

CMP que

计算机硬件基础
与维修系列丛书

intel 系列微处理器 结构、编程和接口技术大全 —— 80X86、Pentium 和 Pentium Pro

(美) Barry B. Brey 著

陈 谊 等译

机械工业出版社 西蒙与舒斯特国际出版公司

计算机硬件基础与维修系列丛书

Intel 系列微处理器结构、 编程和接口技术大全 ——80×86、Pentium 和 Pentium Pro

(美) Barry B. Brey 著

陈谊 译

机械工业出版社
西蒙与舒斯特国际出版公司

本书全面、系统、深入地介绍了 Intel 系列微处理器的体系结构、程序设计和接口技术。全书共 17 章,详细讲解了 8086~Pentium Pro 微处理器的内部结构、指令系统、引脚功能、工作原理以及它们与存储器和 I/O 系统的接口技术;介绍了 8087~Pentium Pro 协处理器的数据格式、指令系统以及过程、宏、条件汇编、DOS 功能调用, BIOS 功能调用等高级汇编语言程序设计技术;还介绍了 ISA、VESA 局部总线和 PCI 总线接口技术。本书涉及到的外围器件有 82C55、8279、8254、16550UART、8259A、8237、8288 和 8289 等。附录 A 给出了较完整的 DOS 功能调用和 BIOS 功能调用说明。

本书可作为高等院校计算机和自动化专业本科生和研究生的教学参考书,也可作为从事 PC 系统开发及应用的工程技术人员的参考书和工具书。

Barry B. Brey: The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium Pro Processor Architecture, Programming, and Interfacing. Fourth Edition.

Authorized translation from the English Language edition Published by Prentice Hall.

Copyright 1997 by Prentice Hall.

All rights reserved. For sale in Mainland China only.

本书中文简体字版由机械工业出版社和美国西蒙与舒斯特国际出版公司合作出版,未经出版者书面许可,本书的任何部分不得以任何方式复制或抄袭。

本书封面贴有 Prentice Hall 防伪标签,无标签者不得销售。

版权所有,翻印必究。

本书版权登记号:图字 01-97-1253

图书在版编目 (CIP) 数据

Intel 系列微处理器结构、编程和接口技术大全: 80X86、Pentium 和 PentiumPro/
(美)布瑞 (Brey, B.B.) 著;陈谊等译.-北京:机械工业出版社,1997.12

(计算机硬件基础与维修系列丛书)

ISBN 7-111-06038-5

I. I… II. ①布… ②陈… III. ①微处理器, Intel-系统结构 ②微处理器, Intel-程序设计 ③微处理器, Intel-接口 IV. TP332

中国版本图书馆 CIP 数据核字 (97) 第 25632 号

出 版 人: 马九荣 (北京市百万庄南街 1 号 邮政编码 100037)

责任编辑: 蒋 克

北京昌平第二印刷厂印刷 • 新华书店北京发行所发行

1998 年 1 月第 1 版第 1 次印刷

787×1092mm 1/16 • 60.5 印张 • 1484 千字

印数: 0001-5000 册

定价: 93.00 元

凡购本书,如有倒页、脱页、缺页、由本社发行部调换

译 者 序

Intel 系列微处理器以其强大的功能已成为当前微型计算机的主流 CPU，被广泛应用于各个领域，并在以强大的势头继续向前发展。预计在未来几年中，Intel 系列微处理器仍将是微型计算机的主流 CPU。

当今，凡致力于微型计算机应用领域的人员都必须懂得汇编语言程序设计及接口技术。而微处理器的发展日新月异，谁最先掌握了新技术，谁就能在激烈的竞争中立于不败之地。本书为读者提供了最新的 Intel 微处理器的技术资料、全面、系统、深入地讲解了 Intel 8086/8088、80186/80188、80286、80386、80486、Pentium 和 Pentium Pro 微处理器的体系结构、程序设计和接口技术，并对该系列的各种产品进行了综合比较。全书内容深入浅出，全面系统并附有大量实例和习题，其目的在于帮助具有一定计算机基础的读者全面、系统地掌握 Intel 系列微处理器及其软、硬件应用，是一本非常有价值且实用的参考书，通过学习本书可使读者具备如下能力：

- 为控制与 8086/8088、80186/80188、80286、80386、80486、Pentium 和 Pentium Pro 微处理器的应用接口，开发控制软件。

- 使用汇编语言及 DOS 功能调用开发控制键盘、视频显示系统和磁盘存储器的程序
- 使用 BIOS 功能调用控制键盘、显示器及计算机系统其他部件
- 使用宏、过程、条件汇编和流程控制伪指令开发软件
- 使用中断接管和热键技术获取对内存驻留程序的访问权
- 对协处理器编程以解决复杂的数学运算问题
- 将小系统与 PC 系统中的 ISA、VESA 局部总线和 PCI 总线接口
- 将存储器、I/O 系统与微处理器接口

本书共 17 章，其中第 2~7 章由陈谊翻译，第 8~11 由湛云莉翻译，第 1、12~14 章由荀竄、刘卓翻译，第 15、16 章由谢小强翻译，第 17 章由赵丰年翻译，附录 A、B、C 由曹圻、郭凡、陈谊翻译，附录 D 由陈强、王蕊翻译。全书由陈谊统稿和修改。此外，傅豫波、施广老师对本书的翻译工作给予了大力支持和热情帮助，在此向他们表示衷心的感谢。

限于水平和时间，书中难免有遗漏之处，敬请读者批评指正。

陈 谊

1997 年 7 月

前 言

本书适用于需要全面掌握 Intel 系列微处理器编程及接口技术的读者。对于对这些技术感兴趣的人员来说，它是一本非常实用的参考书。当今，凡想在使用计算机的研究领域工作的人员都需要懂得汇编语言编程及接口技术。Intel 微处理器已在电子、通信、控制、特别是桌面计算机系统等许多领域中得到了广泛的应用。

1. 本书的组织形式及涉及内容

本书每一章的开始都提供了一个简要描述该章内容的概述，以便读者在学习前有个全面了解。接下来是该章的具体内容，其中包括许多说明其主题的编程实例。每章还给出了小结，它可作为该章的学习指导和复习要点。最后提供的思考与练习可帮助读者实践和练习该章表述的概念。

