计算机与通信专业英语
徐秀兰/主编
学出版社

计算机与通信专业英语

（修订第二版）

徐秀兰／主编

ENGLISH FOR COMPUTERS AND TELECOMMUNICATIONS

北京邮电大学出版社
http:// www.buptpress.com

内容简介

本书主要取材于20世纪90年代以后国外发表的共40余种最新材料。内容广泛，语言现象丰富。所选内容既有基础理论，又尽量跟踪最近两三年内公众关心的热点与新技术。教材内容基本能覆盖计算机、通信这两个专业常用的技术词汇、词组及常见的科技语法。

在教材的编写上，所选专题除正文外，还列出了较多的关键字、注释、习题及部分参考译文。对于某些较难的语法现象，除给出相应的参考译文外还对语法现象予以分析。在本书的最后还附有习题解答及生词表，以供读者快速查阅。

图书在版编目（CIP）数据

计算机与通信专业英语/徐秀兰主编. －修订第二版. －北京：北京邮电大学出版社，2000.12

ISBN 7-5635-0335-8

Ⅰ. 计… Ⅱ. 徐… Ⅲ. ①计算机－英语 ②通信－英语 Ⅳ.H31

中国版本图书馆CIP数据核字（98）第29622号

出版发行：北京邮电大学出版社　电话：010-62282185（发行部）
社　　址：北京市海淀区西土城路10号（100876）
经　　销：各地新华书店销售
印　　刷：北京源海印刷厂印刷
开　　本：787 mm×1 092 mm　1/16
印　　张：22.75
字　　数：564千字
版　　次：2000年12月第1版　2000年12月第1次印刷
印　　数：1—5 000册
书　　号：ISBN 7-5635-0335-8/TN·148
定　　价：27.50元

新版前言

本书的前身是在1998年12月出版的ENGLISH FOR COMPUTERS & TELECOMMUNICATIONS（修订版）。承蒙广大读者厚爱，该修订版在出版后不到两年的时间即告脱销，应广大读者的要求出版社进行了第二次印刷。鉴于近两年来通信与计算技术融合的势头不可阻挡以及计算机技术本身的飞速发展，编者考虑，宜在上次修改的基础上再进行大幅度的增删和改编，以适应技术的发展和满足读者的需要。内容方面，除保留了大部分上次修改中新增加的内容如RISC，面向对象程序设计，JAVA语言，Internet网，多媒体与图像处理外，硬件方面向读者新奉献的内容有以Pentium为代表的CISC及以PowerPC为代表的RISC以及CPU的最新设计技术VLIW。软件方面，有新推出的Windows 2000及前途看好的Linux操作系统，并行与分布式数据库，决策系统及网络安全系统，包括密码概念，防火墙等新技术。多媒体技术方面，有语音处理内容以及在通信方面近年来十分热门的用户接入网及全球通等新技术。

在编写上，保留了本书的原有风格，即对课文中出现的大量生词及词组作出注解以尽可能帮助读者从不同的角度来理解。对文中出现的各种缩略词也一一作了查找和核对。对课文中较难词句的注解上，除给出句子的参考译文外，还尽可能对句子的结构作出必要的分析。此外，在书的末尾给出了生词表以利读者快速查询。

经过两年来的教学实践，编者对原修订版中出现的疏漏和错误也进行了更正。

本书由徐秀兰同志主编并担任第1，2，3，4，5，6，7，8，9，11，12，13，14，19，20等15个单元的编写，徐劲同志担任第15，16，17，18等四个单元的编写。成文德同志担任第10单元的编写。全书注解由徐秀兰及成文德同志完成。徐秀兰同志担任了全书的统稿工作。此外，王明鉴，李程及杨燕群等同志协助了书稿的部分校对与输入。在本次改编中，对所有的习题解答也进行了全面的核对和修改。尽管如此，限于编者水平，各种错误怕仍难以避免。不足之处，恳请读者批评指正。

