术	30.00
——基于 TM320C5000 系列 DSPs	22.00	VHDL 与复杂数字系统设计	29.00
模糊聚类分析及其应用	21.00	Verilog HDL 数字系统设计及其应用	25.00
集成稳压器使用指南与应用电路	35.00	基于 Verilog 的 FPGA 设计基础	35.00
新型集成电路使用手册及应用实例	43.50	VHDL 硬件描述语言	
新型集成电路简明手册及典型应用 (上)	55.00	与数字逻辑电路设计 (修订版)	20.80
新型集成电路简明手册及典型应用 (下)	52.00	PCI 局部总线及其应用	29.00
专用集成电路设计基础	29.00	DSP 接口电路设计与编程	24.00
电路设计与制版——Protel DXP 实用教程	27.00	DSP 程序开发	
DC/DC 变换器集成电路及应用		——MATLAB 调试及直接目标代码生成	37.00
——降压式 DC/DC 变换器	38.00	高性能 DSP 与高速实时信号处理 (第二版)	35.00
DC/DC 变换器集成电路及应用		Windows 环境下的设备驱动程序设计	28.00
——升压式 DC/DC 变换器	39.00	IEEE-1394 (Fire Wire) 系统原理与应用技术	22.00
现代电子电路应用基础	32.00	IEEE 1394 协议及接口设计	45.00
电子标签原理与应用 (含光盘)	36.00	CPLD 技术及应用	25.00
弱电系统综合布线	18.00	IC 设计基础	32.00
综合布线系统方案设计	30.00	IP 电话终端设备——原理、电路及应用	22.00
基于现场总线 Device Net		Multisim 2001 与电子技术仿真实验	20.00
的智能设备开发指南	22.00	数字水印技术	20.00
无线局域网 (WLAN)		高性能 SoC 模拟信号处理单片机	
——原理、技术与应用	55.00	MSC1210 原理与开发应用	21.00
工程电磁兼容	19.00	新型单片机接口器件与技术	34.00
网络技术	25.00	单片机系统设计与工程应用	26.00

单片机原理及实用技术:		PLD 与数字系统设计	25.00
凌阳 16 位单片机原理及应用	30.00	EDA 技术入门与提高	25.00
流行单片机实用子程序及应用实例	19.00	OPNET Moderler 网络建模与仿真	30.00
单片机数据通信技术从入门到精通	27.00	Pro/ENGINEER 建模实例及快速成型技术	25.00
电子电路 CAD 技术		现代测试技术及应用	33.00
(基于 ORCAD9.2 版, 含光盘)	29.00	短信与 BREW 开发技术及实践 (含光盘)	42.00
PWM 控制与驱动器使用指南及应用电路		OrCAD/Pspice 9 实用教程	31.00
——单端控制与驱动器部分	28.00	语音编码	24.00
——双端控制与驱动器部分	30.00	变速率语音编码	34.00
——SPWM、PPC 和 IGB 控制与驱动器部分	39.00	知识元挖掘	25.00
开关稳压电源		MATLAB 基础与编程入门	23.00
——原理设计及实用电路 (修订版)	19.00	MATLAB 外部接口编程	20.00
数据采集与分析技术	30.00	MATLAB 应用程序集成与发布	30.00
组态软件设计与开发	28.00	MATLAB 辅助模糊系统设计	22.00
软件质量工程	18.00	Simulink 建模与仿真	24.00
便携式电子设备电源管理技术	39.00	Simulink 动态系统建模与仿真基础	26.00
智能传感器系统	35.00	MATLAB 遗传算法工具箱及应用	26.00
传感器与检测技术	45.00	MATLAB 仿真在通信与电子工程中的应用	36.00
传感器简明手册及应用电路		MATLAB 应用图象处理	22.00
——温度传感器分册 (上册)	36.00	MATLAB 小波分析高级技术	26.00
虚拟仪器图形化编程语言 LabVIEW 教程	26.00	MATLAB 及其在理工课程中的应用指南	
现代通信理论与技术导论	25.00	(第二版)	20.00
网络对抗原理	42.00	基于 MATLAB6.X 的系统分析与设计	
天线理论与技术	30.00	——神经网络 (第二版)	20.00
无线电抗干扰通信原理及应用	32.00	——时频分析	20.00
通信光缆线路工程与维护	30.00	基于 MATLAB7.X 的系统分析与设计	
图像通信	36.00	——信号处理 (第二版)	34.00
卫星光通信导论	13.00	——控制系统 (第二版)	22.00
电磁场并行计算	22.00	基于 MATLAB 的系统分析与设计	
数据压缩	30.00	——小波分析 (第二版)	27.00
智能仪器与数据采集系统中的新器件及应用	26.00	——模糊系统	14.00
用 ISP 器件设计现代电路与系统	26.00		