本书包含了许多采用 Microsoft MACRO 宏汇编规则编写的实例程序，由此使读者有机会学习对 Intel 系列微处理器编程。本书编程环境包括 linker、librarg、macros、DOS 功能调用、BIOS 功能调用等。

本书还对该系列中各微处理器、存储器系统以及磁盘存储器、ADC 和 DAC、16550 UART、PIAS、定时器、键盘/显示接口、协处理器、视频显示系统等各种 I/O 系统作了详尽的描述，对 PC 系统总成（ISA、VESA、PCI）进行了讨论。通过这些讨论，读者可以学到非常实用的微处理器接口技术。

2. 本书导读

由于 Intel 系列微处理器的种类繁多，故本书主要阐述对各型号都兼容的实方式编程技术，并对 80386、80486、Pentium 和 Pentium Pro 等 Intel 系列处理器与 8086/8088 的指令系统进行了比较。Intel 系列中各型号的微处理器非常相似，因此一旦读者理解了基本的 8086/8088 微处理器，就可学习掌握那些更高级的微处理器。请注意：8086/8088 以及它的升级产品嵌入式 80186/80188 和 80386EX 仍被用在控制器中。

本书除详细讲解了微处理器的编程和操作外，还讲解了协处理器（8087/80287/80387/80486/80487/Pentium/Pentium Pro）的编程和操作。协处理器的作用是在系统中提供浮点运算功能。浮点运算在诸如控制系统、视频图像、计算机辅助设计（CAD）等应用中非常重要。它能够实现普通微处理器难以实现的复杂算术运算。

本书还描述了 80×86 Pentium/Pentium Pro 微处理器的输出引脚及其功能。接口技术部分则先介绍 8086/8088 和一些通用外围部件的接口，然后将重点进一步转移到 80186/80188、80386、80486、Pentium 和 Pentium Pro 微处理器的接口技术上。由于 Intel 系列各微处理器都与 8086、80386 很相似，因此本书对 80286 叙述较少，以便能较详尽地叙述 80386、80486、Pentium 及 Pentium Pro。

通过学习微处理器和协处理器的操作和编程，以及 Intel 系列各微处理器的接口技术，

读者可掌握 Intel 系列微处理器的工作原理及应用技术。当完成本书的学习后, 读者能够:

- 开发用来控制 8086/8088、80186/80188、80286、80386、80486、Pentium、Pentium Pro 应用接口的控制软件。通常这些软件可在各种型号的微处理器上运行。这些软件中通常还包含基于 DOS 的应用。

- 使用汇编语言通过 DOS 功能调用来编写控制键盘、视频显示系统及磁盘存储器的程序。

- 使用 BIOS 功能控制键盘、显示器以及计算机系统其他部件。

- 使用宏指令序列、过程、条件汇编伪指令、流程控制伪指令开发软件。

- 使用中断接管和热键技术开发访问驻留内存程序的软件。

- 对协处理器 (80287/80387/80486/80487/Pentium/Pentium Pro) 编程, 以解决复杂的运算。

- 解释 Intel 系列各种处理器的区别、明确每一型号的特性。

- 描述并使用 80286、80386、80486、Pentium、Pentium Pro 微处理器的实方式和保护方式的工作原理。

- 设计存储器、I/O 系统到微处理器的接口。

- 能对 Intel 系列中各微处理器及其软件、硬件接口进行详细、全面的比较。

- 解释磁盘及视频系统的操作。

- 将小系统与 PC 系统中的 ISA、VESA 局部总线和 PCI 总线接口。

3. 内容一览

本书第 1 章以基于微处理器的计算机系统为重点, 介绍了 Intel 微处理器系列。包括微处理器的历史、操作和基于微处理器系统中的数据存储方式。并针对对数字系统缺乏了解的读者, 对数字系统做了简单介绍。第 2 章介绍了微处理器实方式和保护方式工作原理。

对计算机有了基本了解之后, 第 3 章到第 6 章阐明了 Intel 微处理器系列每条指令的功能。介绍指令的同时, 还提供了简单的应用程序来说明这些指令的操作, 并使读者逐渐建立程序设计的基本概念。

有了程序设计基础之后, 第 7 章提供了一些汇编语言应用程序。这些应用程序使用了 DOS 和 BIOS 功能调用以及鼠标器功能调用。同时, 对 PC 系统上的磁盘文件、键盘以及视频操作也进行了介绍, 这一章还提供了在 PC 系统上开发程序所需的工具, 介绍了中断接管和热键的概念。

第 8 章介绍了 8086/8088 微处理器, 作为后面几章理解内存和 I/O 基本概念的基础。这一章还阐述了缓冲系统和系统时钟。

第 9 章详尽地讲解了如何用译码器或可编程逻辑部件设计存储器接口。阐明了奇偶校验和动态存储器技术。为介绍 8086~Pentium Pro 各种微处理器与存储器的接口, 本章还提供了 8 位、16 位、32 位和 64 位存储器系统。

第 10 章通过讨论 PIA、定时器、键盘/显示器接口、16550 UART 及 ADC/DAC, 详细论述了基本的 I/O 接口技术。还介绍了与直流和步进电动机的接口。

在了解了这些基本的 I/O 部件和它们与微处理器的接口技术之后, 第 11、12 章详细介绍包括中断和直接存储器存取 (DMA) 的高级 I/O 技术, 给出了打印机接口、实时时钟、

磁盘存储器和视频系统的应用实例。

第 13 章详细论述了 8087~Pentium Pro 系列协处理器的操作和编程技术。如今，几乎没有不利用协处理器却能高效运行的应用程序。请记住，80486 及更高档的所有 Intel 微处理器内都有一个协处理器。

第 14 章讲述了如何通过 ISA、VESA 和 PCI 总线接口来建立小系统与 PC 的接口。本书中还包含了对许多设计用来将 PC 嵌入到工业控制系统中的接口卡的论述。

