

汇编语言与接口技术

李元章等编著



北京理工大学出版社



普通高等教育“十三五”规划教材

汇编语言与接口技术

李元章 王娟 张全新 编著
张华平 谭毓安



 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

版权专有 侵权必究

图书在版编目 (CIP) 数据

汇编语言与接口技术 / 李元章等编著. —北京: 北京理工大学出版社, 2018.10
ISBN 978-7-5682-6426-6

I. ①汇… II. ①李… III. ①汇编语言-程序设计②微型计算机-接口技术
IV. ①TP3

中国版本图书馆 CIP 数据核字 (2018) 第 231170 号

出版发行 / 北京理工大学出版社有限责任公司
社 址 / 北京市海淀区中关村南大街 5 号
邮 编 / 100081
电 话 / (010) 68914775 (总编室)
(010) 82562903 (教材售后服务热线)
(010) 68948351 (其他图书服务热线)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 /

开 本 / 787 毫米×1092 毫米 1/16

印 张 / 24.25

字 数 / 570 千字

版 次 / 2018 年 10 月第 1 版 2018 年 10 月第 1 次印刷

定 价 / 53.00 元

责任编辑 / 王玲玲

文案编辑 / 王玲玲

责任校对 / 周瑞红

责任印制 / 李 洋

图书出现印装质量问题, 请拨打售后服务热线, 本社负责调换

汇编语言与计算机接口技术是各高等学校计算机及信息类专业学生必修的专业基础课程。汇编语言的显著特点是可以直接控制硬件并充分发挥计算机硬件的功能,对于编写高性能的系统软件和需实时响应的应用软件具有不可忽视的作用;微机接口技术则侧重于计算机硬件的结构和 I/O 系统的组成,是实现高性能的存储与网络设备、提升计算机硬件性能所必须研究与解决的核心技术。

随着高等学校教学改革的深入,将汇编语言与微机接口技术结合起来,以 IEEE/ACM CS2013 课程体系为指导,融合为一门新的计算机基础专业课程,已经为许多高校所采用。为了满足新形势下计算机与信息类专业课程建设和教学内容改革的需求,作者在多年承担汇编语言、微机接口技术教学实践的基础上,积极跟踪汇编语言与计算机接口技术的最新发展,按照高等学校课程体系的要求,对标国际一流大学及 CS2013 课程体系,在对国内外汇编语言与接口技术教材进行充分调研的基础上,结合在相关科研项目中所取得的研究成果,进行了本教材的编写。

教材的编写在强调理论的基础上,着力引导学生进行实践,激发学生的主动性及创新能力。与国内现有同类教材比较,本教材的特色及创新包括以下几点:

1. 教材内容新颖

教材着眼于 32 位计算机的汇编语言与接口技术,大量增加了反映现代计算机领域内的先进技术的教学内容,体现计算机硬件技术的升级换代,淘汰了同类教材中比较陈旧的内容。在汇编语言程序设计方面,教材主要内容和实例程序以 Windows 操作系统及保护模式为主,对于学生了解现代计算机内部运行机制和操作系统细节、训练学生掌握最新的程序设计和调试技术等都具有重要的作用。在接口技术方面,去除传统的 ISA、IDE 等数据传输接口内容,增加了 USB3.0、Wi-Fi 等新的接口类型。相比于同类教材,本教材新增及加强的主要内容包括保护模式的原理及编程模型、多核技术、浮点运算寄存器结构及程序设计、高级可编程中断机制 APIC、USB 总线技术及应用等。

2. 组织结构合理

在组织形式上,教材编写过程中放弃了传统的知识点简单堆叠、汇编指令的简单罗列及芯片手册似的技术介绍,在结合典型应用的基础上,将汇编语言和接口技术融合,在对 CPU、硬件底层充分了解的基础上掌握汇编语言程序设计的基本方法和接口技术的基本原理,培养学生的硬件组成与设计思维,强调并介绍最新接口技术,并用汇编语言实现其典型应用。教材的编写遵循循序渐进的原则,注重从理论基础到实践应用的过渡,以微处理器、总线、外围接口为核心内容,将汇编语言与 PCI、USB、DMA、中断机制、保护模式、无线接口等具体应用紧密结合,在宏观上勾画出计算机硬件系统架构的同时,结合具体的程序设计技术,

