计算机英语（第三版）实用教程

刘兆毓 郑家农 等编著

a b c d e f g h i j k l
m n o p q r s t u v w x
y z
A B C D E F G H I J K L
M N O P Q R S T U V W
Y Z

清华大学出版社

计算机英语（第三版）实用教程

刘兆毓 郑家农 等编著

清华大学出版社
北京

内容简介

本书是按计算机科学与技术的知识结构，采用英文编写的教材。本书内容包括计算机系统的组成、数据结构、程序设计语言和操作系统等基础知识；并且重点介绍了计算机应用技术，包括数据库、软件工程、计算机辅助工程、计算机图形图像技术、多媒体技术、计算机网络、电子商务和计算机安全等。

为帮助读者学习并掌握计算机英语，对书中较难翻译和理解的句子和语法现象做了注释；每一节后列出关键词汇，给出练习题；书后附有习题答案和参考译文，供读者参考。

本书可供高等院校学生使用，也可供参加计算机水平考试的学生和广大工程技术人员使用和参考。

图书在版编目(CIP)数据

计算机英语(第三版)实用教程/刘兆毓等编著．—北京：清华大学出版社，2006.5

ISBN 7-302-12706-9

Ⅰ．计…　Ⅱ．刘…　Ⅲ．电子计算机—英语—高等学校—教材　Ⅳ．H31

中国版本图书馆 CIP 数据核字(2006)第 021103 号

出 版 者：清华大学出版社　　**地　　址**：北京清华大学学研大厦

http://www.tup.com.cn　　**邮　　编**：100084

社 总 机：010-62770175　　**客户服务**：010-62776969

责任编辑：张　龙

印 刷 者：北京市昌平环球印刷厂

装 订 者：三河市新茂装订有限公司

发 行 者：新华书店总店北京发行所

开　　本：185×260　**印张**：14.5　**字数**：509 千字

版　　次：2006 年 5 月第 1 版　2006 年 5 月第 1 次印刷

书　　号：ISBN 7-302-12706-9/TP・8111

印　　数：1～5000

定　　价：19.00 元

前　言

《计算机英语》一共出版了三个版本，累计销售近50万册，得到了广泛的采用。本书在保留《计算机英语(第三版)》基本结构的基础上，进行了课程内容的重新组合，使得读者在学时较少的情况下，同样能够掌握涉及计算机技术基础、系统和应用等各个方面的计算机英语知识。

与《计算机英语(第三版)》相比，本书删去了数学基础理论部分，保留了计算机技术及应用中的基本内容。鉴于因特网的发展和日渐普及，为节省篇幅，删除了原书中每节课后面的阅读材料(Reading Materials)和相关词汇(Associative Words)。

本书共有14章，其中第1章为计算机硬件基础知识，第2章～第4章为软件基础知识，第5章～第14章为计算机应用知识。本书由刘兆毓(第1章，第4章，第6章，第11章，第12章)、郑家农(第7章，第8章，第9章，第13章)、施家元(第10章)、闫金平(第14章)、郭庆云(第2章和第3章)、武华(第5章)等人共同撰写完成，并由刘兆毓担任主编。

本书配有电子教案，使用本书的教师可以通过电子邮件(jsjjc@tup.tsinghua.edu.cn)向清华大学出版社免费索取。

本书内容相对精简，学习难度适当下调。读者可以根据个人实际情况从《计算机英语(第三版)》和《计算机英语(第三版)实用教程》两个版本中选择适合自己的教材。

编　者

2006年2月

CONTENTS

CHAPTER 1 PRINCIPLES OF COMPUTER ORGANIZATION

1.1 COMPUTER HARDWARE

We build computers to solve problems. Early computers solved mathematical and engineering problems, and later computers emphasized information processing for business applications. Today, computers also control machines as diverse as automobile engines, robots, and microwave ovens. A computer system solves a problem from any of these domains by accepting input, processing it, and producing output. Fig. 1-1 illustrates the function of a computer system.

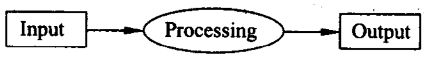

Fig. 1-1 The three activities of a computer system

Computer systems consist of hardware and software. *Hardware* is the physical part of the system. Once designed, hardware is difficult and expensive to change. *Software* is the set of programs that instruct the hardware and is easier to modify than hardware. Computers are valuable because they are general-purpose machines that can solve many different kinds of problems, as opposed to special-purpose machines that can each solve only one kind of problems. Different problems can be solved with the same hardware by supplying the system with a different set of instructions, that is, with different software.

Every computer has four basic hardware components:

- Input devices.
- Output devices.
- Main memory.
- Central processing unit(CPU).

Fig. 1-2 shows these components in a block diagram. The lines between the blocks represent the flow of information from one component to another on the *bus*, which is simply a group of wires connecting the components. [1] Processing occurs in the CPU and main memory. The organization in Fig. 1-2, with the components connected to each other by the bus, is common. However, other configurations are possible as well.