第 15、16 章论述了 80186/80188~80486 微处理器的细节；揭示了这些微处理器与 8086/8088 微处理器的区别，以及它们的增强功能特性；讨论了用于 80386 和 80486 微处理器系统中的 Cache 存储器，以及隔行（interleaved）与脉冲式（burst）存储器，还讨论了内存管理和分页技术。

第 17 章详细论述了 Pentium 和 Pentium Pro 微处理器。这两类新型处理器仍基于最初的 8086/8088 微处理器并使得 Intel 公司阔步迈向 21 世纪。

为增强实用性，本书包括以下四个附录：

（1）附录 A 列出了一个完整的 DOS INT 21H 功能调用表。此附录还详述了汇编程序的使用和许多 BIOS 功能调用，其中包括 BIOS INT 10H 功能调用。

（2）附录 B 列出了一个完整的 8086~Pentium Pro 各级微处理器的指令系统表，其中包括许多指令实例和十六进制机器代码以及指令的执行时间。

（3）附录 C 简要列出了所有能改变标志位的指令。

（4）附录 D 提供了每章后编号为偶数的思考与练习题的答案。

第 1 章 关于微处理器和计算机的介绍

本章概述

这一章将对 Intel 系列的微处理器作一个全面的介绍。其中包括计算机的历史和基于微处理器的计算机系统中微处理器功能的讨论。当然还包括计算机领域中的术语，以便当我们讨论微处理器和计算机时能使用计算机专业术语，并用读者能够理解这些术语。

系统框图和每一功能块的描述详细说明了一个计算机系统的运行过程。本章还介绍了 PC 的存储器和输入/输出系统是如何工作的。最后将讲述数据在存储器中的存储方式，以便读者编制软件时会运用每一种数据类型。数值型数据以整型、浮点型和二进制编码的十进制 (BCD) 形式存储，字母型数据则通过 ASCII 码 (美国标准信息交换码) 方式存储。

本章目的

当完成本章的学习后，读者将能够：

- (1) 使用相应的计算机术语，包括位、字节、数据、实存储器系统、扩充存储器系统 (EMS) 扩展存储器系统 (XMS)、DOS、BIOS、I/O 等等。
- (2) 简单介绍计算机的历史及应用。
- (3) 对于各种 80X86 及 Pentium/Pentium Pro 微处理器进行总体介绍。
- (4) 给出一个计算机系统的方框图，并对每一模块的功能作出解释。
- (5) 描述微处理器的功能，详述它的基本操作。
- (6) 定义 PC 机中存储器的内容。
- (7) 进行二进制、十进制、十六进制数据之间的转换。
- (8) 区分以整型、浮点型、BCD 码和 ASCII 码方式存储的数值和字母信息。

1.1 历史背景

这一节首先概述导致微处理器发展的历史事件，同时分别介绍功能极强的 80X86[○]和 Pentium/Pentium Pro[◎]微处理器。虽然对于历史的讨论并不是理解微处理器的必不可少的部分，但它却对飞速发展的微处理器作出历史的回顾，并展望它的未来。

1.1.1 机器时代

计算机系统的思想并不新颖——远在现代电子设备发明之前它就存在了。用机器运算的思想可以追溯到公元前 500 年，当时的巴比伦人发明了算盘 (abacus) ——第一种机械计算器。算盘用串起来的算珠完成运算，它被当时的巴比伦牧师用来看管他们不计其数的存粮。算盘一直被广泛地使用直至今日。直到 1642 年它才有一点小小的改动。当时的数学家 Blaise

○ 80X86 代表 8086、8088、80186、80188、80286、80386 和 80486 微处理器。

◎ Pentium 和 Pentium Pro 是 Intel 公司注册的商标，在本书中代表 Pentium 和 Pentium Pro 处理器。

Pascal 发明了用来运算的齿轮。每一齿轮有 10 个齿，当这 10 个齿转动一周，第二个齿轮就动一格。这和汽车计程器的机械原理相同，也是所有机械计算器的基础。巧妙的是，PASCAL 程序设计语言即是以 Blaise Pascal 的名字来命名的，以纪念他在数学和机械计算器领域中探索性的工作。

第一种实用齿轮式的机械计算机产生于 19 世纪的早期，它可以自动地计算信息。此发明的到来早在人类发明电灯泡之前，当时人们甚至对电还没有概念。在这个计算机的摇篮时期，人类梦想有一种用程序来计算数值的机器——不仅仅是用计算器来计算。

机械计算机的一位早期的先驱 Charles Babbage，得到了 Lovelace 伯爵夫人 Augusta Ada Byron 的支持。1823 年大不列颠皇家天文学协会授权 Babbage 发明一种可编程的计算机。此机器的用途为替皇家海军绘制航海图表。他接受了挑战，开始动手制造被他称为分析机 (Analytical Engine) 的机器。此机器存储了 1000 个 20 位长的十进制数字，以及一段能改变机器功能，完成各种运算任务的可更改的程序。他的机器通过一些穿孔卡进行输入，与五六十年代计算机使用的穿孔卡一样。大家普遍认为他是从法国人 Joseph Jacquard 那里继承了穿孔卡的思想，早在 1801 年 Jacquard 在他发明的纺织机上使用了穿孔卡，这架机器今天被称为 Jacquard 机。纺织机本身通过穿孔卡来选择复杂多样的布的图案。穿孔卡程序化了纺织机。

经过多年的努力，当 Babbage 意识到当时的机械师根本无法制造出他需要的部件时，他的热情开始衰退。分析机需要 5 万多个机械部件，没有一定的精度，他的机器根本不可能有效可靠地工作。

1.1.2 电子时代

19 世纪出现了电动机，与此同时出现了许多电动机驱动的机器，这些都建立在 Blaise Pascal 的机械运算器基础上。这些电动的机械计算器一直作为办公设备广泛应用着，直到本世纪 70 年代初 Bomar 发明了手持的小型电子计算器。另一位重要的电子计算器领域的先驱是 Monroe，但他们的机器都是桌面的，其尺寸像四功能型现金出纳机那般大小。

