

<<安腾体系结构>>

图书基本信息

书名：<<安腾体系结构>>

13位ISBN编号：9787302096085

10位ISBN编号：7302096082

出版时间：2005-1-1

出版时间：清华大学出版社

作者：James S.Evans, Gregory L.Trimper, 蒋敬旗, 李华伟, 尹志刚, 唐冬蕾

页数：416

字数：647000

译者：蒋敬旗, 李华伟, 尹志刚, 唐冬蕾

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>

<<安腾体系结构>>

内容概要

本书全面介绍了新型64位安腾体系结构及其具有突破性的性能。

通过使用标准命令行工具和大量实例，循序渐进地介绍了安腾汇编语言，并结合当代计算机体系结构的发展，详细阐述了安腾体系结构设计。

本书涵盖的内容包括：
· 显式并行指令计算（EPIC）的潜能
· 安腾体系结构中的创新，例如寄存器栈引擎（RSE）和广泛使用的谓词等
· 过程调用以及过程调用机制
· 浮点操作
· I/O技术，包括从简单的调试到文件的使用等
· 开主和源码、Intel和HP-UX编译程序的输出优化
本书每章都包括翔实的数字、论述以及编程练习，并且提供了大量的参考文献和丰富的网上资源。

本书可作为计算机及相关专业学生学习计算机体系结构或汇编语言的教材，也是研究人员系统了解安腾体系结构的必备资源。

<<安腾体系结构>>

作者简介

蒋敬旗，男，1967年7月生，2001年在北京理工大学计算机科学与工程系获计算机应用技术专业工学博士学位。

2001年1~6月在信息产业部数字技术研究中心技术总监。

2001年7月至2003年10月在中国科学院计算技术研究所做博士后研究工作。

1993年曾获大学英语专业文学学士学位。

现在

<<安腾体系结构>>

书籍目录

第1章 体系结构和实现方法 1.1 类比：钢琴的体系结构 1.2 计算机语言的类型 1.3 为什么要学习汇编语言 1.4 二进制倍数的词头 1.5 指令集体系结构 1.6 计算机体系结构的生命周期 1.6.1 32位Intel体系结构及其先前结构 1.6.2 AlphaTM体系结构及其先前结构 1.6.3 安腾体系结构及其先前结构 1.6.4 体系结构和实现方法的命名 1.7 SQUARES：第一个编程实例 1.7.1 C、FORTRAN和COBOL语言描述 1.7.2 安腾体系结构的汇编语言描述 1.8 记数系统的回顾 1.8.1 位置系数和权值 1.8.2 二进制和十六进制表示 1.8.3 带符号整数 本章总结 参考文献 练习题第2章 计算机结构和数据表示 2.1 计算机结构 2.1.1 中央处理器 2.1.2 存储器 2.1.3 输入输出系统 2.2 指令的执行 2.3 指令集体系结构的类别 2.4 向64位体系结构过渡 2.5 安腾体系结构的信息单位和数据类型 2.5.1 整数 2.5.2 浮点数 2.5.3 字母数字字符 本章总结 参考文献 练习题第3章 汇编程序和调试程序 3.1 编程环境 3.2 程序开发步骤 3.3 比较源文件的不同 3.4 汇编语句的类型 3.4.1 语句格式 3.4.2 符号地址 3.4.3 汇编语言操作符的种类 3.5 符号汇编程序的功能 3.5.1 常量 3.5.2 符号或标识符 3.5.3 存储器分配 3.5.4 单元计数器 3.5.5 表达式 3.5.6 控制语句 3.5.7 清单文件的组成部分 3.6 汇编过程 3.7 连接过程 3.8 调试程序 3.8.1 调试程序的功能 3.8.2 使用gdb(Linux和HP-UX)运行SQUARES 3.8.3 使用adb(HP-UX)运行SQUARES 3.8.4 调试命令举例 3.9 编写程序的约定 本章总结 参考文献 练习题第4章 安腾指令格式和寻址方式 4.1 安腾指令格式概述 4.1.1 指令包 4.1.2 指令位字段格式 4.1.3 安腾指令类型 4.2 整数算术指令 4.2.1 加法和减法 4.2.2 算术溢出 4.2.3 左移相加指令第5章 比较、转移和判断第6章 逻辑操作、移位操作和字节操作第7章 子例程、过程和函数第8章 浮点操作第9章 文本的输入和输出第10章 性能考虑第11章 查看编译程序的输出第12章 并行操作第13章 实现方法的变异附录A 命令行环境附录B 推荐的系统资源附录C 安腾指令集附录D 安腾寄存器及其用途附录E 条件汇编和宏(GCC汇编程序)附录F 直接插入式汇编参考文献汇总部分练习题答案及提示术语表

<<安腾体系结构>>

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>