★★

欢迎来函索取本社最新书目和教材介绍, 欢迎投稿!

从邮局或银行汇款邮购者, 汇款单上务必写清收书人姓名、地址、邮编、电话。款到后我社将挂号发书, 加收5元包装邮寄费 (一次购书30元以上者可免收邮费)。

通信地址: 西安市太白南路2号 西安电子科技大学出版社发行部 邮 编: 710071

电 话: (029) 88201467 传 真: (029) 88213675

主 页: <http://www.xduph.com> E-mail: xdupfbx@pub.xaonline.com

前 言

随着电子技术和计算机应用技术的深入发展以及 EDA 设计技术的不断进步与完善, 不仅给电子系统的设计和应用带来了新的设计思路和发展机遇, 也对传统的电子系统设计手段提出了严峻的挑战。

传统的电子系统设计, 是以各种不同的集成电路芯片为基础, 按照功能要求在印制电路板上将不同的芯片拼连、组合, 构成实现某种功能的电子系统。这样的设计方法不仅繁琐, 而且设计过程中的错误和不足之处不能及早地发现; 进入调试阶段后, 一旦发现错误或缺陷, 也不能现场更正。无疑, 这将使研发时间变长, 研发成本加大。能够克服上述缺点的是片上可编程系统(简称 SOPC), 它向传统的电子系统设计方法发起了挑战。

微电子技术、计算机应用技术的飞速发展, 不仅使得电子系统的小型化、微型化进程加快, 而且给电子系统设计带来了前所未有的变革。大规模 FPGA 芯片的问世, 为电子系统设计提供了硬件基础, 几乎大多数电子系统都可以在一块芯片上实现。ALTERA 公司的 Quartus II 软件就是 EDA 设计的最优秀软件之一, 使用它不仅可以灵活地设计电子系统, 而且还可以对设计方案进行模拟仿真, 及早发现错误和缺陷。使用 FPGA 的好处不仅表现在设计的初级阶段, 即使在电子系统设计完成之后, 甚至在投入实际使用的过程中, 还可以根据实际需要添加功能。更为激动人心的是, SOPC 设计是可裁剪和可配置的, 尤其 Nios CPU 的构造是可以灵活配制的。

SOPC 还是 SOC 设计的基础, 也是很好的硬件实践平台。对于某种专用电路, 可以首先在 FPGA 上实现, 然后根据实际需要制作成 SOC 专用集成芯片。

在 SOPC 课程的教学过程中, 我们发现有关这一方面的资料很稀少, 为此我们编写了教学参考和实践指南等一些资料, 得到了学生们的好评, 也得到很多同仁的认同。是他们给了我们热情的支持与鼓励, 并希望尽快成书。不辜负大家的期望, 本书就要和大家见面了。

此处, 我们要向参与实践本书内容的学生, 向热情支持与鼓励我们的同仁, 表示最诚挚的感谢。

由于作者水平有限, 书中难免存在错误和遗漏, 殷切希望读者批评指正。还希望广大读者将使用过程中所遇到的问题、成功的经验告诉我们, 以便再版时修订或增补, 从而使本书日趋完善。

作者 E-mail: wjxdzx@sohu.com

作 者
于西安交通大学
2006 年 1 月

Preface

The ever increasing pace of the electronics industry is presenting many challenges and opportunities for engineers. Universities and engineering students must prepare themselves by strengthening EDA, SOPC knowledge and capability. Altera's goal is to provide the most effective and flexible platform to enable the electronics industry to realize its full potential of innovation and creativity.