1889 年，Herman Hollerith 发明了存储数据的穿孔卡，正如 Babbage 一样，他也明显地借鉴了 Jacquard 的穿孔卡思想。他还发明了一个由电动机驱动的机械计算器——通过把信息存储在穿孔卡上完成计数、分类、纠正信息的功能。当时美国政府正迫切地想发明机械计算器，因此 Hollerith 奉命用他的穿孔卡系统来存储 1890 年的人口普查信息并绘制统计表。

1896 年，Hollerith 创建了一家名叫制表机的公司。该公司专门设计一系列用穿孔卡制表的机器，经过几次兼并，此制表公司变成了国际商业机器公司，现在更为通俗的叫法是 IBM 公司。为了纪念 Herman Hollerith，在计算机系统中使用的穿孔卡都叫做哈氏卡 (Hollerith card)。穿孔卡使用的 12 位码被称为哈氏码 (Hollerith Code)。

到 1941 年，德国发明家 Konrad Zuse 发明的第一台电子计算机出现之前，电动机驱动的机器一直占领着信息处理的市场。Konrad Zuse 的计算机——Z3 被二战中的德军用于飞机和导弹的设计。如果德国政府能给 Zuse 提供充足的资金的话，他很可能设计出一部功能更强的计算机系统。今天 Zuse 终于因为他在数字电子领域和 Z3 计算机系统（起始于 1930 年）的杰出成就受到赞誉。

最近发现（通过对英国军事文件的整理），第一台真正的电子计算机于 1943 年开始运作，目的在于破译德国军事密码。第一台使用真空管的电子计算机系统是由 Alan Turing 发

明的。他把他的机器称为“巨人”，大概是由它的庞大躯体而得名。“巨人”的问题在于虽然它能破译由谜机产生的军事密码，但却不能解决其他问题。“巨人”并非可编程的——它有一固化的程序系统，今天我们把这样的系统叫做专用计算机。

第一台通用可编程的电子计算机是于 1946 年由宾夕法尼亚大学设计制造的。第一台现代计算机叫做 ENIAC (Electronics Numerical Integrator and Calculator, 电子数字积分计算器)。它体积庞大, 使用了 1.7 万个真空管和 500 多米长的导线, 重达 30 多吨, 但每秒钟只能完成大约 10 万次操作。伴随着 ENIAC 的到来, 人类进入了电子计算机的时代。ENIAC 通过改装线路达到可编程的目的——该过程要花费许多工人数天的时间才能完成。工人改变插口板上的电路连接, 这看起来很像电话切换板。ENIAC 的另外一个问题是真空管元件的寿命有限, 需要经常进行维护。

紧跟着 1948 年贝尔实验室晶体管设计的成功, 1958 年 Jack Kilby 在得克萨斯州的实验室发明了集成电路芯片。此集成电路芯片导致了 60 年代数字集成芯片的发展 (RTL 或电阻晶体管逻辑电路), 以及 1971 年 Intel 公司第一个微处理器的产生。此时的 Intel, 及该公司的一名工程师 Marcian E. Hoff, 设计出了 4004 微处理器——它是微处理器历史上的一次革命, 今天微处理器正以持续增长的势头在发展着。

1.1.3 程序设计的发展

既然可编程机器出现了, 程序和程序设计语言应运而生。如上所述, 第一个可编程的电子计算机系统通过改装它的电子线路来实现编程。这种方法实际用起来很复杂, 因此在计算机系统发展的早期, 就产生了控制计算机的计算机语言。第一种计算机语言叫做机器语言 (machine language), 它把很多的 1 和 0 以二进制码的方式储存在计算机存储器中, 一段这样的代码叫做程序。这种方式虽然比改装机器的电子线路有更高的效率, 但编制程序仍然是非常复杂的, 因为还要涉及到很多代码。约翰·冯·诺依曼, 一位数学家, 第一个提出了一种接受指令并在内存中存储指令的系统。为了纪念他, 计算机经常被叫做冯·诺依曼机 (Von Neumann machines)。

本世纪 50 年代的早期, 紧随着 UNIAC I 和 II 这样的计算机的问世, 汇编语言的出现省去了人们以二进制码方式输入指令时遇到的许多麻烦。汇编语言允许编程人员用助记符, 如 ADD 来表示加法, 这样就取代了类似 01000111 这样的二进制码, 尽管汇编语言为编程提供了帮助, 但直到 1957 年 Grace Hopper 开发出了第一种高级程序设计语言——FLDW-MATIC 之后编制程序才变得容易起来。同年, IBM 公司提出了专门为它们的计算机设计的 FORTRAN 语言 (FoRmula TRANslator 公式翻译器程序语言)。FORTRAN 语言允许编程人员编程时用公式解决数学问题。直至今日此语言还被一些科学家使用着。FORTRAN 出现一年之后, 另外一种类似的语言 ALGOL (ALGO rithmic Language, 算法语言) 也问世了。

第一种成功并广泛地应用于商务领域的程序设计语言是 COBAL (COmputer Business Oriented Algorithmic Language, 面向商务的计算机算法语言)。尽管 COBAL 的使用近年来越来越少, 但在很多大规模的商业系统中它仍然发挥着重要作用。另外一种比较流行的商务语言是 RPG (Report Program Generator, 报表程序产生器), 它允许通过说明输入、输出和运算类型的方式来编制程序。

在这些语言产生之后, 另外一些语言也相继出现。比较普及的有 BASIC, C/C++, PASCAL 和 ADA 语言。BASIC 和 PASCAL 都被指定为教学专用语言, 但它们的应用范围大

大超越了课堂，广泛地使用于许多计算机系统中。BASIC 语言是所有语言中最容易学习的，有人估计 PC 中 80% 的用户程序是由 BASIC 来书写的。最近，BASIC 的一个新版本 VISUAL BASIC 问世了，这样在 Windows 环境下编程就容易多了。VISUAL BASIC 也许最终会取代 C/C++ 和 PASCAL。

在科学界，C/C++ 和 PASCAL 常用于控制程序，这两者，特别是 C/C++，允许编程人员几乎完全控制编程环境和计算机系统。在许多情况下，C/C++ 取代了一些早期是由汇编语言编制的低级机器控制软件。尽管如此，汇编语言仍然在编制程序中发挥重要作用。大多数为 PC 编写的视频游戏几乎都是专门用汇编语言来完成的。为了更有效地控制机器和发挥机器功能，汇编语言常常和 C/C++、PASCAL 一起使用，进行混合编程。