使学生对现代计算机系统具有更加全面深入的了解。

3. 理论与实践结合

本教材贯穿了理论、实践应用于一体的思想，重点突出了理论与实践相结合。本书的大部分实例内容来自编者多年的科研总结和项目成果，是学生口中“有用的”技术，能够极大地激发了学生的学习兴趣。本教材在编写过程中，注重将计算机中的各种复杂抽象的原理实例化，在介绍功能及概念之后，结合实例说明它们的应用，通过这种方式，使读者在获得知识的同时，还能够学会灵活地运用这些知识。书中通过屏幕截图、运行实例程序等手段，将抽象的计算机部件的运行机制以易于理解的形式展现出来，便于学生学习和掌握。

4. 开放式的实验环境

汇编与接口课程是实践性比较强的课程，需要通过实验加强学生对现代微机更深层次的理解，并且提高学生的动手能力。但是我们通过调研发现，大部分同类教材须依赖于某种实验箱或者实验设备才能搭建实验环境，使实验过程受到限制。同时，不同高校间由于实验设备的不同，不方便开展交流，且微机实验设备的更新换代需要大量投资等，为了解决这些问题，本教材采用开放式的实验环境和配套实验，只要有微机，就能开展接口实验，摆脱了实验箱的限制。本教材提供的实验均可以在基于 Intel 80x86 系列的微机上进行，利用最接近实用的技术，采用目前主流微机的接口，直接在微机主板上开展实验，既具有最大的广泛性，有利于推广，又具有最大的实用性，能极大地激发学生的学习兴趣。

参加本书编写的均为多年工作在教学和科研第一线、有着丰富经验的教师，本书适合用作高等院校计算机与信息类专业的课程教材。全书由李元章组织编写，参加本书各章节包括习题和实验设计与编写的还有王娟、张全新、张华平和谭毓安。李元章、谭毓安统阅了全稿。

本书在编写过程中得到了北京理工大学教务处的大力支持，在此表示诚挚的谢意，并对所参考的国内外教材和资料的创作者致以衷心感谢。

由于计算机技术发展迅速，再加上作者水平有限，书中难免会有不足之处，殷切希望得到广大同仁和读者的批评指正。

编 者

目 录

CONTENTS

第 1 章 微型计算机硬件系统	001
1.1 微处理器	001
1.1.1 微处理器概述	001
1.1.2 80x86 系列 CPU 发展	002
1.1.3 CPU 的微结构	005
1.1.4 微处理器性能指标	006
1.1.5 微处理器软件特性	008
1.1.6 多核技术	011
1.2 主板	013
1.2.1 主板结构	013
1.2.2 芯片组	014
1.2.3 主板插槽	018
1.2.4 外部接口	021
1.3 内存	024
1.3.1 基本概念	024
1.3.2 存储器访问	027
1.4 扩展卡	029
习题 1	030
第 2 章 微处理器管理模式	031
2.1 微处理器的基本结构	031
2.2 CPU 工作模式	032
2.2.1 实模式	033
2.2.2 保护模式	033
2.2.3 虚拟 8086 模式	034
2.2.4 64 位 CPU 的工作模式	034
2.3 寄存器	035
2.3.1 程序可见寄存器	035
2.3.2 保护模式下的寄存器	039
2.4 内存管理	044
2.4.1 实模式下分段管理	044

2.4.2 保护模式下分段管理	045
2.4.3 页式内存管理	051
2.5 任务	056
2.5.1 任务执行环境	057
2.5.2 任务状态段	057
2.5.3 门	060
2.5.4 任务切换	062
2.6 保护	066
2.6.1 数据访问保护	066
2.6.2 对程序的保护	068
2.6.3 输入输出保护	071
习题 2	073
第 3 章 指令系统	076
3.1 数据寻址方式	076
3.1.1 CPU 操作数寻址	076
3.1.2 存储器操作数寻址	077
3.2 数据运算指令	081
3.2.1 数据传送指令	081
3.2.2 算术运算指令	086
3.2.3 位运算指令	095
3.3 程序控制指令	098
3.3.1 转移指令的寻址方式	098
3.3.2 转移指令	102
3.3.3 循环指令	105
3.3.4 子程序调用与返回指令	107
3.3.5 中断调用与返回指令	109
3.4 处理机控制指令	110
3.4.1 标志操作指令	111
3.4.2 常用处理机控制指令	111
3.5 块操作指令	111
3.5.1 块操作指令格式	111
3.5.2 块操作指令示例	113
习题 3	115
第 4 章 汇编语言程序开发	118
4.1 汇编语言基本知识	118
4.1.1 汇编语言概述	118
4.1.2 汇编语言编程环境	119
4.1.3 汇编语言语句格式	121
4.2 常用伪指令	121

