

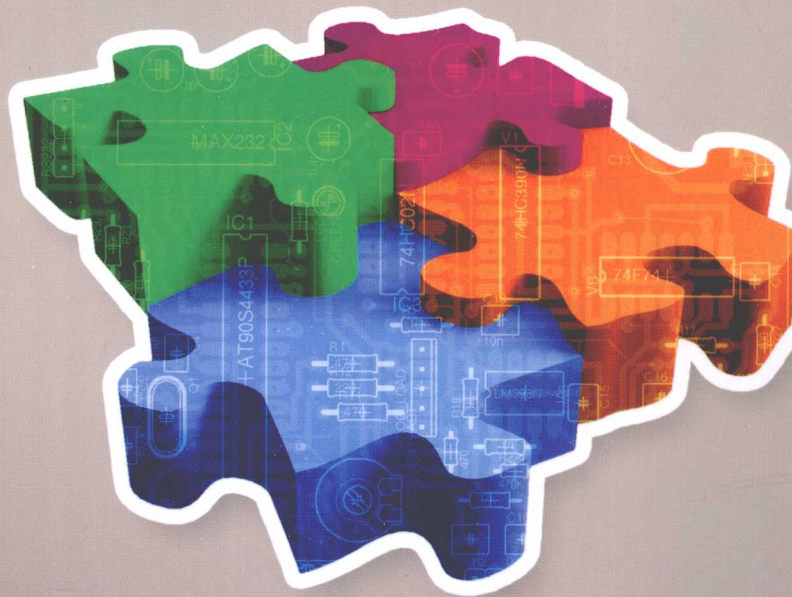


普通高等教育“十一五”国家级规划教材

高等院校重点推荐教材

微型计算机原理 与接口技术 (第二版)

赵宏伟 于秀峰 黄永平 秦贵和 编著



科学出版社

www.sciencep.com

普通高等教育“十一五”国家级规划教材
高等院校重点推荐教材

微型计算机原理与接口技术

(第二版)

赵宏伟 于秀峰 黄永平 秦贵和 编著

科学出版社

北京

内 容 简 介

本书将“微型计算机原理”、“微型计算机接口技术”和“汇编语言程序设计”三门课程的内容有机地融为一体。《微型计算机原理与汇编语言程序设计》和本书为同一门课程连续使用的两本教材。本书以 Pentium 的实模式与保护模式为主线,用 Pentium 实模式的实现技术来替代 Intel 8086 的内容;通过分析 Pentium 的保护模式,把当今微机领域内具有代表性的新设计、新技术、新思想和新潮流展示给读者;列举了一定数量的 I/O 接口硬件及程序设计实例,有助于建立微机系统的整机概念,以加深对微机工作过程的理解。

本书共 8 章,内容包括: Pentium 保护模式存储管理;输入输出;中断;总线技术;可编程接口芯片及其应用;串行通信和可编程串行接口芯片 8251A;模数转换及数模转换;人机交互接口。

本书可作为高等学校计算机科学与技术、通信工程、电气工程及其自动化等专业的教材,也可供从事计算机应用工作的工程技术人员及其他自学者学习和参考。

图书在版编目(CIP)数据

微型计算机原理与接口技术/赵宏伟等编著. —2 版. —北京:科学出版社, 2010

(普通高等教育“十一五”国家级规划教材·高等院校重点推荐教材)
ISBN 978-7-03-027636-0

I. ①微… II. ①赵… III. ①微型计算机-理论②微型计算机-接口
IV. ①TP36

中国版本图书馆 CIP 数据核字 (2010) 第 090625 号

责任编辑:鞠丽娜/责任校对:刘玉靖

责任印制:吕春珉/封面设计:三函设计

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

源海印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

*

2004 年 4 月第 一 版 开本: B5 (720×1000)

2010 年 6 月第 二 版 印张: 22 3/4

2010 年 6 月第四次印刷 字数: 456 000

印数: 9 001—12 000

定价: 36.00 元

(如有印装质量问题,我社负责调换〈新欣〉)

销售部电话 010-62134988 编辑部电话 010-62138978-8002

版权所有, 侵权必究

举报电话: 010-64030229; 010-64034315; 13501151303

第二版前言

《微型计算机原理与汇编语言程序设计》(科学出版社, 2004 年 3 月第一版)和本书(科学出版社, 2004 年 4 月第一版)出版至今已有 6 年多的时间, 这期间很多高等院校选择本书作为教材, 并根据教学经验反馈了宝贵的修订建议。在此基础上, 我们综合吸纳了各方面的意见, 及时修订推出了第二版教材。