ADA 语言在军事防御中得到广泛使用。ADA 语言的命名是为了纪念 Augusta Ada Byron, Lovelace 伯爵夫人，她和 Charles Babbage 一起在 19 世纪初期发明了分析机。

1.1.4 微处理器时代

世界上第一个微处理器，Intel4004，是一个 4 位可编程控制的芯片——按今天的标准它可能是微不足道的。它的寻址空间仅限于 4096 个 4 位宽的存储单元 [一位 (bit) 指二进制赋值为 0 或 1 的一个数值位，一个 4 位宽的存储单元经常被称作半字节 (nibble)]。4004 的指令系统只包括 45 种指令。它是用当时流行的 P 沟道 MOSFET 技术制造的，因此只能以 50KIPs (每秒千条指令) 那样的低速执行指令。这比起 1946 年重达 30t 的 ENIAC 的 100KIPs 的速度自然低很多，但 4004 的特色是它的重量不到 1 盎司 (1 盎司 = 28.35g)。

最初，4004 的用途很广泛，它最初用来制做视频游戏系统以及一些小的基于微处理器的控制系统。其中一个早期的游戏，Shuffleboard，是由 Baller 设计的。这个早期的微处理器在速度、字长以及存储容量上都存在问题。Intel 公司设计出的 4040 算是对早期 4004 的一次改进。尽管字长和存储容量方面没有改进，4040 的运行速度却有所提高。其它一些公司，尤其是得克萨斯设备仪器公司 (TMS-1000)，也生产出一些 4 位微处理器。现在一些小的控制系统如微波炉中仍然使用 4 位微处理器。有一些微处理器芯片生产商还在生产它们。大多数计算器仍然使用 4 位微处理器来处理 4 位 BCD (二进制编码的十进制数) 码。

1971 年底，Intel 公司意识到微处理器是一个能带来利润的产品，因此又投资生产了 8008——8 位的微处理器。8008 可在扩充的存储空间编程 (16KB)，同时还包括其他一些附加的指令 (总共 48 条)，因此为它在更高级系统中的应用奠定了基础。[一个字节 (byte) 通常指 8 位宽的二进制数，1KB 指 1024 个字节。存储容量一般都是用 KB 表示的]。

工程技术人员对 8008 微处理器的改进提出了迫切的要求，他们发现它的存储容量太小，速度还是很慢，指令系统也有限，这些都限制了它的应用。Intel 公司认识到了这些缺陷，因此在 1973 年又推出了 8080 微处理器——第一个现代的 8 位微处理器。大约半年之后，Motorola 公司也推出了它的 MC6800 微处理器。8080 和 MC6800 开辟了微处理器历史的新纪元。很快地，其他一些公司也相继推出它们的 8 位微处理器产品，表 1-1 列出了这些早期的微处理器芯片和它们的生产厂家。在这些生产厂商中，只有 Intel 公司和 Motorola 公司成功地继续发展它们的事业，并不断推陈出新。Zilog 公司还在生产微处理器，但基本是在幕后，它决定集中力量发展微控制器和嵌入式控制器，而不是通用微处理器。Rockwell 公司已经完全放弃了微处理器的生产，转到生产调制解调器上去了。

1.8080 究竟特殊在哪里

8080 不仅扩充了内存容量和指令系统，它的速度也比 8008 提高了 10 倍。在 8008 芯片（每秒 5 万条指令）上做加法需 20 μ s，而在每秒执行 50 万条指令的 8080 芯片上只需 2.0 μ s。而且，8080 可以直接和 TTL（晶体管－晶体管逻辑电路）相兼容；而 8008 无法做到。这就使接口更容易且价格更便宜。8080 的寻址范围（64KB）是 8008（16KB）的 4 倍。这些改进都为微处理器的发展与繁荣奠定了基础。碰巧 1974 年第一部 PC（MITS Altair 8800）也问世了（注意选择 8800 可能是为了避免有盗窃 Intel 公司版权的嫌疑）。该电脑的 BASIC 语言翻译器是由 Bill Gates 编写的——他是 Microsoft 公司的奠基人。该电脑使用的汇编程序由数字研究（DR）公司编制，此公司现在专门为 PC 开发 DR－DOS。

表 1-1 早期的 8 位微处理器

生 产 商	部 件 号	生 产 商	部 件 号
Fairchild	F-8	National Semiconductor	IMP-8
Intel	8080	Rockwell International	PPS-8
MOS Technology	6502	Zilog	Z-8
Motorola	MC6800		

2.8085 微处理器

1977 年，Intel 公司又发布了 8080 的升级产品——8085。这是 Intel 公司生产的最后一种 8 位通用微处理器。尽管它比 8080 只有少量的改进，但它运行软件的速度还是提高了很多。在 8080 上做加法（每秒 50 万条指令）需 2.0 μ s，现在 8085 只需 1.3 μ s（每秒 769230 条指令）。8085 的主要优势在于它的内部时钟产生器、内部系统控制器以及更高的时钟频率。这些高级元件的组合减少了 8085 的造价，提高了它的使用价值。Intel 公司售出了 1 亿个 8085 微处理器，再加上由其他合法的生产厂商生产的 8085，比如说 AMD 公司（Advanced Micro Devices，超微设备公司），总共生产的 8085 大约达到 2 亿个。含有 8085 的电器至今仍被使用，并且很可能将来也很受欢迎。Zilog 公司也售出了 5 万个 8 位微处理器——Z80 微处理器。执行 8085/Z80 机器语言代码的微处理器超过了 7 亿。

1.1.5 当今的微处理器

1978 年，Intel 公司又研制出了 8086 微处理器，大约一年后，8088 也问世了。这两种设备都是 16 位微处理器，执行一条指令的时间还不到 400ns（2.5MIPs，即每秒 2500 万条指令）。比起 8085，这自然是很大的进步。此外，8086 和 8088 的可寻址空间为 1MB，是 8085 的 16 倍（1MB=1024KB，即 1048576 字节）。更高的速度和更大的内存允许 8086 和 8088 在许多应用中取代了小型机。8086/8088 的另外一个显著特征是它使用了一个很小的 4 或 6 字节的指令高速缓冲存储器（cache），在指令执行之前预先取出进行排队。这种做法加快了指令的执行速度，为现在的采用更大的 cache 技术奠定了基础。