One of the biggest challenges faced by design engineers is the selection of a processor for their next design. FPGA-based embedded systems can provide scalable performance, allowing last-minute changes to boost system performance based on end market demands. Compute-intensive algorithms, converted to logic in an FPGA, can run orders of magnitude faster than the same algorithm run in software by a microprocessor or digital signal processor. More importantly, hardware resources can be applied to performance-hungry algorithms where they are needed most, potentially reducing the need for a high-performance CPU, reducing clock frequency, reducing power consumption, and simplifying the board design.

With the advent of low-cost FPGA families, in which you can implement one or more embedded processors, custom embedded solutions are now viable for cost-sensitive applications and are well within the reach of the average embedded design engineer. Altera's Nios® II embedded 32-bit processors, implemented in programmable logic, use on-chip resources such as multipliers and memory, and can be instantiated in almost any FPGA family. The performance and cost of a soft-core processor depend mainly on the FPGA in which it is instantiated.

FPGA-based embedded systems open a powerful set of new options for the embedded design engineer. Nios II based SOPC designs can be changed and updated right up to the time the product goes into final test allowing maximum flexibility.

I am grateful to Professor Wang for leading the effort to compile such a comprehensive book on Quartus® II software based SOPC design. I am sure this book will help students and industry to fully realize their potential for innovation and creativity.

Ben Lee
Vice President, Asia Pacific
Altera Corporation

目 录

第一篇 Nios CPU 及外设

第 1 章 Nios CPU	1
1.1 指令总线主端口	2
1.2 数据总线主端口	2
1.3 高速缓存	2
1.4 移位单元	3
1.5 乘法支持	3
1.6 中断支持	3
1.7 Nios 片上调试模块(OCI 模式)	4
第 2 章 Nios CPU 编程模型	5
2.1 寄存器	5
2.2 存储器结构	10
2.3 寻址方式	15
2.4 程序流程控制	17
2.5 异常事件	18
2.6 流水线	22
第 3 章 Avalon 总线	24
3.1 概述	24
3.2 术语和概念	25
3.3 Avalon 总线传输	28
3.4 Avalon 从端口传输	30
3.5 Avalon 主端口传输	34
3.6 高级 Avalon 总线传输	35
3.7 片外设备与 Avalon 总线的接口	37
3.8 Avalon 总线地址对齐方式	40
3.9 连接到外部设备	41
第 4 章 外设的组织与使用	43
4.1 SOPC Builder 与 PTF 文件	43
4.2 定时器	44
4.3 并行输入/输出口	47
4.4 异步收发器(UART)	49
4.5 DMA 控制器	57
4.6 串行外围设备接口(SPI)	60

第二篇 Quartus II 应用入门

第 5 章 Quartus II 应用基础	67
5.1 原理图工程	68
5.2 文本文件(Verilog)工程	87
5.3 文本文件(VHDL)工程	105
5.4 文本文件(AHDL)工程	107
5.5 混合工程(自底而上)	108
5.6 混合工程(自顶而下)	112
第 6 章 设计举例	123
6.1 7 段数码管译码器(DE4_7)	123
6.2 8 选 1 多路选择器(MUL8_1)	125
6.3 计数器(CNT1000)	126
6.4 50 kHz 分频器	127
6.5 动态数码管显示(DISPLAY)	128
6.6 数字电子钟(CLK)	129
6.7 数字频率计(FREQ)	132
6.8 电子抢答器	134
6.9 出租车计价器	138
第 7 章 Quartus II 的工具	144
7.1 在线逻辑分析仪(SignalTap II)	144
7.1.1 捕捉条件	144
7.1.2 存储深度	144
7.1.3 观察信号	145
7.1.4 在线逻辑分析仪使用举例	145
7.2 TCL 屏幕控制台	150
第 8 章 Verilog 语言设计举例	154
8.1 数字电子钟	154
8.1.1 分频器 CNT1000_v	154
8.1.2 分频器 CNT50000_v	155
8.1.3 六十进制计数器 CNT60_v	156
8.1.4 二十四进制计数器 CNT24_v	157
8.1.5 时钟控制 CLK_C_v	157
8.1.6 动态显示 DISPLAY_v	158
8.2 数字频率计	160
8.2.1 14 位计数器 CNT14BITE_v	161
8.2.2 14 位 D 触发器 DFF14BITE_v	162
8.3 电子抢答器	163
8.3.1 抢答识别 MS_SL_v	163