第二版的教材修订是在第一版的基础上进行的, 在编写方针、内容叙述、形式风格上保持了与第一版的一致性。

与第一版相比, 第二版的修订工作主要体现在以下几个方面:

1. 章节组织。按照微型计算机的组成重新组织了章节结构, 删除冗余内容及实际应用中涉及较少的内容, 增加了近年发展较快的技术内容。

2. 侧重原理性。根据课程性质和有关院校教学反馈的信息, 第二版教材在技术原理性方面加大了说明力度, 以期学生从根本上掌握所学技术。本课程不是实训课程, 学生不但要掌握技术本身, 还要达到举一反三、“示之鱼, 授之渔”的目的。

3. 增加实例。从教学经验出发, 为了加强对技术内容的理解和掌握, 增加了实例演示。增加的实例力求与实验环节和实际科研项目相结合, 以期学生能在实验平台下实际操作, 加强学生对技术的兴趣和实践能力、动手能力。同时, 第二版教材也尽量避免复杂内容的项目实例, 避免学生因理解项目内容而转移对技术本身的注意力, 也避免因讲述复杂项目背景而浪费宝贵的教学学时。

4. 补充习题。要想较好地掌握所学技术, 就要做到“学而时习之”。多做习题是一个很好的辅助学习手段。第二版教材补充了精炼的例题和习题, 从教学角度看, 对学生的学习大有益处。

5. 为便于多媒体教学, 本教材配有电子教案, 有教学需求的教师可去网站下载 (<http://www.abook.cn>)。

在本书编写过程中, 参考了有关的优秀教材、专著、应用成果, 以及优秀的网络站点, 恕不能一一列举。在写作过程中能够领略众多新颖的观点和技术, 是原作者的无私贡献, 是读者的集粹之想。本书作者在此向提供各种观点和技术的各位编著者表示最真诚的谢意。

虽然作者从事计算机应用教学工作多年, 但也难免百密一疏, 加之水平所限, 书中差错和不当之处在所难免, 恳请读者斧正。

作 者

2010 年 5 月于吉林大学

第一版前言

微型计算机原理、汇编语言程序设计及接口技术三部分内容是计算机科学与技术、通信工程、电气工程及其自动化等专业的核心课程。在以前的教学体系中,大部分院校将其分成三门课,即“微型计算机原理及应用”、“微型计算机接口技术”和“汇编语言程序设计”。随着集成电路技术的飞速发展,许多大型计算机甚至巨型计算机的成熟技术已逐步下移至微型计算机,促使微型计算机发展非常快,随之带来两个问题:一是微型计算机的结构日趋复杂,这就使微型计算机原理、汇编语言程序设计及接口技术三部分内容彼此相关的程度更加密切、互相交融。二是新课程及新内容不断增加,每门课程的学时越来越少,使得旧的内容删不掉,新的内容又加不进来,于是出现了教学内容与实际严重脱节的现象,家用微型计算机早已使用奔腾(Pentium)微处理器,而课堂上仍在讲 Inter 8088/8086 微处理器。若仍将微型计算机原理、汇编语言程序设计及接口技术三部分内容分为三门课,势必造成在内容上时有冲突,有些内容学生不得不学两遍,甚至还要多,有时还会造成对某些问题或概念理解得不透彻。所以,改革目前微型计算机课程教学体系,把微型计算机原理、汇编语言及接口技术合为一体来讲授,势在必行。

本书将“微型计算机原理”、“微型计算机接口技术”和“汇编语言程序设计”三门课程的内容有机的融为一体。《微型计算机原理与汇编语言程序设计》和《微型计算机原理与接口技术》两本书为同一门课程连续使用的教材。它是在将三门课程合为一门(即“微型计算机原理、汇编语言、接口技术”)的三次教学实践基础上进行修改整理而成的,实际上也是我们二十几年来从事这三门课程的教学总结。本书以 Pentium 的实模式与保护模式为主线,用 Pentium 实模式的实现技术来替代 Intel 8086 的内容(目前流行以 Intel8086 为基础);通过分析 Pentium 的保护模式,把当今微型计算机领域内具有代表性的新设计、新技术、新思想和新潮流展示给读者;列举了一定数量的 I/O 接口硬件及程序设计实例,有助于建立微型计算机系统的整机概念,加深对微型计算机工作过程的理解,使学生初步具有微型计算机系统软、硬件开发的能力。