8086/8088 内存容量的扩展和指令系统的扩充为微处理器的广泛使用提供了前提条件。扩充的指令系统中包括乘除法指令，这在早期的微处理器上是没有的。指令的数量由 4004 的 45 条上升到 8085 上的 246 条，一直增加到 8086/8088 上的 20000 多条。注意，这些微处理器叫做 CISC（complex instruction set computer，复杂指令系统计算机），因为指令系统比

较复杂。复杂指令系统虽然数量大也比较难于掌握，但使用起来更有效。16 位的微处理器提供比 8 位微处理器更多的内部寄存器。这样软件编制的效率就提高了。

16 位微处理器是朝着更大内存容量的方向发展的。1981 年，IBM 公司决定在他们的 PC 上采用 8088 微处理器，这样 Intel 公司的知名度就提高了，地位也更加稳固。像文字处理器、拼写纠错器、电子表格以及计算机字典都要求大容量内存，8 位微处理器可寻址的 64KB 内存显然不能满足它们的需要。而 16 位的 8086/8088 可寻址空间的 1MB。很快地，这 1MB 内存感觉也太有限了。因此 Intel 公司又于 1983 年推出 8086 的升级产品——80286。

1. 80286 微处理器

80286 微处理器除了寻址空间变为 16MB 之外，80286（也是 16 位）几乎和 8086/8088 完全一样。而它的指令系统只附加了一些控制那另外的 15MB 内存的少数几条指令。它的时钟频率也增加了，因此执行某些指令的速度甚至不到 250ns (4.0MIPs)。此外，还有一些其他方面的改进，比起 8086/8088，指令的执行速度几乎增加了 8 倍。

2. 32 位微处理器

应用对微处理器提出了更高的要求，即更快的速度、更大的内存和更宽的数据通路。因此 80386 于 1986 年由 Intel 公司正式推出。80386 是对 16 位的 8086—80286 微处理器的彻底改进。它是 Intel 公司的第一个实用的 32 位微处理器，它的数据总线和内存地址都是 32 位的（注意以前 Intel 公司也生产了 32 位微处理器 iapx-432，但并不成功）。通过 32 位的总线，80386 可寻址空间达到 4GB (1GB=1024MB，即 1073741824 字节)。4GB 的内存能存储 100 万张双面打印纸上的数据。80386 也有其他一些版本，如 80386SX，寻址空间为 16MB，含 16 位数据总线和 24 位地址总线。此外还有 80386SL/80386SLC，寻址空间为 32MB，含 16 位数据总线和 25 位地址总线。80386SLC 包含一个内部高速缓冲存储器以便于高速处理数据。1995 年，Intel 公司推出了 80386EX，也叫嵌入式 PC，它在一个集成芯片上封装了 AT 类 PC 的所有部件，它还有 24 根输入/输出数据线、26 位的地址总线、16 位的数据总线、一个 DRAM 刷新控制器，以及可编程的芯片选择逻辑。

使用 GUI (graphical user interface, 图形用户界面) 的软件系统要求微处理器有更高的速度和更大的内存，当今的图形画面经常包含 256000 或更多的图形元素 (像素点)。最简单的 VGA (Variable Graphics Array, 可变图形阵列) 的分辨率为每扫描行 640 个像素点，总共有 480 个扫描行。为了显示一屏信息，每一像素点都要作变动。这就要求有高速微处理器。很多软件包都使用这种形式的图形界面。这些基于 GUI 的软件包要求微处理器和图形适配器更快速有效地处理图形文本和数据。其中比较高级的系统是微软公司的 Windows，我们经常称 GUI 为 WYSIWYG (What you see is what you get, 所见即所得) 显示。

32 位微处理器，由于它的数据总线宽度适于传送需要 32 位宽内存的实型 (单精度浮点) 数，而受到欢迎。为了有效处理 32 位实型数，微处理器必须能把它高速地从微处理器本身传送到内存。如果它们是通过 8 位数据总线传送的，那么需要 4 个读写周期。而通过 32 位数据总线，则只需一个读写周期。显然这大大提高了程序处理实型数的速度。大多数高级语言、电子表格和数据管理系统都使用实型数来存储数据。实型数在图形设计软件包中也得到应用，因为这些软件用向量来设计屏幕上的图形，例如 AUTOCAD、ORCAD 等 CAD (computer aided drafting/design, 计算机辅助设计) 系统。

除了提供更高速的时钟外，80386 还包括一个内存管理单元，它允许操作系统分配和管

理内存资源。早期的微处理器把内存管理的任务完全留给了软件。80386 包含了专门管理内存和分配内存的硬件电路，由此提高了效率，省去了完成这一功能的软件。

80386 的指令系统和早期 8086、8088、80286 的指令系统是向下兼容的，附加的指令涉及到 32 位的寄存器，还可以管理内存系统。注意 80286 使用的内存管理指令和技术同 80386 也是兼容的。这些特征允许早期的 16 位的软件能在 80386 微处理器上使用。

3.80486 微处理器

1989 年，Intel 公司推出了 80486 微处理器，它包括了类似 80386 的微处理器，类似 80387 的协处理器和 8KB 的高速缓冲存储器，并且把这些都集成在一起。虽然 80486 和 80386 没有根本上的区别，但它还是有一个显著的改进。80486 把 80386 的内部结构做了修改，大约有一半的指令在一个时钟周期内完成，而不是原来的两个。因为 80486 的主频有 50MHz，所以一半的指令在 25ns 内执行完毕（50MIPs）。在同一时钟速度下，执行典型的混合指令的平均速度在 80486 上大约比在 80386 上提高了 50%。后来的 80486 执行指令的速度更高，采用了 66MHz 的双时钟（80486DX2）。它的内存存取速度为 33MHz。Intel 公司的三时钟微处理器，80486DX4，内部执行速度达到 100MHz，内存存取速度为 33MHz。注意 80486DX4 执行指令的速度和 60MHz 的 Pentium 处理器相同。它还把早期 80486DX4 的 8KB 高速缓冲存储器扩充为 16KB。最近，AMD 公司推出了三时钟微处理器，其总线速度为 40MHz，时钟速度为 120MHz。将来一定会研制出主频高达 250MHz 的微处理器。