8.3.2 三选一 MUL3_v	167
8.3.3 积分器 JIFEN_v	167
8.4 出租车计价器	169
8.4.1 分频器 X7456_v	170
8.4.2 二选一 mux_v	171
8.4.3 计价收费 my_jijiaqi_v	171
第 9 章 VHDL 语言设计举例	173
9.1 数字电子钟(CLK)	173
9.1.1 分频器 count	173
9.1.2 计数器 count_bcd	174
9.1.3 动态显示器 display	176
9.1.4 时钟控制 time_set	177
9.2 数字频率计	178
9.2.1 计数器 t_trigger	178
9.2.2 14 位计数器 freq_count	179
9.2.3 14 位锁存器 my_dff	181
9.3 电子抢答器	181
9.4 出租车计价器	185
9.4.1 二选一 mux2	186
9.4.2 计价收费 my_jijiaqi	186

第三篇 Nios 应用入门

第 10 章 SOPC Builder	191
10.1 创建工程项目	191
10.2 构造 Nios CPU 及其外设	193
第 11 章 GERMS 监控	207
11.1 GERMS 监控简介	207
11.2 Nios SDK Shell	208
11.3 Nios SDK Shell 的命令	210
第 12 章 Nios CPU 系统扩展	211
12.1 扩展外部 SRAM	211
12.1.1 SRAM 芯片	211
12.1.2 SRAM 的扩展实例	213
12.2 扩展外部 Flash	220
12.2.1 性能比较	221
12.2.2 接口差别	221
12.2.3 容量和成本	221
12.2.4 易用性	222
12.2.5 软件支持	222

12.2.6 Flash 的扩展实例	222
12.3 串行配置器件	227
12.4 外部中断	235
12.4.1 单一外部中断源	235
12.4.2 多个外部中断源	238
12.5 定时器	240
第 13 章 Nios 系统的接口实验	243
13.1 模/数转换(ADC0832)	243
13.2 数/模转换(TLC5620)	247
13.3 键盘、显示器控制器(7289A)	252
13.4 点阵液晶显示器(T6963C)	257
13.5 时钟日历(DS1337)	276
13.6 IC 卡(24C01)	281
13.7 语音录放(ISD25120)	284

第四篇 Nios II 应用入门

第 14 章 SOPC Builder	289
14.1 创建工程项目	289
14.2 构造 Nios II CPU 及其外设	291
第 15 章 Nios II IDE	302
15.1 新建工程	302
15.2 重新进入已有工程	311
第 16 章 Nios CPU 系统扩展	317
16.1 扩展外部 SRAM	317
16.1.1 SRAM 芯片	318
16.1.2 SRAM 的扩展实例	319
16.2 扩展外部 Flash	326
16.2.1 性能比较	326
16.2.2 接口差别	326
16.2.3 容量和成本	327
16.2.4 易用性	327
16.2.5 软件支持	327
16.2.6 Flash 的扩展实例	327
16.3 串行配置器件	339
16.3.1 配置数字逻辑	339
16.3.2 配置 CPU 系统	346
16.4 外部中断	347
16.4.1 单一外部中断源	347
16.4.2 多个外部中断源	351