编者

2000.7于北京

CONTENTS

目　　录

UNIT 1 Computer Systems

UNIT 2 Design Technologies of CPU

UNIT 3 Parallel Processing

UNIT 4 The Memory Hierarchy

UNIT 5 Input/Output and Other Processors

UNIT 6 Operating System

UNIT 7 Programming

UNIT 8 Database Systems

UNIT 9 Decision Support Systems

UNIT 10 Information Networks and Protocols

UNIT 11 Network Security

UNIT 12 Multimedia

UNIT 13 Digital Sound Technology

UNIT 14 Image Processing

UNIT 15 ISDN

UNIT 16 Satellites

UNIT 17 Optical Fiber Communications

UNIT 18 Synchronous Digital Transmission

UNIT 19 Access Network (Subscriber Loop) Technologies

UNIT 20 The Global System for Mobile Communications (GSM)

UNIT 1

Computer Systems

1-1 Why Should People Care of a Computer System?

What do the insides of a computer "look like", and why do we care?

As users we do not have to know the answer to this question, any more than we have to understand the workings of a car engine in order to drive the car.

We can run standard software packages without understanding exactly how they work; we can program a computer in a high-level language without understanding how the machine executes the individual instructions; we can purchase a computer system from a salesman without understanding the specifications of the system.

And yet, there is something missing. Perhaps the package doesn't do exactly what we want, and we don't understand the machine well enough to risk fooling around with the package's options. Perhaps if we understood the system, we might have written the program to be faster and more efficient.[1] Perhaps the salesman did not sell us the optimum system for our job. Or perhaps it's nothing more than a sense of excitement that's missing. But that's important, too!

The jargon of computers has become a part of the English language. You can open any daily newspaper and find references to "8MB RAM" of "64-bit PCI Video Accelerator" or "256k cache" in articles and advertisements. (In a way, it's scary!)

You'll notice that this computer features a 60-MHz Pentium CPU, 8MB of RAM memory, and a 540 megabyte (MB) hard drive, among other things. But how good a system is this? Are these features important to the user? Is this the right combination of features that you need in your computer to have the computer perform the work that you wish to get done?[2] Is a 60-MHz Pentium the best choice of a CPU? Perhaps we are paying too much for the performance that we need. Or maybe we need more.

What does the presence of a 64-bit PCI video accelerator imply in the context of a long-term investment of computers for your organization? What other information about this system would allow you to make a more informed decision?

Some of the expressions used in these articles and ads are obvious from the context. Other references may be more obscure. Presumably, everyone today knows what a "monitor" is. But how many people know what the terms "cache memory" or "multitasking" or "PCI bus" mean or what their importance is? Yet all these expressions have appeared recently in daily newspaper advertisements with the assumption that people would understand the meaning of the ad.

Perhaps you are a student studying to become a computer professional, or perhaps you are simply a user wanting a deeper understanding of what the computer is all about. In either case, you will probably be interacting with computers for the rest of your life. It's nice (as well as useful) to know something about the tools of the trade. More important, understanding the computer system's operations has an immediate benefit: it will allow you to use the machine more effectively.

Key words

informed	有见识的，见闻广博的
jargon	行话，术语
obscure	暗的，不清楚的
optimum	最优的，最佳的
PCI (Peripheral Control Interface)	外围控制接口
presumable	冒昧的，不客气的
scary	可怕的，耸人听闻的

Notes

1. Perhaps if we understood the system, we might have written the program to be faster and more efficient.
   本句为虚拟语气，从句的谓语动词用过去式来虚拟现在，主句用情态动词完成式来表示猜测。全句意思请见参考译文。
2. Is this the right combination of features that you need in your computer to have the computer perform the work that you wish to get done?
   本句为一个多重主从复合句，句中两个 that 各引导一个定语从句。第一个 that 引导的定语从句修饰 features，而第二个 that 引导的定语从句修饰第一个从句中的 work。全句意思请见参考译文。

1-2 What Should People Do With A Computer System?