其它版本的 80486 叫做 OverDrive[○]处理器。它事实上就是用于取代 80486SX 或低速 80486DX 的双时钟 80486DX。当它插到插座上，可以取代 80486SX 或 80486DX，功能上相当于一双时钟的微处理器。例如，如果一个以 25MHz 工作的 80486SX 被一 OverDrive 微处理器取代，它就相当于内存传送速度为 25MHz 的一个主频为 50MHz 的 80486DX2。

表 1-2 列出了许多由 Intel 公司和 Motorola 公司生产的微处理器，并给出它们的字长和内存容量。其它一些公司也生产微处理器，但没有一家公司能比得上 Intel 公司，就连 Motorola 公司也比不上。

4.Pentium 微处理器

1993 年问世的 Pentium 类似于 80386 和 80486 微处理器。最初它被称为 P5 或 80586，但是 Intel 公司决定不采用数字命名，原因是保护版权很困难。最初的两种 Pentium 处理器中，其中一种主频为 60/66MHz，指令执行速度为 110 MIPs，另外一种主频为 100MHz，105 个时钟，以 150MIPs 的速度执行指令。双时钟的 Pentium，运行速度为 120MHz 和 133MHz，而更高级的版本为 200MHz。另外一个区别是高速缓冲存储器由 80486 的 8KB 增加到 16KB。Pentium 包括一个 8KB 的指令缓冲存储器和 8KB 的数据缓冲存储器。这允许传送大批数据的程序时也能受益于高速缓冲存储器技术。内存系统达到 4GB，数据总线宽度由 80386 和 80486 的 32 位增加到 64 位。根据 Pentium 版本的不同，数据总线的传送速度可为 60MHz 或 66MHz。更宽的数据总线允许采用双精度浮点数来进行高速向量产生图形的显示。同时它还能以更快的速度在内存和微处理器之间传送数据。这将允许虚拟现实在基于 Pentium 的平台上以更接近现实的速度实现。Pentium 拓宽后的数据总线和更高的执行速度

○ OverDrive 是 Intel 公司注册的商标。

同时还允许扫描速度为 30Hz 或更高的全帧图象显示，这可以和商业电视相比。

表 1-2 现代 Intel 和 Motorola 微处理器

生 产 商	部 件	数据总线宽度	内 存 大 小
Intel	8048	8	2KB 内部
	8051	8	8KB 内部
	8085A	8	64KB
	8086	16	1MB
	8088	8	1MB
	8096	16	8KB 内部
	80186	16	1MB
	80188	8	1MB
	80251	8	16KB 内部
	80286	16	16MB
	80386EX	16	64MB
	80386DX	32	4GB
	80386SL	16	32MB
	80386SLC	16	32MB + 1KBcache
	80386SX	16	16MB
	80486DX/DX2	32	4GB + 8KBcache
	80486SX	32	4GB + 8KBcache
	80486DX4	32	4GB + 16KBcache
	Pentium	64	4GB + 16KBcache
	Pentium OverDrive (P24T)	32	4GB + 16KBcache
Motorola	Pentium Pro processor	64	64GB + 16KBL1 cache + 256KBL2cache
	6800	8	64KB
	6805	8	2KB
	6809	8	64KB
	68000	16	16MB
	68008Q	8	1MB
	68008D	8	4MB
	68010	16	16MB
	68020	32	4GB
	68030	32	4GB + 256cache
	68040	32	4GB + 8KBcache
	68050	32	曾提出，但从未公开过
	68060	64	4GB + 16KBcache
	PowerPC	64	4GB + 32KBcache

最近，Intel 公司推出了人们等待已久的 Pentium OverDrive (P24T)，它以 63 或 83MHz 的主频进行工作。63MHz 是对以前 80486DX2 的 50MHz 的改进，而 83MHz 是对 80486DX2 的 66MHz 的改进。改进后的 83MHz 微处理器以介于 66MHz 和 75MHz Pentium 的速度工

作。如果说早期的 VESA 局部总线图象和磁盘缓冲控制器造价太贵的话, Pentium OverDrive 代表了一种从 80486 到 Pentium 的理想升级途径。

也许 Pentium 最有创意的特征是它的双整数处理器技术。Pentium 可以同时独立地执行两条指令。因为它内部有两个独立的整数处理器, 这种技术称为超标量技术。它允许 Pentium 每一时钟周期内都能执行两条指令。另外一个提高性能的特征是它采用了跳跃预测技术, 加速了包括循环语句的程序的执行。像 80486 一样, Pentium 也有一个内部浮点协处理器, 它可以以高于 80486 五倍的速度处理浮点型数据。这些特点都使 Intel 公司的微处理器颇受欢迎。同时它们还使 Pentium 取代了在一个时钟周期内执行指令的 RISC 机 (reduced instruction set computer, 精简指令系统计算机)。注意, 有一些新的 RISC 处理器一个时钟内也可以执行一条以上的指令, 因为它们采用了超标量技术。Motorola、Apple 和 IBM 公司最近推出了 PowerPC, 它是由两个整型单元和一个浮点型单元组成的 RISC 微处理器。PowerPC 虽然是对 Apple Macintosh 的改进, 但目前和 Intel 公司的微处理器比较起来还是速度慢些。最近, 通过执行 DOS 和 Windows 功能对 PowerPC 进行软件测试证明它比 80486SX 25MHz 的微处理器还要慢些。因此, Intel 微处理器在 PC 机中还处于遥遥领先的地位。注意当今大约有 4 百万台 Apple Macintosh[○], 而使用 Intel 微处理器的机器超过 1.6 亿台。