16.5 定时器	353
第 17 章 Nios 系统的接口实验	355
17.1 模/数转换(ADC0832).....	355
17.2 数/模转换(TLC5620).....	359
17.3 键盘、显示器控制器(7289A)	363
17.4 点阵液晶显示器(T6963).....	369
17.5 时钟日历(DS1337)	387
17.6 IC 卡(24C01).....	391
17.7 语音录放(ISD25120).....	395
附录 A 多功能电子学习机系统介绍.....	399
A.1 概述	399
A.1.1 设计目的	399
A.1.2 使用对象	399
A.1.3 多功能电子学习机的特色	399
A.1.4 多功能电子学习机的结构	399
A.2 多功能电子学习机的模块	400
A.2.1 电源模块	401
A.2.2 逻辑指示模块	401
A.2.3 数字量模块	403
A.2.4 按键模块	403
A.2.5 并行输入模块	403
A.2.6 并行输出模块	404
A.2.7 键盘显示器模块	405
A.2.8 点阵液晶模块	406
A.2.9 A/D 转换模块	406
A.2.10 D/A 转换模块	406
A.2.11 IC 卡、日历模块	407
A.2.12 语音模块	407
A.2.13 用户 PLD 模块.....	409
A.2.14 串行(RS232)通信模块.....	409
A.2.15 时钟	410
A.2.16 蜂鸣器	411
A.2.17 配置模块 1	411
A.2.18 配置模块 2	411
附录 B FPGA 系统模块.....	414
B.1 FPGA 模块	414
B.2 静态 RAM 模块	420
B.3 Flash 模块	421
B.4 串行接口模块	422

B.5 时钟模块	423
B.6 电源模块	423
B.7 配置模块	425
B.8 Nios 模块的引脚资源	426
附录 C 实验显示板	433
C.1 数码管及其驱动	434
C.2 数码管位驱动	434
C.3 用户键盘	345
C.4 用户逻辑指示器	436
C.5 用户逻辑笔	436
C.6 连接件	436

第一篇

Nios CPU 及外设

第 1 章 Nios CPU

Nios 3.0 版本的 CPU 是一个具有 5 级流水线、指令总线 and 数据总线分开(Harvard 结构)的 RISC(Reduction Instruction Set Computer, 精简指令集计算机)微处理器, 指令和数据的控制端口都作为 Avalon 总线主端口(Bus-Master)。用户可以把 Nios 的 Avalon 总线主端口和任何从端口(Slaves, 如内存和外设)通过 Avalon 总线互联起来, SOPC Builder 会根据需求自动加入仲裁器。Nios CPU 是可配置、可优化的软核, 可以和用户逻辑一起植入一个 Altera FPGA 芯片之中, 其结构如图 1-1 所示。