As a user, you will be aware of the capabilities, strengths, and limitations of the computer system. You will have a better understanding of the commands that you use. You will understand what is taking place during the operation of the programs that you use. You will be able to make informed decisions about your computer equipment and application programs. You will understand more clearly what an operating system is, and how to use it effectively, and to your advantage. You will know when it is preferable to do a job manually, and when the computer should be used. You will understand what it means to "go online", and what benefits that will mean to you. You will improve your ability to communicate with system analysts, programmers, and other computer specialists.

As a programmer, it will allow you to write better programs. You will be able to use the characteristics of the machine to make your programs operate more effectively. For example, choosing the appropriate data type for a variable can result in significantly faster performance. Soon you will know why this is so, and how to make the appropriate choices.

Many computers perform integer calculations incorrectly if the integers exceed a certain size, and do not warn the user of the error. You will learn how this can occur, and what can be done to assure that your programs generate correct results.

You will discover that some computers will process nested loops much more quickly if the index variables are reversed. A rather surprising idea, perhaps, and you'll understand why this is true.

You will understand why programs written in a compiled language like C or Pascal usually run much faster than those written in interpreted BASIC.

As a system analyst, you will be expected to specify computer systems for purchase, for yourself and for your organization. You would like to purchase the computer that best meets the needs of the application. You must be able to read and understand the technical specifications in order to compare different alternatives and to match the system to the users' needs. This book will teach you what you need to know to specify and purchase a system intelligently. You'll know the difference between a CISC and a RISC processor and the advantages and disadvantages of each. You will learn what peripheral hardware is appropriate for your organization's files, what is required to interconnect different computers, and what the speed and size limitations of a particular system are. You'll be able to compare the features of Windows and UNIX knowledgeably and decide which ones are important to you. You'll learn to understand the jargon used by computer salespeople and judge the validity of their sales claims.

You'll be in a better position to determine whether your computer is the right system for a particular job, or whether a different system would be more appropriate. Perhaps a workstation with an Alpha CPU is better suited to your application than a personal computer with a Pentium chip. You'll be better prepared to analyze the best way to provide appropriate facilities to meet the needs of your users. In an era of fast -changing technology, you'll be more able to differentiate between simple technological obsolescence that does not affect your work significantly and major advances that suggest a real need to replace older equipment.[1]

As a system administrator or manager, your job is to maximize the efficiency of your systems. You will need to understand the reports generated by your systems and be able to use the information in those reports to make changes in the systems that will optimize system performance. You will need to know when additional resources are required, and be able to specify appropriate choices. You will need to specify and configure operating system parameters, set up file systems, provide system security, and perform many other system management tasks. The configuration of large systems can be very challenging. This text will give you an understanding of operating system tools that is essential to the effective management of systems.

In brief, when you complete this book, you will understand what computer hardware and software are and how programs and data interact with the computer system. You will understand the computer hardware and software components that are required to make up a computer system and what the role of each component in the system is.

You will have a better understanding of what is happening inside the computer when you interact with the computer as a user. You will be able to write programs that are more efficient. You will be able to understand the function of the different components of the computer system and to specify the computer system you need in a meaningful way. You will understand the options that you have as a system administrator.

In an era in which technology changes extremely rapidly, the fundamental architecture of the computer rests on a solid foundation that has changed only slightly and gradually over the last 50 years. Understanding the foundations of computer architecture makes it possible to flow with the technological change and to understand these changes in the context of the improvements that they make and the needs that they meet.[2]

This type of understanding is at the very foundation of being a competent and successful system analyst, system administrator, or programmer. It may not be necessary to understand the workings of an automobile engine in order to drive a car, but you can bet that a top-notch race car driver knows his or her engine thoroughly and can use it to win races.[3] Like the professional race car driver, it is our intention to help you to use your computer engine effectively to succeed in using your computer in a winning way.