4.2.1 数据定义伪指令	121
4.2.2 符号定义伪指令	123
4.2.3 操作符伪指令	124
4.2.4 框架定义伪指令	126
4.3 汇编源程序格式	128
4.3.1 用户界面	128
4.3.2 控制台界面的汇编源程序	129
4.3.3 Windows 界面的汇编源程序	135
4.3.4 输入/输出有关的 Windows API 函数	136
4.4 分支与循环程序设计	139
4.4.1 分支程序设计	139
4.4.2 循环程序设计	146
4.5 浮点运算	152
4.5.1 浮点数的表示与存储	152
4.5.2 浮点寄存器	154
4.5.3 浮点指令及其编程	158
4.6 程序优化	163
4.6.1 运行时间优化	163
4.6.2 占用空间优化	167
习题 4	167
第 5 章 子程序设计	169
5.1 子程序基本知识	169
5.1.1 子程序定义	169
5.1.2 堆栈	170
5.1.3 子程序的返回地址	172
5.2 参数传递	173
5.2.1 C 语言函数的参数传递方式	174
5.2.2 汇编语言子程序的参数传递方式	176
5.2.3 带参数子程序的调用	177
5.2.4 子程序中的局部变量	179
5.3 子程序的特殊应用	181
5.3.1 子程序嵌套	181
5.3.2 子程序递归	181
5.3.3 缓冲区溢出	182
5.4 模块化程序设计	184
5.4.1 模块化设计基本概念	184
5.4.2 模块间的通信	185
5.5 C 语言模块的反汇编	187
5.5.1 基本框架	187

5.5.2 选择结构	188
5.5.3 循环结构	189
5.5.4 变量定义	190
5.5.5 指针	190
5.5.6 函数	191
5.6 C语言和汇编语言的混合编程	193
5.6.1 直接嵌入	193
5.6.2 C程序调用汇编子程序	193
5.6.3 汇编调用C函数	196
5.6.4 C++与汇编的联合编程	197
习题5	197
第6章 存储系统与技术	198
6.1 高速缓冲存储器	198
6.1.1 Cache工作原理	198
6.1.2 Cache一致性协议	201
6.2 内部存储器	203
6.2.1 内存分类	203
6.2.2 主要技术指标和参数	204
6.2.3 内存模组	205
6.3 辅助存储器	212
6.3.1 硬盘概述	213
6.3.2 HDD原理和主要技术指标	213
6.3.3 ATAPI标准	214
6.3.4 ATA接口的编程模型	218
6.3.5 ATA设备寄存器	221
6.3.6 硬盘读写方式	226
6.3.7 串行ATA	233
6.3.8 固态硬盘	235
习题6	237
第7章 总线技术	238
7.1 总线概述	238
7.1.1 总线的分类	238
7.1.2 总线技术指标	239
7.2 PCI总线	240
7.2.1 PCI总线特点	240
7.2.2 PCI总线的体系结构	240
7.2.3 PCI总线引脚信号定义	241
7.2.4 PCI总线命令	244
7.2.5 PCI总线协议	246

7.2.6 PCI总线数据传输过程	248
7.2.7 总线仲裁	250
7.2.8 PCI总线配置	251
7.3 PCI-E总线	253
7.3.1 PCI-E概述	253
7.3.2 PCI-E的协议层次	254
7.4 USB总线	255
7.4.1 USB的起源和发展	255
7.4.2 USB接口的硬件特性	258
7.4.3 USB OTG技术及其扩展	264
7.4.4 USB通信协议	269
7.5 I ² C总线	273
7.5.1 I ² C概述	273
7.5.2 I ² C接口访问EEPROM	277
习题7	278
第8章 接口技术	279
8.1 串行接口及应用	279
8.1.1 串行通信概述	279
8.1.2 可编程串行通信接口	288
8.2 定时与计数技术	303
8.2.1 定时与计数概述	303
8.2.2 可编程定时器芯片	304
8.2.3 微机系统中的定时	319
8.3 红外	323
8.3.1 红外技术概述	323
8.3.2 IrDA协议分析	324
8.3.3 IrDA建立连接的过程	325
8.4 Wi-Fi	326
8.4.1 WLAN的组成	326
8.4.2 WLAN的结构	327
8.4.3 IEEE 802.11协议	328
习题8	330
第9章 中断技术	332
9.1 中断概述	332
9.1.1 中断基本原理	332
9.1.2 中断和异常	332
9.1.3 中断服务程序	336
9.2 实模式的中断处理	336
9.2.1 中断向量表	336