“微型计算机原理与汇编语言程序设计”共 8 章。第 1 章主要讲述微处理器发展简况,分别介绍第 1~4 代微处理器 Intel 8008、Zilog 的 Z80、Intel 8086、Intel 80386 的基本编程结构和功能特征。第 2 章主要介绍 Pentium~PentiumIV 微处理器的编程结构及功能,讲述总线接口、预取缓冲部件、整数流水线、浮点流水线、Cache 部件、指令译码部件、控制部件、分段部件、分页部件。超流水

线和超标量技术流水线结构、指令译码操作、寄存器重命名技术、乱序执行技术、退出流水线操作、饱和运算、积和运算能力、动态执行技术。Pentium 微处理器的引脚功能、Pentium 微处理器的基本时序（非流水线式读/写周期、突发式读写总线周期、流水线式读写总线周期）。第 3 章讲述在 16 位模式及 32 位模式的指令格式、寻址方式和指令系统。第 4 章讲述汇编语言程序格式、伪指令和汇编语言上机过程。第 5 章讲述双分支程序设计、多分支程序设计、循环程序设计的结构、循环程序设计方法和多重循环程序设计。第 6 章讲述子程序的结构、子程序的参数传递方法、子程序的嵌套与递归和子程序设计举例。第 7 章讲述宏汇编、重复汇编、条件汇编、模块化程序设计。第 8 章讲述半导体存储器的分类及性能指标、ROM 及 RAM 存储芯片、Pentium 的存储器接口、Pentium 的高速缓冲存储器（Cache）及二级 Cache 与一级 Cache 的关系等。

“微型计算机原理与接口技术”共 10 章。第 1 章主要讲述虚拟存储器、Pentium 分段存储管理和分页存储管理。第 2 章主要介绍为什么要用接口电路、I/O 接口的一般编程结构、CPU 与外设之间数据传送的控制方式（程序查询传送方式、程序中断方式、DMA 传送方式、I/O 处理机方式）、DMA 控制器 8237A 及其应用。第 3 章主要讲述中断的基本概念、中断接口电路、中断处理过程、Pentium 中断机制、实模式中断处理过程、保护模式中断操作和可编程中断控制器 8259A。第 4 章主要讲述总线的概念及分类、ISA 总线、PCI 总线。第 5 章讲述可编程并行输入/输出接口芯片 8255A、8255A 各种工作方式的应用举例、可编程计数器/定时器 8253 及其在计数和实时测频系统中的应用举例。第 6 章讲述数字串行通信系统模型、RS-232-C 串行通信接口总线、通用串行总线 USB 简介、可编程串行通信接口芯片 8251A、串行通信系统实例。第 7 章主要讲述实时微型计算机控制系统的硬件结构、传感器、数/模转换器及应用、模/数转换器及应用、功率开关器件及接口。第 8 章讲述键盘的结构、键的识别（行扫描法、行反转法）、微型计算机与键盘的接口、BIOS 键盘中断及 DOS 键盘功能调用。第 9 章讲述 CRT 显示器的工作原理、黑白字符显示器的基本原理、CRT 控制器、IBM PC 系列机的显示系统（MDA 适配器、CGA 适配器/EGA、VGA 适配器）、对显示适配器的编程。第 10 章讲述数据磁记录的基本原理、硬磁盘存储器类型、硬磁盘上信息的分布、硬磁盘驱动器、硬磁盘控制器、硬磁盘接口、磁盘文件存取技术（DOS 文件代号式磁盘存取、BIOS 磁盘文件存取）。

在本书的写作过程中得到了张长海、胡成全教授的大力支持与帮助，在此表示感谢！

由于作者水平有限，书中难免有错误和不当之处，恳请读者和同行专家批评指正。

作 者

2003 年 11 月于吉林大学

目 录

第二版前言

第一版前言

第 1 章 Pentium 保护模式存储管理	1
1.1 虚拟存储器及其工作原理	1
1.1.1 地址空间及地址	2
1.1.2 虚拟存储器工作原理	2
1.2 分段存储管理	3
1.2.1 分段存储管理的基本思想	3
1.2.2 段描述符	6
1.2.3 全局描述符表及寄存器	12
1.2.4 局部描述符表及寄存器	13
1.2.5 中断描述符表及寄存器	14
1.2.6 任务状态段及寄存器	15
1.2.7 段选择符及寄存器	19
1.3 保护模式下的访问操作与保护机制	20
1.3.1 保护机制的分类	20
1.3.2 数据段访问及其特权级检查	22
1.3.3 任务内的段间转移及其特权级检查	24
1.3.4 任务切换及其特权级检查	31
1.4 向保护模式的转换	37
1.5 分页存储管理	38
1.5.1 页目录与页表	38
1.5.2 分页转换机制	41
1.5.3 转换旁视缓冲存储器 TLB	43
1.6 段页式存储管理的寻址过程	44
1.7 虚拟 8086 模式	46
习题	47
第 2 章 输入输出	49
2.1 接口概述	49
2.1.1 接口与端口	49
2.1.2 接口的功能	50
2.1.3 接口的一般编程结构	51