Key words

competent	有能力的，有知识的，有权威的
configuration	配置，安置
obsolescence	过时，陈旧，无用
top-notch	第一流的，最佳的

Notes

1. In an era of fast-changing technology, you'll be more able to differentiate between simple technological obsolescence that does not affect your work significantly and major advances that suggest a real need to replace other equipment.

   本句中，between…and…后面的介词宾语是 simple technological obsolescence 及 major advances，每个宾语后面又各有一个由 that 引导的定语从句分别修饰其先行词 obsolescence 及 advances。全句可译为：在这技术日新月异的纪元里，你可能区别那些不能显著改进你的工作的过了时的技术及那些提醒你真正有必要替换其他的设备的主要技术的进展。

2. Understanding the foundations of computer architecture makes it possible to flow with technological change and to understand these changes in the context of the improvements that they make and the needs that they meet.

   本句中，makes 为谓语，it possible 为复合宾语，to flow with…change 及 to understand…improvements 为两个并列的不定式动词短语，在第二个短语后面又跟有两个由 that 引导的并列的定语从句来修饰 improvements。全句可译为：了解计算机的结构有可能使

你跟上技术的变化以及了解这些改进中的变化和这些变化所能满足的要求。

3. It may not be necessary to understand the workings of an automobile engine in order to drive a car, but you can bet that a top-notch race car driver knows his or her engine thoroughly and can use it win races.

本句为一个并列复合句，两句由 but 连接，但第二个主句带有一个由 that 引导的宾语从句，从句中有两个并列的谓语。全句可译为：为了驾驶汽车也许没有必要去了解汽车发动机的工作情况，但是你敢说一个第一流的汽车赛手透彻了解他（或她）的汽车发动机并能用它获取比赛的胜利。

1-3 Designing for Performance

Year by year, the cost of computer systems continues to drop dramatically, while the performance and capacity of those systems continue to rise equally dramatically, At a local warehouse club, you can pick up a personal computer for less than $1000 that packs the wallop of an IBM mainframe from 10 years ago. Inside that personal computer, including the microprocessor and memory and other chips, you get roughly 100 million transistors. You cannot buy 100 million of anything else for so little. That many sheets of toilet paper would run more than $100,000.[1]

Thus, we have virtually "free" computer power. And this continuing technological revolution has enabled the development of applications of astounding complexity and power. For example, desktop applications that require the great power of today's microprocessor-based systems include

- Image processing
- Speech recognition
- Videoconferencing
- Multimedia authoring
- Voice and video annotation of files

Workstation systems now support highly sophisticated engineering and scientific applications, as well as simulation systems, and the ability to apply workgroup principles to image and video applications. In addition, businesses are relying on increasingly powerful servers to handle transaction and database processing and to support massive client-server networks that have replaced the huge mainframe computer centers of yesteryear.

What is fascinating about all this from the perspective of computer organization and architecture is that, on the one hand, the basic building blocks for today's computer miracles are virtually the same as those of the IAS computer from nearly 50 years ago, while on the other hand, the techniques for squeezing the last iota of performance out of the materials at hand have become increasingly sophisticated.[2]

This observation serves as a guiding principle for the presentation in this book. As we progress through the various elements and components of a computer, two objectives are pursued. First, the book explains the fundamental functionality in each area under consideration, and second, the book explores those techniques required to achieve maximum performance. In the remainder of this section, we highlight some of the driving factors behind the need to design for performance.

Microprocessor Speed

What gives the Pentium or the Power PC such mind-boggling power is the relentless pursuit of speed by processor chip manufacturers. The evolution of these machines continues to bear out what is known as Moore's law. Intel Chairman Gordon Moore observed in the mid-1960s that, by shrinking the size of the tiny lines that form transistor circuits in silicon roughly 10% a year, chipmakers could unleash a new generation of chips every three years—with four times as many transistors. In memory chips, this has quadrupled the capacity of dynamic random-access memory (DRAM), still the basic technology for computer main memory, every three years. In microprocessors, the addition of new circuits, and the speed boost that comes from reducing the distances between them, has improved performance four- or five-fold every three years since Intel launched its X86 family in 1979.[3]