9.2.2 中断处理过程	337
9.2.3 写中断向量表	337
9.3 保护模式的中断处理	338
9.3.1 中断描述符表	338
9.3.2 中断和异常的处理过程	339
9.4 可编程中断控制器 8259	344
9.4.1 内部结构	344
9.4.2 8259 中断过程	346
9.4.3 8259 的级联	347
9.4.4 8259 的编程	348
9.4.5 8259 在 PC 中的应用	356
9.5 高级可编程中断控制器	357
9.5.1 APIC 概述	357
9.5.2 LAPIC	358
9.5.3 IO APIC	365
习题 9	367
附录 Visual Studio 2017 编写汇编语言程序步骤	369
参考文献	377

第 1 章

微型计算机硬件系统

微型计算机系统包括计算机硬件系统和计算机软件系统。硬件系统是 CPU、内存、外部设备等实体部分，软件系统是指计算机硬件系统上运行的程序集合。一般书上常见的硬件划分方式是从组成计算机功能模块的角度出发，将冯·诺依曼型计算机的硬件分成五大部件，包括运算器、控制器、存储器、输入设备和输出设备。本书为了更直观、更切合实际地描述计算机硬件，给出微型计算机硬件系统组成，如图 1-1 所示。

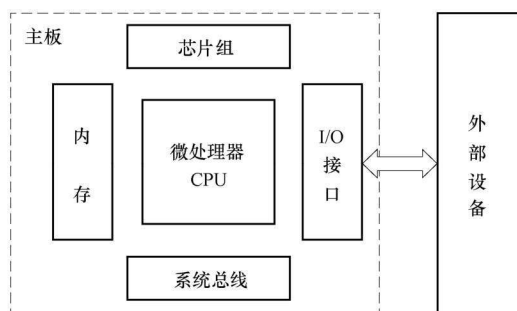


图 1-1 微机系统的硬件组成

图 1-1 中主要包括主板（含芯片组、内存、微处理器、系统总线、I/O 接口）和外部设备。本章主要讨论微处理器、主板、内存和 I/O 接口，系统总线和外部设备接口见第 7 章和第 8 章内容。

1.1 微处理器

微处理器（Microprocessor）简称 μP 、MP 或 MPU（Microprocess Unit）。微处理器是微型计算机的核心，它的性能决定了整个计算机的各项关键指标。MPU 是采用大规模和超大规模集成电路技术将算术逻辑单元 ALU（Arithmetic Logic Unit）、控制单元 CU（Control Unit）和寄存器组（Registers）三个基本部分及内部总线集成在一块半导体芯片上构成的电子器件。微处理器又称为“中央处理单元”（Central Processing Unit），简称 CPU。自 20 世纪 80 年代以来，微型计算机大都采用 Intel 公司设计制造的 CPU 或 AMD 等公司的其他兼容产品。

1.1.1 微处理器概述

自 20 世纪 70 年代微处理器诞生以来，除了用于 PC、笔记本电脑、工作站及服务器上的

通用微处理器 (General-purpose Microprocessors) 外, 还包括一些专用的微处理器 (Dedicated Microprocessors)。专用微处理器面向特定的应用, 包括单片机 (Single Chip Computer) 和数字信号处理器 (Digital Signal Processor, DSP) 等。单片机又叫微控制器 (Micro Controller Unit, MCU), 常用于控制领域。芯片内部除了 CPU 外, 还集成了其他如 RAM/ROM、输入/输出接口等主要部件, 一个芯片配上少量的外部电路和设备就能构成具体的应用系统。比较著名的包括 Intel 公司的 8 位 MCS51 系列、16 位/32 位的 MCS96/98 系列及其他各大处理器厂商基于 ARM 架构的各系列处理器。数字信号处理器专注于数字信号的高速处理, 这类微处理器主要在通信、消费类电子产品领域使用, 比较知名的是美国德州仪器公司 (Texas Instruments, TI) 的系列产品, 如 TMS320 系列各代 DSP 等。

通用微处理器和专用微处理器基本原理大同小异, 技术上相通, 应用在不同的领域, 各具特色。本书主要讨论通用 Intel 80x86 系列微处理器的基本原理和接口, 其基本原理可以推广适用于其他类型的应用系统。

1.1.2 80x86 系列 CPU 发展

自 20 世纪 70 年代开始出现微型计算机以来, CPU 经历了飞速的发展。而 Intel 公司也逐步取得了巨大的成功, 成为这个领域的“霸主”。

1. 4 位微处理器

1971 年, Intel 成功设计了世界上第一个微处理器: 4 位微处理器 Intel 4004。它有 45 条指令, 执行速度为 50 KIPS (Kilo Instructions Per Second), 即每秒执行 5 万条指令。直到今天, 由于其良好的性能价格比, 4 位微处理器仍然应用于一些嵌入式系统, 比如微波炉、洗衣机、计算器等。