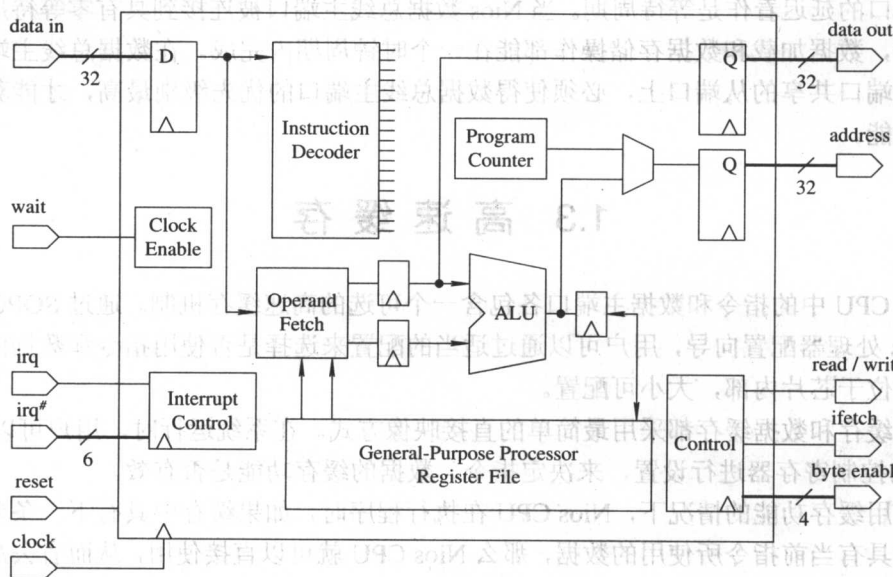


图 1-1 Nios CPU 结构图

1.1 指令总线主端口

Nios 指令总线主端口(Instruction Bus-Master)是 16 位宽的端口,支持延时操作。此主端口仅仅是负责从存储器中读取指令的通道,不支持任何写操作。因为主端口支持延时操作,所以能够适用于各种不同速度的存储器。指令主端口可以在上一条指令返回之前发出新的取指请求。Nios CPU 采用“假设无分支(branch-not-taken)”的预测方法来生成预取指的地址。由于支持具有操作延迟的存储器,因此使得在使用慢速存储器时对 CPU 的影响减少到最小,并能在整体上提高系统的最高频率;只有分支预测失败时,才会有比较大的延迟发生。当访问慢速存储器时,用户还可以选用片内缓存机制来提高读取指令的平均速度。

由 SOPC Builder 自动产生的 Avalon 总线具有动态总线宽度对齐逻辑的功能。因此,在 Nios 指令总线主端口上可以连接 8、16 和 32 位宽的存储器,以满足不同应用场合的需要。

1.2 数据总线主端口

对于 32 位的体系结构, Nios 数据总线主端口(Data Bus-Master)的宽度为 32 位;对于 16 位的体系结构,则宽度为 16 位。数据主端口有以下三种用途:

- 当 CPU 执行一个数据加载指令(LD, LDP, LDS)时,从存储器中读取数据。
- 当 CPU 执行一个数据存储指令(ST, STP, STS, ST8s, ST8d, ST16s, ST16d, STS8s, STS16s)时,向存储器写入数据。
- 当 CPU 执行 TRAP 指令或处理内、外部异常时,从中断向量表中取出中断向量。

Nios 数据总线主端口不支持延迟操作(因为预测地址没有意义),因此,数据主端口把来自从端口的延迟看作是等待周期。当 Nios 数据总线主端口被连接到具有零等待周期的存储器上时,数据加载和数据存储操作都能在一个时钟周期内完成。在数据总线主端口和指令总线主端口共享的从端口上,必须使得数据总线主端口的优先级别最高,才能获得最高的工作性能。

1.3 高速缓存

Nios CPU 中的指令和数据主端口各包含一个可选的高速缓存机制。通过 SOPC Builder 中的 Nios 处理器配置向导,用户可以通过适当的配置来选择是否使用指令和数据的缓存功能。缓存位于芯片内部,大小可配置。

指令缓存和数据缓存都采用最简单的直接映像方式。在系统运行时,用户可以对 Nios CPU 内的控制寄存器进行设置,来决定指令、数据的缓存功能是否有效。

在启用缓存功能的情况下, Nios CPU 在执行程序时,如果缓存中具有下一条要执行的指令或者具有当前指令所使用的数据,那么 Nios CPU 就可以直接使用,从而省去从外部存储器中获取指令或数据的时间,我们把这种情况简称为缓存命中。当缓存有效时,缓存命中就会使得存储器的加载操作在单个时钟周期内完成;而当缓存不命中时,就会引起额外

的延迟。当禁止缓存时(暂时以软件方式禁止缓存功能),访问存储器就会引起额外的延迟。但当重新启用缓存机制时,如果对存储器进行读/写操作,则将导致一个或两个额外的延迟周期(类似地,对于使用缓存机制的存储器,写操作也将导致一个或两个额外的延迟周期)。

值得注意的是,高速缓存仅仅在 32 位 Nios CPU 系统中才能实现。另外, Nios 缓存机制也仅仅在 Stratix、Stratix GX 和 Cyclone 系列芯片中才能配置。