2.1.4 接口的分类	54
2.2 I/O 端口的地址选择	55
2.2.1 输入输出的寻址方式	56
2.2.2 用门电路组合法进行端口地址选择	58
2.2.3 用译码器译码法进行端口地址选择	59
2.2.4 用比较器比较法进行端口地址选择	60
2.3 输入输出控制方式	61
2.3.1 程序查询方式	62
2.3.2 程序中断方式	67
2.3.3 DMA 方式	67
2.3.4 I/O 处理机方式	71
2.4 DMA 控制器 8237A	71
2.4.1 8237A 的内部结构及引脚功能	72
2.4.2 8237A 的工作方式	80
2.4.3 8237A 的工作时序	84
2.4.4 8237A 的编程	87
2.4.5 8237A 在 PC 机中的应用	92
习题	97
第 3 章 中断	99
3.1 概述	99
3.1.1 中断的基本概念	99
3.1.2 中断接口电路	101
3.1.3 中断处理过程	106
3.2 Pentium 的中断机制	110
3.2.1 中断向量表	110
3.2.2 可屏蔽中断 INTR	111
3.2.3 非屏蔽中断 NMI	112
3.2.4 软件中断	113
3.2.5 异常简介	114
3.2.6 实模式中断处理过程	116
3.2.7 保护模式中中断操作	119
3.3 可编程中断控制器 8259A	120
3.3.1 8259A 的内部结构及引脚功能	120
3.3.2 8259A 的工作方式	125
3.3.3 8259A 的编程	130
3.3.4 8259A 的应用举例	138
习题	139

第 4 章 总线技术	141
4.1 总线的基本概念	141
4.1.1 总线分类	141
4.1.2 总线标准的基本内容	143
4.1.3 总线的操作过程	143
4.1.4 总线的数据传输方式	144
4.2 IBM PC 总线	146
4.3 ISA 总线	149
4.4 PCI 总线	150
4.4.1 PCI 总线的特点	150
4.4.2 PCI 总线的系统结构	152
4.4.3 PCI 总线信号	153
4.5 STD 总线	157
4.6 主要外设总线介绍	159
4.6.1 USB 总线	159
4.6.2 IDE 总线	162
4.6.3 SCSI 总线	163
4.6.4 IEEE 1394 总线	166
4.6.5 AGP 总线	167
4.6.6 IEEE 488 总线	169
4.6.7 CAN 总线	171
4.6.8 Centronic 总线	173
习题	175
第 5 章 可编程接口芯片及其应用	176
5.1 可编程并行输入/输出接口芯片 8255A	176
5.1.1 8255A 的内部结构及引脚功能	176
5.1.2 8255A 的控制字	178
5.1.3 8255A 的工作方式	179
5.1.4 8255A 应用举例	184
5.2 可编程计数器/定时器 8253	191
5.2.1 8253 的基本功能	192
5.2.2 8253 的引脚信号与内部结构	192
5.2.3 8253 的控制字	194
5.2.4 8253 的工作方式	197
5.2.5 8253 的应用举例	210
习题	217

第 6 章 串行通信和可编程串行接口芯片 8251A	219
6.1 串行通信概述	219
6.1.1 数字通信系统模型	219
6.1.2 串行通信的传送方向	221
6.1.3 传输速率	222
6.1.4 异步通信与同步通信	223
6.2 RS-232-C 串行通信接口总线	225
6.3 可编程串行通信接口芯片 8251A	230
6.3.1 8251A 内部逻辑与工作原理	231
6.3.2 8251A 的引脚功能	233
6.3.3 8251A 的控制字	235
6.3.4 8251A 的应用举例	239
习题	242
第 7 章 模数转换及数模转换	245
7.1 概述	245
7.2 传感器	246
7.3 D/A 转换	249
7.3.1 D/A 转换原理	249
7.3.2 D/A 转换器的主要参数	251
7.3.3 D/A 转换器的输入输出特性	252
7.3.4 DAC 0832 转换器及应用	253
7.3.5 DAC 1210 与 CPU 的接口	258
7.4 A/D 转换	261
7.4.1 多路开关	261
7.4.2 A/D 转换原理	263
7.4.3 A/D 转换器的主要技术指标	271
7.4.4 ADC 0809 八位 A/D 转换器及应用	271
7.4.5 12 位 A/D 转换器 AD574	278
7.4.6 采样保持器	284
7.5 功率开关器件及接口	287
7.5.1 光电隔离器	287
7.5.2 功率晶体管驱动电路	288
7.5.3 机械继电器及接口	289
7.5.4 固态继电器	290
习题	290
第 8 章 人机交互接口	292
8.1 键盘接口	292