2. 8 位微处理器

1972 年, Intel 公司推出了世界上第一款 8 位处理器 8008, 8008 可以支持到 16 KB 的内存。整体性能比 4004 有了较大的提升。基于 Intel 4004 和 Intel 8008 诞生了第一代的微机 MCS-4 和 MCS-8 微机。1973 年, Intel 推出的 8080, 执行速度达到 500 KIPS, 寻址范围达到 64 KB。1974 年, 基于 Intel 8080 的个人计算机问世。微软 (Microsoft) 公司创始人 Bill Gates 为这种 PC 开发了 BASIC 语言解释程序。1977 年, Intel 推出了 8085, 基于 8085 的微机执行速度达到 770 KIPS。这一时期的典型微机产品还有 Motorola 公司的 M6800、Zilog 公司的 Z80 等。

3. 16 位微处理器

1978 年 6 月, Intel 8086 问世, 它是 Intel 公司的第一个 16 位 CPU。1979 年 6 月, Intel 推出了 Intel 8088, 执行速度为 2.5 MIPS (Million Instructions Per Second)。8086 和 8088 都是 16 位 CPU, 能够进行 16 位数据的运算和处理, 寻址范围达到 1 MB。它们的主要区别在于外部数据总线的宽度, 8086 的外部数据总线为 16 位, 而 8088 为 8 位。当时与微处理器配套的外围接口电路大都是 8 位, 因此, 尽管 8086 的数据传输能力要强于 8088, 但是 8088 的兼容性更好。8088 在市场上获得了极大的成功, IBM 选择 8088 作为 CPU, 在 1981 年 8 月推出了它的第一代个人计算机 IBM PC。自此, Intel 逐步确立了 PC 行业的 CPU 霸主地位。同时, Microsoft 公司的 MS DOS 被 IBM 公司采用后, 经过多年的经营, 成为软件领域的“巨无霸”。

8086/8088 微处理器只能支持整数运算, 浮点运算通过转换为整数指令来完成, 即“浮点

仿真”。为了提高浮点运算的速度，Intel 公司在 1976 年推出了数字协处理器 8087，它能够在 8086/8088 的控制下执行浮点运算指令，进行复杂的数学运算，大幅度提升运算速度。

1982 年，Intel 公司在 8086 的基础上，研制出了 80286 微处理器，该微处理器的最大主频为 20 MHz，内、外部数据传输均为 16 位，使用 24 位地址总线，内存寻址范围达到 16 MB。在 Intel 系列 CPU 中，80286 首次引入了保护模式。兼容 8086/8088 的运行模式称为实模式，而 80286 可以运行在实模式和保护模式下。

4. 32 位微处理器

1986 年，Intel 推出了 80386，它的数据总线和地址总线都是 32 位的，内部寄存器和操作也都是 32 位，能够进行 32 位数据的运算和处理，它的寻址范围达到 4 GB。80386 首次引入了虚拟 8086 模式，它可以运行在实模式、保护模式和虚拟 8086 三种模式。与 80286 相比，80386 的保护模式功能更强，支持内存分页机制。到目前为止，Windows 和 Linux 操作系统都运行在保护模式下，分页机制是这些操作系统实现虚拟内存所必需的硬件环境。

1989 年，Intel 公司推出了 80486，它集成了 80386、80387 和 8 KB 片内高速缓存（Level 1 Cache，也称 L1 Cache）。80387 是与 80386 配套的浮点协处理器。80486 的运行速度和处理能力比 80386 有了大幅度的提高，80486 推动了图形用户界面 GUI（Graphic User Interface）的广泛应用。从 80486 开始，Intel 采用了倍频技术，CPU 主频（处理器工作频率）可以设置为外频（系统总线工作频率）的若干倍，从而使 CPU 的工作频率远远高于其外围电路。

1993 年 3 月，Intel 推出新一代奔腾（Pentium）CPU。在此之前，Intel 以 80x86 来命名其 CPU，数字命名不能得到商标的保护，所以生产兼容 Intel CPU 的 AMD、Cyrix 等公司也采用与 Intel 相同的数字命名。从奔腾开始，Intel 不再使用数字来命名。