1.4 移位单元

Nios CPU 使用固定的、不可配置的桶形移位逻辑来执行所有的移位指令(ASR, ASRI, ASL, ASLI, LSL, LSLI, RLC 和 RRC),并在 2 个时钟周期内完成(与移位的位数无关)。在做控制和数字信号处理时,移位单元对所需的移位操作提供了必要、方便的支持。

1.5 乘法支持

Nios CPU 有三种不同的方法来实现整数乘法,分别介绍如下。

1) MUL 指令

32 位的 Nios CPU 可以选择配置成 $16 \times 16 \rightarrow 32$ 的整数硬件乘法器, MUL 指令可以在 3 个或更短的时钟周期内计算出一个 32 位的结果。MUL 选项被选中后, MUL 指令将利用由 SDK(软件开发包)自动产生的 C 语言运行支持库(C-runtime 库)来实现乘法运算。值得注意: 16 位的 Nios 指令系统不支持此选项。

2) MSTEP 指令

32 位的 Nios CPU 可以配置成执行单步 16×16 的乘法运算器,硬件乘法器将在 2 个时钟周期内完成部分(其余部分在软件辅助下完成)乘法运算。MSTEP 选项被选中后, SDK 生成的 C 语言运行支持库就会支持相应的乘法操作,通过使用 MSTEP 指令,可连续实现乘法操作,例如用 16 个连续的 MSTEP 指令实现 $16 \times 16 \rightarrow 32$ 乘法。由于使用 MSTEP 所占用的 CPU 硬件资源还不到 5%,很多情况下利用如此小的硬件开销换得了对硬件乘法的支持,这无疑给软件设计带来很大的方便,因此系统将 MSTEP 作为默认选项。同样值得注意: 16 位的 Nios 指令系统不支持此选项。

3) 软件乘法器

当禁止 MSTEP 和 MUL 两种选项而需要完成乘法运算时,则由 SDK 中的 C 语言运行支持库利用移位和加法指令来实现整数乘法运算。软件乘法器虽然占用了很少的 CPU 硬件逻辑,但是比起以硬件实现的乘法运算,其执行速度要慢很多。

1.6 中断支持

Nios CPU 允许用户取消对 TRAP 指令、硬件中断或内部异常的事件处理,这种选择仅适用于非常简单的系统。在这种配置下, Nios CPU 将具有如下特征:

- 不包括 irq 输入引脚。

- 无异常处理(Trap 指令未定义)。
- 在执行 SAVE 和 RESTORE 指令使寄存器文件上溢/下溢时, 不产生异常中断。仅当有如下要求时才能取消陷阱指令、硬件中断或内部异常: 需要最小化的 Nios CPU 核; 确定应用程序不会产生寄存器窗口溢出异常, 即子程序的调用深度小于寄存器窗口数。

- 系统没有任何硬件中断源。
- 汇编代码不包含陷阱指令。

注意: Nios 的 SDK 在编译过的代码中不会产生陷阱指令。缺省情况下, 中断支持开启功能。

1.7 Nios 片上调试模块(OCI 模式)

Nios CPU 有个可选的片上 JTAG 调试模块, 它通过标准的 JTAG 接口实现与 CPU 的通信。此模块是由 First Silicon Solutions (FS2)公司开发的 IP 核, 被称为 Nios OCI(On-Chip Instrumentation)调试模块。

在调试应用程序时, Nios OCI 调试模块支持用户设置硬件断点, 还可以实现软件跟踪。所跟踪的数据保存在片上的存储器或者外部的系统分析器中(如 FS2 公司的 ISA-Nios In-System Analyzer)。该模块还可以向寄存器和存储器读/写数据, 允许在程序运行期间向存储器下载软件程序和检查寄存器。Nios OCI 调试模块可控制 CPU 的执行, 无论何时, 用户都可以对正在执行的程序进行控制。也就是说, 这种控制作用的优先级别最高。

除了调试功能外, 在与主设备通信时, Nios OCI 也能被用作标准的输入/输出。该功能与调试功能可同时使用, 且互不干扰, 但数据的传输速率比起利用串口作为输入/输出时要慢一些。