8.1.1 消除抖动及重键处理	292
8.1.2 线性键盘	294
8.1.3 矩阵键盘	296
8.1.4 键盘工作方式	301
8.1.5 PC 机键盘与接口	301
8.1.6 BIOS 键盘中断及 DOS 键盘功能调用	306
8.2 LED 显示器接口	309
8.2.1 LED 七段显示器结构	309
8.2.2 LED 显示器组成与显示方式	310
8.2.3 LED 显示器接口及应用举例	311
8.3 视频系统	316
8.3.1 CRT 显示器	317
8.3.2 液晶显示器	319
8.3.3 字符和图形显示的基本原理	322
8.3.4 显示器的主要性能指标	328
8.3.5 显示适配器	330
8.3.6 对显示器的编程	336
习题	341
附录 A BIOS 功能调用	344
附录 B 中断类型表	349
主要参考文献	351

第 1 章 Pentium 保护模式存储管理

1.1 虚拟存储器及其工作原理

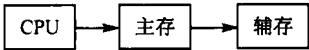
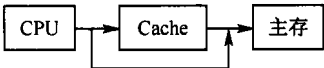
虚拟存储器（virtual memory）又称为虚拟存储系统。虚拟存储器是为满足用户对存储空间不断扩大的要求而提出的，随着用户程序复杂性的增加，占用存储空间越来越大，其解决办法是，可扩大主存，但是造价高，空间利用率很低，并不是好的途径。采用虚拟存储器，可较好地解决这个问题。

虚拟存储器这个概念是 1961 年由英国曼彻斯特大学的 Kilburn 等人提出的，并于 20 世纪 70 年代广泛应用于大中型计算机之中，现在的微型计算机也都采用了这种技术。

虚拟存储器是由主存储器、辅助存储器、辅助硬件和操作系统管理软件组成的一种存储体系。它把辅助存储器作为主存储器的扩充，对应用程序员来说，好像计算机系统有一个容量很大的主存。虚拟存储系统的目标是为了增加存储器的存储容量，它的速度接近于主存，单位造价接近于辅存，因此性能价格比很高。

虚拟存储器和 Cache 存储器是存储系统中两个不同的存储层次，它们在功能、结构、操作过程等方面的比较如表 1.1.1 所示。

表 1.1.1 虚拟存储器和 Cache 存储器的比较

存储体系	虚拟存储器	Cache 存储器
存储层次	主存-辅存	Cache-主存
主要功能	主存速度，辅存容量	CPU 速度，主存容量
信息传送单位	信息块（比如段、页），有多种划分，长度较大	信息块（比如块、行），长度较小，并且固定
结构差别	<pre>graph LR; CPU --> MM[主存]; MM --> AM[辅存]</pre> 主存不命中，进行辅存调度，而 CPU 程序换道	<pre>graph LR; CPU --> Cache; Cache --> MM[主存]; CPU --> MM</pre> 在 CPU 与主存之间具有直接访问通路
操作过程	由部分硬件和操作系统存储管理软件实现，对应用程序员透明，对存储管理软件程序员不透明	全部用硬件实现，对各类程序员透明

1.1.1 地址空间及地址

我们知道，CPU 只能执行已经装入主存的那一部分程序块，与此同时，为了提高主存的空间利用率，还应及时释放已不使用的信息在主存中所占用的空间，以便装入其他有用的信息。这样，随着程序的运行，各种信息就会在主存与辅存之间不断地调进、调出。