But the raw speed of the microprocessor will not achieve its potential unless it is fed a constant stream of work to do in the form of computer instructions. Anything that gets in the way of that smooth flow undermines the power of the processor. Accordingly, while the chipmakers have been busy learning how to fabricate chips of greater and greater density, the processor designers must come up with ever more elaborate techniques for feeding the monster. Among the techniques built into contemporary processors are

- *Branch Prediction*: The processor looks ahead in the software and predicts which branches, or groups of instructions, are likely to be processed next. If the processor guesses right most of the time, it can prefetch the correct instructions and buffer them so that the processor is kept busy. The more sophisticated examples of this strategy predict not just the next branch but multiple branches ahead. Thus, branch prediction increases the amount of work available for the processor to execute.
- *Data Flow Analysis*: The processor analyzes which instructions are dependent on each other's results, or data, to create an optimized schedule of instructions. In fact, instructions are scheduled to be executed when ready, independent of the original program order. This prevents unnecessary delay.
- *Speculative Execution*: Using branch prediction and data flow analysis, some processors speculatively execute instructions ahead of their actual appearance in the program execution, holding the results in temporary locations. This enables the processor to keep its execution engines as busy as possible by executing instructions that are likely to be needed.

These and other sophisticated techniques are made necessary by the sheer power of the processor. They make it possible to exploit the raw speed of the processor.

Performance Balance

While processor power has raced ahead at breakneck speed, other critical components of the computer have not kept up. The result is a need to look for performance balance: an adjusting of the organization and architecture to compensate for the mismatch among the capabilities of the various components.

Nowhere is the problem created by such mismatches more critical than in the interface between processor and main memory. While processor speed and memory capacity have grown rapidly, the speed with which data can be transferred between main memory and the processor has lagged badly. The interface between processor and main memory is the most crucial pathway in the entire computer, because it is responsible for carrying a constant flow of program instructions and data between memory chips and the processor. If memory or the pathway fails to keep pace with the processor's insistent demands, the processor stalls in a wait state, and valuable processing time is lost.

The effects of these trends are shown vividly in Figure 1.1. The amount of main memory needed is going up , but DRAM density is going up faster. The net result is that, on average, the number of DRAMS per system is going down. The solid black lines in the figure show that, for a fixed-size memory, the number of DRAMs needed is declining. But this has an effect on transfer rates, because with fewer DRAMs, there is less opportunity for parallel transfer of data. The shaded bands show that for a particular type of system, main memory size has slowly increased while the number of DRAMs has declined.