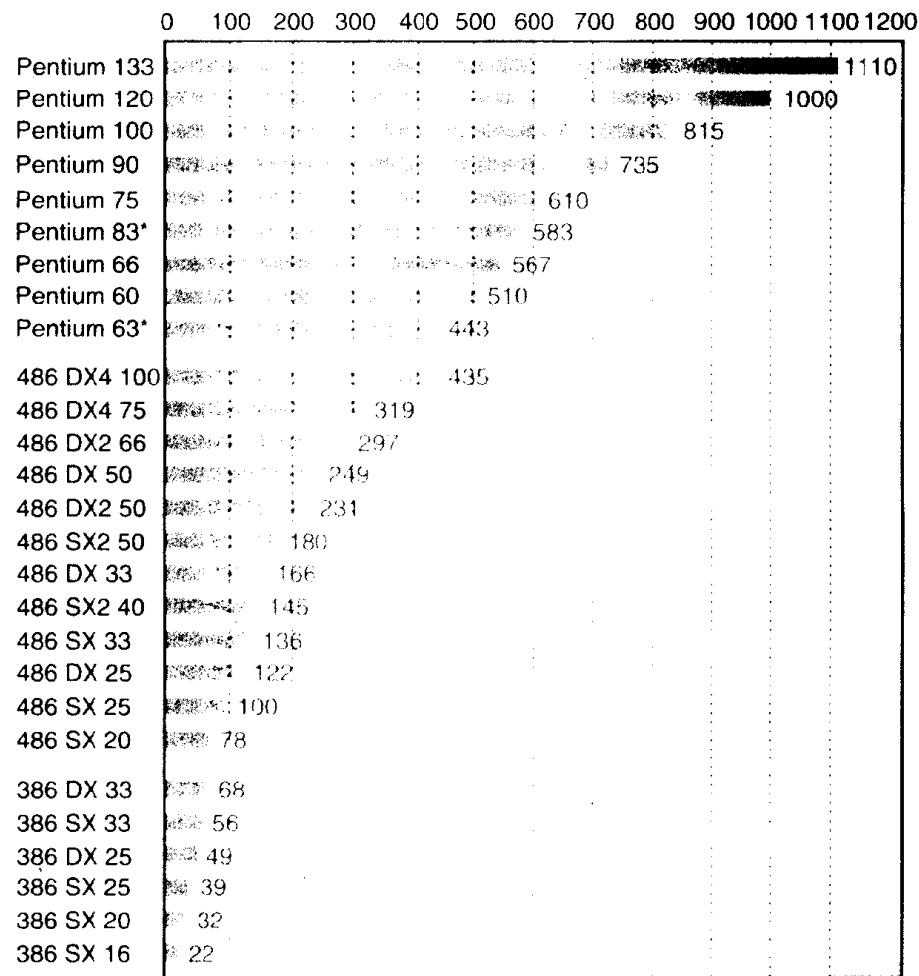
为了比较各种不同的微处理器的速度, Intel 公司提出一种叫做 iCOMP 速率指数的方法。该指数由 SPEC92、ZD Bench 和 Power Meter 组成。从图 1-1 可以看出 80386DX 25MHz 在图的下端, 而 Pentium 133MHz 在图的最上端。

5. Pentium Pro 微处理器

Intel 公司的最新产品 Pentium Pro 微处理器, 原来被称为 P6 微处理器。它有 2100 万个晶体管, 3 个处理整数的单元, 一个用于提高大多数软件执行速度的浮点型单元。1995 年底生产的产品基本时钟频率为 150/166MHz。另外, 内部还有 16KB 第一级 (L₁) 的高速缓冲存储器 (8KB 存放数据, 8KB 存放指令)。它还有一个 256KB 的第二级 (L₂) 高速缓冲存储器。另外, 一个比较显著的变化是它有三个指令执行部件, 自然同时能执行三条指令, 若三条指令有冲突的话, 仍然能并行执行。这和不发生冲突时才能并行执行两条指令的 Pentium 自然不同。通过优化, Pentium Pro 可以高效率地处理 32 位代码。因此它装载了 Windows NT, 而不是普通版本的 Windows 95。另外一个区别在于它既能在 4GB 内存空间寻址, 也能在 64GB 空间寻址。如果是后者的话, 它采用了 36 位的地址总线。

没有人能准确地预测将来, 但 Intel 系列产品的繁荣至少还要持续几年。很有可能朝 RISC 技术方向发展, 但更可能朝 Intel 和 Hewlett-Packard 联合研制的一种新技术方向发展。即使是这种新的技术也会毫无疑问地体现 80×86 微处理器的 CISC 指令系统, 以便于该系统的软件都能继续发挥作用。这种技术的基本思想是许多微处理器都能直接和别的微处理器交流, 不需要修改指令系统和程序就能做到并行处理。现在的超标量技术使用了许多微处理器, 但它们都共享同一寄存器组。这种将用于 P7 上的尚未研制成功的技术包含了许多微处理器, 并且每一微处理器都有自己的寄存器组, 且和别的微处理器寄存器相连通。这种技术不需要专门编制软件就能真正地并行处理。

[○] Macintosh 是 Apple 计算机公司注册的商标



注: * = Pentium OverDrive

图 1-1 Intel 的 iComp 指数

1.2 基于微处理器的 PC 机系统

近来的计算机系统发生了许多变化。由于微处理器的存在，原来占据相当大的空间的机器被微缩成办公桌上的计算机系统。虽然这些系统都是压缩的，但其处理和计算数据的功能在几年之前是想都不敢想的。80 年代耗资百万的巨型机其功能还不如当今的 80486、Pentium 和 Pentium Pro 强大。事实上，许多小型的公司正用基于微处理器的计算机系统取代他们的巨型机。类似 DEC (Digital Equipment Corporation, 数字设备公司) 这样的厂家都不再生产巨型机了，以便于能专门从事基于微处理器计算机系统的生产。

本节将给出基于微处理器 PC 机系统的结构。包括内存和操作系统知识。

图 1-2 给出了 PC 机系统框图。该图适用于早至巨型机晚至微机的任何计算机。该图由三个模块组成，每模块之间由总线相互连接起来 (总线是传送同一种类型信息的通用连线，例如地址总线，包括 20 根或更多根连线，把内存地址送到内存)。PC 机中的这些模块和它们的功能将在本节中概述。