在虚拟存储器中有 3 种地址空间及对应的 3 种地址。3 种地址空间分别是虚拟地址空间、主存地址空间、辅存地址空间。

虚拟地址空间又称为虚存地址空间，是应用程序员用来编写程序的地址空间，与此相对应的地址称为虚地址或逻辑地址。

主存地址空间又称为实存地址空间，是存储、运行程序的空间，其相应的地址称为主存物理地址或实地址。

辅存地址空间也就是磁盘存储器的地址空间，是用来存放程序的空间，相应的地址称为辅存地址或磁盘地址。

1.1.2 虚拟存储器工作原理

虚拟存储器中，信息的调度和管理由硬件和软件（操作系统）共同完成，其工作过程如图 1.1.1 所示。

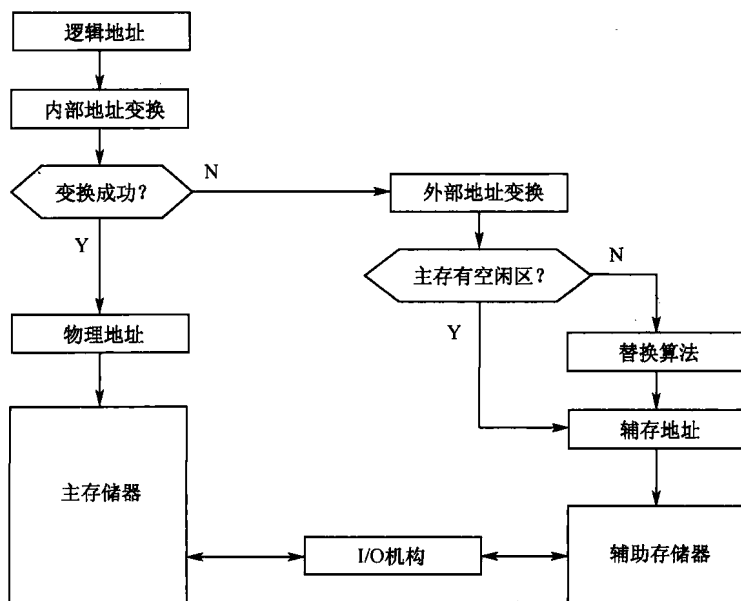


图 1.1.1 虚拟存储器工作过程示意图

应用程序访问虚拟存储器时,必须首先给出逻辑地址,然后进行内部地址变换,如果要访问的信息在主存中(也就是内部地址变换成功),则根据变换所得到的物理地址访问主存储器;如果内部地址变换失败,则要根据逻辑地址进行外部地址变换,得到辅存地址;与此同时,还需检查主存中是否有空闲区,如果没有,就要根据替换算法,把主存中暂时不用的某个(或某些)块信息通过 I/O 机构调出,送往辅存,再把得到的辅存地址中的信息块送往主存;如果主存中有空闲区域,则直接把辅存中有关的信息块送往主存。

由于采用的存储映像算法不同,就形成了不同的存储管理方式,其中主要有段式、页式和段页式 3 种。尽管使用的存储管理方式不同,但虚拟存储器的基本原理、工作过程及有关技术问题还是有许多相似之处。

Pentium 支持分段存储管理、分页存储管理和段页式存储管理。Pentium 微处理器片内存储管理部件负责对物理存储器实施安全可靠且行之有效的存储管理操作。当存储管理部件正常运转时,程序是不能直接对物理存储器进行寻址操作的,只能对一个被称之为虚拟存储器的存储器模型进行寻址操作。

Pentium 微处理机的存储管理部件由分段部件和分页部件组成。分段部件可以提供多个各自独立的地址空间,而分页部件可以使用少量的随机存储器(RAM)和磁盘存储器去支持一个很大的地址空间模型。Pentium 微处理机的分段部件和分页部件既可以单独使用其中的一种,也可以两种同时使用。

由程序提供的地址称作逻辑地址。分段部件的功能之一就是将逻辑地址转换成一个连续的不分段的地址空间,这个地址空间的地址称作线性地址。而分页部件的主要功能就是将线性地址转换成物理地址。

1.2 分段存储管理

1.2.1 分段存储管理的基本思想

通常,一个程序由多个模块组成,特别是在结构化程序设计思想提出之后,程序的模块性就更强了。一个复杂的大程序总可以分解成多个在逻辑上相对独立的模块,模块间的界面和调用关系是可以清楚定义的。这些模块可以是主程序、各种能赋予名称的子程序或过程,也可以是表格、数组、树、向量等某类数据元素的集合。模块的大小可以各不相同,有的甚至事先无法确定。但每一个模块都是一个特定功能的独立的程序段,都是以该段的起点为 0 相对编址。当某一程序段(模块)从辅存调入主存时,只要由系统赋予该段一个基址,就可以把基址和每个单元在段内的相对位移量组合起来,形成这些单元在主存中各自的实际地址。