Pentium 不是 64 位 CPU，尽管它的外部数据总线为 64 位，一次内存总线操作可以存取 8 字节的数据，但 Pentium 内部的寄存器和运算操作仍然是 32 位，地址总线仍然为 32 位。Pentium 体系结构中包括了两个整型处理单元和一个浮点协处理单元，内设两条指令流水线，这种超级标量技术（Super Scalar）在每个时钟周期可并行执行两条 32 位的指令，并且通过动态转移预取技术，保证了流水线操作的连续性。因此 Pentium 被称为准 64 位 CPU。

Pentium 有许多不同主频的版本，如 60 MHz、66 MHz、75 MHz、90 MHz、100 MHz、120 MHz 等。66 MHz 的 Pentium 的指令执行速度为 110 MIPS。Pentium 的片内高速缓存为 16 KB，数据高速缓存（Data Cache）和指令高速缓存（Instruction Cache）各占 8 KB。

1995 年 11 月，Intel 推出的 Pentium Pro，地址总线为 36 位，寻址范围达到 $2^{36}=64$ GB。其片内高速缓存有两级，分别为 L1 和 L2，L2 缓存为 256 KB 或 512 KB。

1997 年 1 月，Intel 发布了 Pentium MMX 指令集，增强了对多媒体数据的支持。

1997 年 5 月，Intel 推出 Pentium II（也称 P II），它的主频达到 233~400 MHz。片内高速缓存为 32 KB，数据 Cache 和指令 Cache 各占 16 KB。它的 L2 缓存为 512 KB，但没有包含在 CPU 内部，采用新的封装形式 SECC（Single Edge Contact Cartridge，单边接触盒）来连接 P II 和 L2 缓存。P II 不是一个芯片，而是一个多芯片模组，包括：P II CPU、L2 芯片组及电阻电容等配套电路。这几个部件被放置在一个电路板上，密封在一个保护壳的盒子里。Intel 为 P II 的插座和插槽申请了专利（Slot1），目的是避免兼容厂商仿制。1998 年，Intel 为进入低端市场，推出了赛扬（Celeron）处理器，它是 P II 的简化版，去掉了它的 L2 缓存，因此其性能比 P II 的低，但价格低廉。

1999年2月, Intel 推出 Pentium III 处理器(也称 P III), 主频为 450 MHz、500 MHz。P III 具有一个流式指令扩展 SSE (Streaming SIMD Extensions, SIMD 即 Single Instruction Multiple Data, 单指令多数据流) 的指令集, 全面增强三维图形运算, 也对动画、图像、声音、网络、语音识别等功能进一步增强。P III 芯片内部都有一个 128 位处理器序号, 每个 CPU 的序号是唯一的。这个序号的设计目的是识别用户, 提高网上电子商务的安全性。用户可将这个功能关闭。

2000年11月, Intel 推出 Pentium IV 处理器(也称 Pentium 4 或 P4), 主频为 1.4 GHz。系统总线速度达到了 400 MHz, 指令流水线达到 20 级, 增加了 SSE2 指令集, 进一步增强了多媒体、网络等密集运算能力。

2002年, Intel 在 CPU 中加入了超线程技术 HT (Hyper-Threading)。之后主频逐步提高, 达到了 3.2 GHz。

2004年2月, Intel 发布了 Pentium 4 E 系列处理器, 90 nm 制造工艺, 800 MHz 系统总线频率, CPU 名称也改为数字命名, 主要有 P4E 580/570/560/550/540/530/520, 对应频率为 4.0 GHz /3.8 GHz /3.6 GHz /3.4 GHz /3.2 GHz /3.0 GHz /2.8 GHz。

Intel 系列的 CPU 至今仍是桌面计算机的主流。采用 Intel CPU 制造的计算机, 被称为 IA (Intel Architecture) 架构。内部寄存器和运算位数为 32 位的 Pentium 系列 CPU 统称为 IA-32 (Intel Architecture-32)。8086、80286、80386、80486、Pentium、P II、P III、P4 的一系列 CPU, 被统称为 80x86 系列, 也称为 x86 系列。这些 CPU 保持了兼容的特点, 即后推出的 CPU 的指令系统完全覆盖了以前推出的 CPU 指令系统, 因此各种已有软件可以在新推出的 CPU 上运行。

5. 64 位微处理器

由于 IA 结构的计算机在高端市场(大型服务器系统)中所占比例不够理想, 其他的 RISC CPU 在 20 世纪 90 年代早期就发展了 64 位的 CPU, 占据了高端市场的主流地位。为了在企业服务器与高性能运算市场上占据一席之地, HP (Hewlett-Packard, 惠普) 公司与 Intel 自 1994 年开始共同研发基于 IA-64 (Intel Architecture 64) 的 64 位 CPU, 并于 2001 年推出了第一款 64 位的 Itanium(安腾)微处理器。Itanium 的微结构基于显式并行指令计算(Explicitly Parallel Instruction Computing, EPIC), 由编译器来决定哪些指令并发处理。该架构彻底不同于其他 Intel 处理器采用的 x86 (包含 x86-64) 架构, 也不与 x86 架构兼容, 在市场上没有取得很大的成功。