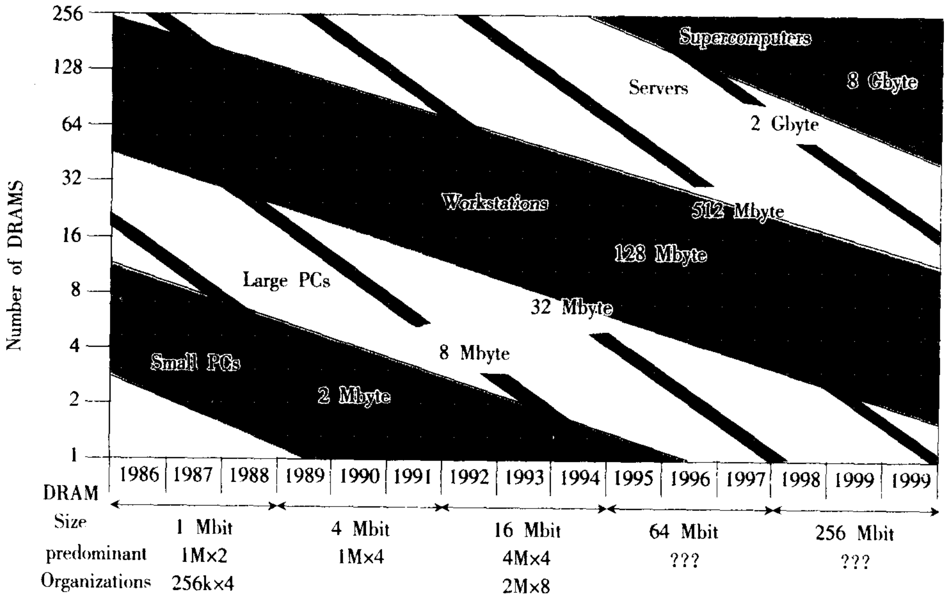

Figure 1.1 Trends in DRAM Use.

There are a number of ways that a system architect can attack this problem, all of which are reflected in contemporary computer designs. Some examples:

- Increase the number of bits that are retrieved at one time by making DRAMs "wider" rather than "deeper" and by using wide bus data paths.
- Changing the DRAM interface to make it more efficient, by including a cache or other buffering scheme on the DRAM chip.

- Reduce the frequency of memory access by incorporating increasingly complex and efficient cache structures between the processor and main memory. This includes the incorporation of one or more caches on the processor chip as well as on an off-chip cache close to the processor chip.
- Increase the interconnect bandwidth between processors and memory by using higher-speed buses and by using a hierarchy of buses to buffer and structure data flow.

Another area of design focus is the handling of I/O devices. As computers become faster and more capable, more sophisticated applications are developed that support the use of peripherals with intensive I/O demands. Table 1.1 gives some examples of typical peripheral devices in use on personal computers and workstations. These devices create tremendous data throughput demands. While the current generation of processors can handle the data pumped out by these devices, there remains the problem of getting that data moved between processor and peripheral. Strategies here include caching and buffering schemes plus the use of higherspeed interconnection buses and more elaborate structures of buses. In addition, the use of multiple-processor configurations can aid in satisfying I/O demands.

The key in all this is balance. Designers all constantly strive to balance the throughput and processing demands of the processor components, main memory, I/O devices, and the interconnection structures. And this design must constantly be rethought to cope with two constantly evolving factors:

1. The rate at which performance is changing in the various technology areas (processor, buses, memory, peripherals) differs greatly from one type of element to another.
2. New applications and new peripheral devices constantly change the nature of the demand on the system in terms of typical instruction profile and the data access patterns.

Thus, computer design is a constantly evolving art form. This book attempts to present the fundamentals on which this art form is based and to present a survey of the current state of that art.

Table 1.1 Typical Bandwidth Requirements for Various Peripheral Technologies

Peripheral	Technology	Required Bandwidth
Graphics	24-bit color	30 MBytes/sec
Local area network	100BASEX or FDDI	12 MBytes/sec
Disk controller	SCSI or P1394	10 MBytes/sec
Full-motion video	1 024 × 768@30bps	67 + MBytes/sec
I/O Peripherals	Other miscellaneous	5 + MBytes/sec

Key words

breakneck	危险的速度
FDDI (Fiber Distributed Digital Interface)	光纤分布式数字接口
hierachy	层次，分层
iota	希腊文中的第九个字母，极少量
lag	延迟，滞后
mind-boggling	令人瞠目结舌的，令人惊愕的
prefetch	预取，预拿

relentless	无情的，残忍的
speculative	推测的，假设的
squeeze	压榨，榨取，挤出
throughput	吞吐量
undermine	削弱，从基础损坏
unleash	解开，放出，推出