把主存按段分配的存储管理方式就称为段式管理。程序进入内存时,各程序

段要求占据相对独立的内存区间。因此，现代微机系统把物理空间分成相对独立的许多内存段，每个内存段放置一个程序段，至此内存段与程序段统一，统称为段。一个程序拥有多个段、不同程序占据不完全相同的几个段，而且管理系统所需要的信息放置在属于系统所有的段中。

分段管理后，系统必须知道每个段的必要信息（段信息）才能完善地管理各段，这些信息包括：段在物理空间的开始地址、段的界限、段是数据型还是程序型，内存标志等多方面的内容。

32 位机把每个段的段信息放入一个数据结构中，称作段描述符（或简称描述符）。又把所有的段描述符分类，组成顺序排列的表，称作段描述符表（或简称段表）。每个段描述符表都放在内存中备用。显然每个段描述符表占据的物理空间也形成一个段。每个段描述符表也有自己的描述符，称作描述表描述符。

1. 分段存储管理工作过程

图 1.2.1 给出了分段存储管理的示意图。一个程序 A 具有 4 个模块，程序 A 对应的段描述符表中有 4 个段描述符，每个模块对应一个段描述符。段表的一行为一个段描述符，段描述符内容包括基址、界限和访问控制等。基址是装入模块的首地址，界限指出该段的长度。

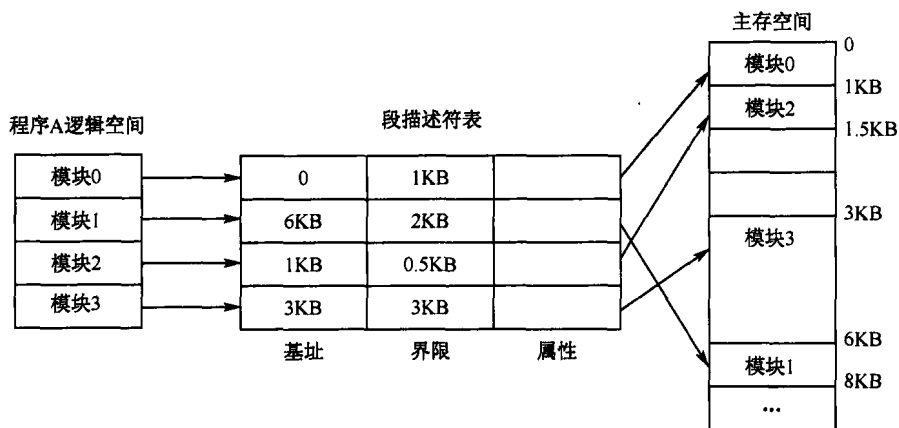


图 1.2.1 分段存储管理的示意图

由图不难看出，当需要访问某程序中的一个信息时，需要进行以下几个步骤：

1) 从内存中找到段描述符表。

2) 从段描述符表中找到相应的段描述符，由段描述符中的内存标志指出该段目前是否在内存。若内存标志不成立，系统就知道该段目前不在内存，系统去外存寻找此程序段，若找到就把它调入内存，然后修改段描述符以便与内存段

统一。

- 3) 从段描述符找到段在内存的位置。
- 4) 从段中找到信息所在内存的物理地址。
- 5) 访问该物理地址。

由此可知，只要系统建立了一个程序段的描述符，系统就开始管理此程序段，而无论该段内容是否真正在内存。也就是说，此标志位使系统可以把外存的一部分作为内存的延伸，与内存统一管理。这一复合存储空间称为虚拟内存。程序段只要进入虚拟内存就可以被系统管理，自动调入内存执行。因此程序员编程时使用虚拟空间即可，无需考虑物理空间大小。因此常称虚拟空间为编程空间。

2. 虚拟地址和虚拟地址空间

Pentium 微处理机在保护模式下的存储器管理单元使用 48 位的存储器指针，如图 1.2.2 所示。它分为段选择符（或简称为选择符）和偏移量两部分。该 48 位存储器指针称为虚拟地址，它在程序中用以规定指令或数据的存储器位置。段选择符 16 位，偏移量 32 位。段选择符可放在 Pentium 微处理机段寄存器中。若要访问存储器中的代码，则段选择符应放在 CS 中。若要访问存储器中的数据，则段选择符应放在 DS、ES、FS、GS、SS 中的任意一个。