与 Intel 公司抛开 IA-32 而发展新的 64 位 CPU 不同, 一直生产 Intel 兼容 CPU 的 AMD 公司, 沿袭了 IA-32 的思路, 发布了与 x86 兼容的 64 位 CPU。它在 32 位 x86 指令集的基础上加入扩展的 64 位 x86 指令集, 这款芯片在硬件上兼容原来的 32 位 x86 软件, 并同时支持 64 位计算, 使这款芯片成为真正的 64 位 x86 芯片, 即 x86-64。x86-64 在市场上取得了很大的成功。

为了和 AMD 64 位技术竞争, Intel 也回到 x86 路线上, 在 AMD 之后发布了与 x86 兼容的 64 位技术, 命名为 Intel 64 位扩展技术 (Extended Memory 64 Technology, EM64T)。该技术指令集和体系结构向下与 8086 兼容, 在 Pentium 4 6xx、Pentium 4 5x1 (如 541、551、561)、Celeron D 3x1 和 3x6 (331、336、341、346) 等处理器产品中采用。在体系结构上新增了一组附加的 SSE 寄存器, 通过 SSE、SSE2 和 SSE3 指令访问。

在 x86 的 16 位和 32 位时代,技术上都是由 Intel 公司来主导,AMD 与之兼容。而在 x86-64 上,部分技术的主导权已经属于 AMD。例如,AX 和 EAX 是 Intel 公司命名的 16 位和 32 位的寄存器,而 AMD 将 x86 寄存器扩展为 64 位以后,命名为 RAX。Intel 的 EM64T 也使用了 RAX 作为寄存器名称。

1.1.3 CPU 的微结构

微结构 (Micro-Architecture) 也叫作计算机组织,它包含处理器内部的构成及这些构成的部分如何执行指令集。微结构通常被表示成流程图的形式来描述 CPU 内部元件的连接状况,如流水线 (Pipeline)、缓存 (Cache) 设计及各种总线设计等。

Intel 系列 CPU 产品种类繁多,根据 CPU 的微结构,大致可以分为 i386、i486、P5、P6、Netburst、Pentium-M、Core (Merom、Penryn)、Nehalem (Nehalem、Westmere)、Sandy Bridge (Sandy Bridge、Ivy Bridge)、Haswell (Haswell、Rockwell)。市面上的多个 CPU 产品可能源于同一个微结构,属于同一类或同一代产品。同一个微结构下存在多款 CPU 的原因很多,例如版本简化的需要,或者根据市场的需要定制版本等。同一个名称的 CPU 也可以来自不同的微结构,如 Core i7 就包括有 Nehalem 微结构的和 Westmere 微结构的。不同微结构的 CPU 性能和价格都有差异,因此,购买 CPU 时不能只关注 CPU 的产品名称,还需要知道其来自哪个微结构。表 1-1 给出了 Intel 各种微结构和与之相应的产品型号。

表 1-1 Intel CPU 微结构和对应产品型号

发布时间	微结构	CPU 型号
1985 年	i386	80386DX, 80386SX, 80376, 80386SL, 80386EX
1989 年	i486	80486DX, 80486SX, 80486DX2, 80486SL, 80486DX4
1993 年	P5	Pentium, Pentium with MMX
1995 年	P6	Pentium Pro, Pentium II, Celeron (Pentium II-based), Pentium III, Pentium II 和 III Xeon, Celeron (Pentium III Coppermine-based), Celeron (Pentium III Tualatin-based)
2000 年	NetBurst	(32 位) Pentium 4 Xeon, Mobile Pentium 4-M, Pentium 4 EE, Pentium 4E, Pentium 4F, (64 位) Pentium D, Pentium Extreme Edition Xeon
2003 年	Pentium-M	Pentium M, Celeron M, Intel Core, Dual-Core Xeon LV, Intel Pentium Dual-Core
2006 年	Core	(64 位) Xeon, Intel Core 2, Pentium Dual Core, Celeron M
2008 年	Nehalem	Xeon, Core i7, Core i7 Extreme, Core i5
2010 年	Westmere	Xeon, Core i7, Core i7 Extreme, Core i5, Core i3, Pentium, Celeron
2011 年	sandy bridge	Xeon, Core i7, Core i7 Extreme, Core i5, Core i3, Pentium, Celeron
2012 年	Ivy Bridge	Xeon, Core i7, Core i7 Extreme, Core i5, Core i3, Pentium, Celeron
2013 年	Haswell	Core i7, Core i7 Extreme, Core i5, Core i3, Pentium
2014 年	Broadwell	Haswell 架构的升级版本, Core i7 及 i5 的不同版本
2015 年	SkyLake	Core i3、i5 和 i7 的不同版本