Notes

1. That many sheets of toilet paper would run more than $100,000.
   本句中的 that many sheets 是指与上句呼应的 100 million sheets.
2. What is fascinating about all this from the perspective of computer organization and architecture is that , on the one hand, the basic building blocks for today's computer miracles are virtually the same as those of the IAS computer from nearly 50 years ago. While on the other hand , the techniques for squeezing the last iota of performance out of the materials at hand have become increasingly sophisticated.
   本句的主语是一个由 what 引导的主语从句，谓语动词是 is，句中有两个并列的表语从句。在表语从句之前分别由 *on the one hand* 及 while *on the other hand* 插入语分开。全句可译为：从计算机组织及结构的观点来看，所有这些令人惊讶的是，一方面在于今天所成就的计算机方面的奇迹实际上差不多等于五十年前 IAS 计算机所取得的。而另一方面，从现有材料中获取最后一丁点儿性能方面潜力的技术已大大的复杂了。
3. In microprocessors, the addition of new circuits, and the speed boost that comes from reducing the distances between them, has improved performance four- or five-fold every three years since Intel launched its X86 family in 1979.
   本句是一个主从复合句。主句主语是 the addition of new circuits, and the speed boost。即两个内容用 and 连接当作一件事情，因而主句谓语动词用第三人称单数“has improved”。主句中有一个由 that 引导的一个定语从句来修饰其主语并有一个由 since 引导的时间状语从句来说明主句动作发生的时间。全句可译为：在微处理器方面，自从 Intel 在 1979 年推出其 X86 系列以来，从减少电路间的距离而得到新增加的电路以及速度的提高已经获得每三年改善性能四至五倍。

1-4 Pentium and PowerPC Evolution

Throughout this book, we rely on many concrete examples of computer design and implementation to illustrate concepts and to illuminate trade-offs. Most of the time, the book relies on examples from two computer families: the Intel Pentium and the PowerPC. The Pentium represents the results of decades of design effort on complex instruction set computers (CISCs). It incorporates the sophisticated design principles once found only on mainframes and supercomputers and serves as an excellent example of CISC design. The PowerPC is a direct descendant of the first RISC system, the IBM 801, and is one of the most powerful and best-designed RISC-based systems on the market.

In this section, we provide a brief overview of both systems.

Pentium

Intel has ranked as the number one maker of microprocessors for decades, a position it seems unlikely to yield. The evolution of its flagship microprocessor product serves as a good indicator of the evolution of computer technology in general.

Table 1.2 shows that evolution. Although the Pentium is now the star performer in Intel's product line, they already have two more processors in the pipeline: the P6, introduced in 1995, and the P7, still under development. Interestingly, as microprocessors have grown faster and much more complex, Intel has actually picked up the pace. Intel used to develop microprocessors one after another, every four years. But for the Pentium, the generation gap shrank to three years. And for its P6 and P7 chips, Intel hopes to keep rivals at bay by trimming another year off.[1]

It is worthwhile to list some of the highlights of the evolution of the Intel product line:

- 8080: The world's first general-purpose microprocessor. This was an 8-bit machine, with an 8-bit data path to memory.
- 8086: A far more powerful, 16-bit machine. In addition to a wider data path and larger registers, the 8086 sported an instruction cache, or queue, that prefetches a few instructions before they are executed.
- 80286: This extension of the 8086 enabled addressing a 16 MByte memory instead of just 1 MByte.
- 80386: Intel's first 32-bit machine, and a major overhaul of the product. With a 32-bit architecture, the 80386 rivaled the complexity and power of minicomputers and mainframes introduced just a few years earlier.
- 80486: The 80486 introduces the use of much more sophisticated and powerful cache technology and sophisticated instruction pipelining.

Table 1.2 Intel Microprocessors

Characteristic	286	386	486	Pentium	P6	P7
Start of design work	1978	1982	1986	1989	1990	1993
Formal introduction	Feb. 1982	Oct. 1985	Apr. 1989	Mar. 1993	Q3 1995	1997 or 1998[b]
Volume shipments	1983	1986	1990	1994	1996	1998 or 1999[b]
Number of transistors	130,000	275,000	1.2 million	3.1 million	5.5 million	10 + million[b]
Initial speed in MIPS[a]	1	5	20	100	250[b]	500[b]
Peak sales year	1989	1992	1995[b]	1997[b]	1999[b]	2000[b]
Installed units	9.7 million	44.2 million	75 million	4.5 million	none	none

a Millions of instructions per second.

b Estimated.