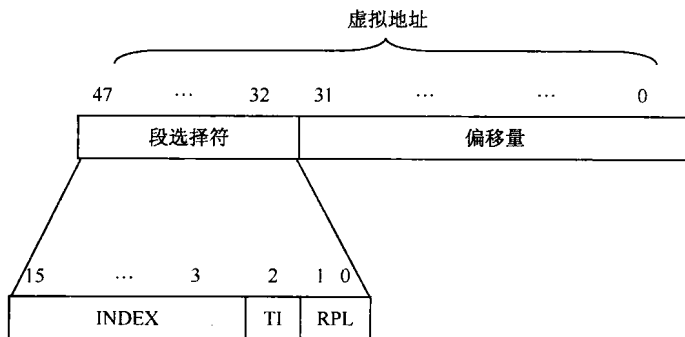


图 1.2.2 保护模式下的存储器指针及段选择符格式

偏移量放在 Pentium 微处理机的用户可访问的寄存器中。若要访问存储器中的代码，则偏移量放在 EIP 寄存器中。若要访问存储器中的数据，则偏移量放在 EAX、EBX、ECX、EDX、ESI、EDI 等寄存器中。由于偏移量是 32 位长，所以段大小可达 4G 字节。我们说段大小可达 4G 字节是因为段大小实际上可变的，它可从 1 个字节到 4G 字节。

16 位的段选择符分为 3 个字段：13 位索引字段，1 位表选择字段 TI 和 2 位的请求特权级字段 RPL。2 位的 RPL 并不用于存储器段选择，因此，16 位中只有 14 位用于寻址，这样虚拟地址空间可容纳 2^{14} (16K) 个存储器段，每个段最

大可达 4G 字节。这些段就是 Pentium 微处理机存储器管理部件所管理的虚拟地址空间的基本元素。

另外一种计算虚拟地址空间大小的办法是将 14 位段选择符和 32 位偏移量结合起来，得到 46 位虚拟地址，从而 Pentium 微处理机的虚拟地址空间可包含 2^{46} (64T) 个字节。

3. 虚实地址转换

Pentium 微处理机的分段存储管理机制允许将 46 位虚拟地址映射到硬件所需的 32 位物理地址。该地址转换大致过程如图 1.2.3 所示。首先由虚拟地址（也称逻辑地址）段选择符部分的 13 位索引字段确定段描述符在段表（也称段描述符表）的位置，然后取出段描述符中的 32 位基地址并与偏移量相加，得到 32 位的线性地址。如果不启用分页功能，则线性地址就直接作为物理地址。

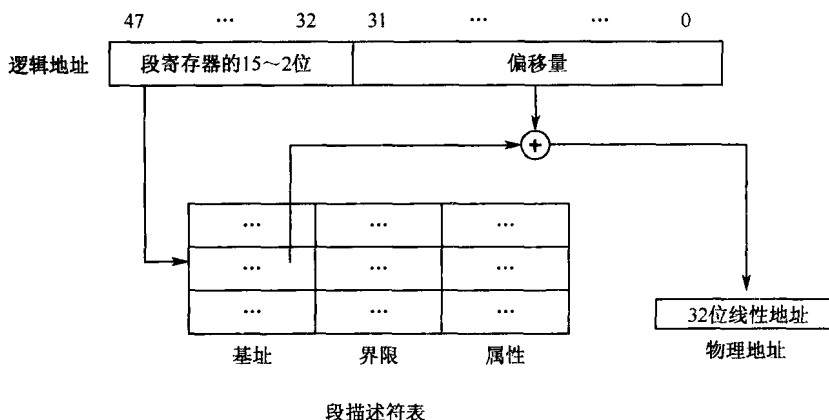


图 1.2.3 虚实地址转换示意图

1.2.2 段描述符

段描述符是 Pentium 微处理机存储管理部件用以管理 64T 字节虚拟存储地址空间分段的基本元素。一个段描述符对应于虚拟地址空间中的一个存储器段。

段描述符是位于主存中的一种数据结构，由系统程序创建，它为处理机提供段的基本信息。所有段描述符均由 8 个字节组成。段描述符内保存着供处理机使用的有关段的属性、段的大小、段在存储器中的位置以及控制和状态信息。一般说来，各段描述符都是由各种编译程序、各种连接程序、各种装入程序或者是操作系统产生的，而不是由各种应用程序生成的。

段描述符按段的性质可分为程序段描述符、系统段描述符和门描述符，如图 1.2.4 所示，其中程序段描述符又分为代码段描述符、堆栈段描述符和数据段