1.1.4 微处理器性能指标

1. 速度指标

主频也叫主时钟频率，表示在 CPU 内数字脉冲信号振荡的速度。主频越高，一个时钟周期里完成的指令数也越多，CPU 的运算速度也就越快，执行程序的时间就能缩短。但由于微处理器内部结构不同，并非所有时钟频率相同的 CPU 性能也一样。

外频是系统总线的工作频率，即 CPU 与周边设备传输数据的频率。目前绝大部分计算机系统中外频也是内存与主板之间同步运行的速度。在这种方式下，可以理解为 CPU 的外频直接影响内存的访问速度，外频越高，CPU 就可以同时接收更多的来自外围设备的数据，从而使整个系统的速度进一步提高。目前外频有 66 MHz、100 MHz 和 133 MHz。

倍频是指 CPU 和系统总线之间工作频率相差的倍数，当外频不变时，倍频越高，CPU 主频也就越高。倍频可使系统总线工作在相对较低的频率上，而 CPU 速度可以通过倍频来提升。CPU 主频的计算方式变为：主频=外频×倍频。倍频可以从 1.5 一直到 23，甚至更高，以 0.5 为一个间隔单位。例如，当外频等于 200 MHz 时，倍频为 9，则主频为 $200\text{ MHz} \times 9 = 1.8\text{ GHz}$ 。

例 1.1 假定购买了一颗 CPU，它的工作频率是 2.4 GHz，倍频系数设定为 18，请问外频是多少？

$2.4\text{ GHz} / 18 \approx 133\text{ MHz}$ ，因此，在设置主板跳线时，应将主板频率置为 133 MHz。

前端总线 (Front Side Bus, FSB) 是指处理器到北桥之间的总线。前端总线的数据带宽 = (总线频率 × 数据位宽) ÷ 8。在 Pentium 4 出现之前和初期，前端总线频率与外频是相同的，因此以往往往直接称前端总线为外频，将二者混淆了。前端总线的速度是指处理器和北桥之间的总线速度，更实质性地表示了 CPU 和外界数据传输的速度。而外频的概念是建立在数字脉冲信号震荡速度基础之上的，它影响总线的频率。随着处理器工作频率的增加，流水线的加长，CPU 对总线带宽的要求越来越高。前端总线频率直接影响 CPU 与内存直接数据交换速度，前端总线频率越大，代表着 CPU 与内存之间的数据传输量越大，更能充分发挥出 CPU 的功能，因此采用了 QDR (Quad Data Rate) 技术及其他类似的技术实现这个目的。这些技术使得前端总线的频率成为外频的 2 倍、4 倍甚至更高。如 Intel 公司在 Pentium 4 处理器采用了 4 倍传输率的前端总线，该技术可以使系统总线在一个时钟周期内传送 4 次数据。常见 CPU 所能达到的前端总线频率有 266 MHz、333 MHz、400 MHz、533 MHz、800 MHz 几种。

近年来，随着计算机技术的发展，内存控制器被逐步集成到了处理器中，CPU 直接和内存通信，不再通过北桥，前端总线也就消失了。如 AMD 速龙 64 系列 (K8) 以后，处理器集成了内存控制器，前端总线消失，取而代之的就是超传输技术 (HyperTransport, HT) 总线。HT 总线带宽更大，内存性能超过 Intel 公司的产品。Intel 微处理器相对滞后，直到 Core i 系列以后，内存控制器才集成到了 CPU 中，改变了内存性能被超越的状况。取代 Intel 平台前端总线的就是快速通道互联 (QuickPath Interconnect, QPI) 总线。AMD 的 HT 总线和 Intel 的 QPI 总线具有相同的原理，只不过不同的公司采用不同的名称。

目前，最高的 QPI 速率为 6.4 GT/s，QPI 的传输速率比 FSB 的传输速率快一倍。QPI 总线采用的是 2:1 比率，意思就是实际的数据传输速率两倍于实际的总线时钟速率。所以 6.4 GT/s 的总线速率其实际的总线时钟频率是 3.2 GHz。