Computer hardware is often classified by its relative physical size:

- *Small* microcomputer;
- *Medium* minicomputer;
- *Large* mainframe.

Just the CPU of a mainframe often occupies an entire cabinet. Its input/output (I/O) devices and memory might fill an entire room. Microcomputers can be small

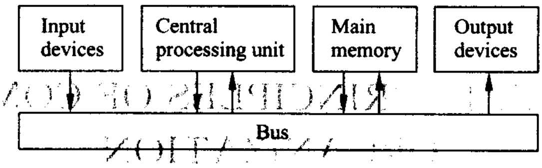

Fig. 1-2 Block diagram of the four components of a computer system

enough to fit on a desk or in a briefcase. As technology advances, the processing capabilities previously possible only on large machines become possible on smaller machines. Microcomputers now can do much of the work that only minicomputers or mainframes could do in the past.

The classification just described is based on physical size as opposed to storage size. A computer system uset is generally more concerned with storage size, because that is a more direct indication of the amount of useful work that the hardware can perform. Speed of computation is another characteristic that is important to users. Generally speaking, users want a fast CPU and large capacity of storage, but physically small I/O devices and main memory.

When computer scientists study problems, therefore, they are concerned with space and time—the space necessary inside a computer system to store a problem and the time required to solve it. They commonly use the metric prefixes of Table 1-1 to express large or small quantities of space or time.

Table 1-1 Prefixes for powers of 10

Multiple	Prefix	Abbrev
10^9	giga-	G
10^6	mega-	M
10^3	kilo-	K
10^{-3}	milli-	m
10^{-6}	micro-	μ
10^{-9}	nano-	n

Example Suppose it takes 4. 5 microseconds, also written 4. 5 μs, to transfer some information across the bus from one component to another. (a) How many seconds are required for the transfer? (b) How many transfers can take place during one minute?[2]

(a) A time of 4. 5μs is 4.5×10^{-6} from Table 1-1 or 0. 000 004 5 s. (b) Because there are 60 seconds in one minute, the number of times the transfer can occur is (60 s)/(0. 000 004 5 s/transfer) or 13 300 000 transfers. Note that since the original value was given with two significant figures, the result should not be given to more than two or three significant figures.

Table 1-1 shows that in the metric system the prefix kilo - is 1000 and mega - is 1 000 000. But in computer science, a kilo - is 2^{10} or 1024. The different between 1000 and 1024 is less than 3%, so you can think of a computer science kilo- as being about 1000 even though it is a little more. The same applies to mega -and giga-, as in Table

1-2. This time, the approximation is a little worse, but for mega-it is still within 5%.

Table 1-2 Computer science values of the large prefixes

Prefix	Computer science value
giga-	$2^{30}=1\ 073\ 741\ 824$
mega-	$2^{20}=1\ 048\ 576$
kilo-	$2^{10}=1\ 024$

NOTES

[1] 总线不仅仅是一组线缆，它还包括一些逻辑电路。此处是一种形象说法，有关总线的概念请见本章 1.6 节。

[2] 因为没有给出总线宽度，故每次传送可以理解为每秒钟或每分钟所传送的字节数、字数等。

KEYWORDS

computer	计算机	input device	输入设备
information processing	信息处理	output device	输出设备
hardware	硬件	main memory	主存储器
software	软件	central processing unit (CPU)	中央处理器
program	程序	bus	总线
general-purpose machine	通用(计算)机	microcomputer	微型计算机
special-purpose machine	专用(计算)机	minicomputer	小型计算机
instruction	指令	mainframe	主机，大型机
set of instructions	指令集，指令系统		

EXERCISES

1. Multiple choices.

(1) When we store a problem into a computer, ________ is necessary.

a. space　　b. time
c. input device　　d. output device

(2) Early computer solved ________ problems.

a. control　　b. business applications
c. engineering　　d. mathematical

(3) We can use prefix micro to express ________.

a. time metric　　b. space metric
c. both time and space metrics　　d. 10^{-6}

(4) We can say a bus is simply ________.

a. a group of wires　　b. a wire
c. a 8-bit bus　　d. a 16-bit bus

(5) A computer system user generally more cares for ________.

a. physical size of the computer
b. storage size
c. speed of computation
d. efficiency of the computer

(6) According to the physical size of computers we can classify the computers into ________.

a. microcomputer　　b. minicomputer
c. mainframe　　d. supercomputer

(7) Prefix "mega" used for computer science is ________.

a. larger than 10^6　　b. smaller than 10^6

c. equal to 2^{20} d. 1 048 576

(8) The basic hardware components of any computer include ________.

a. CPU b. main memory

c. input devices d. output devices

2. Fill in the blank with appropriate words or phrases found behind this exercise.

(1) A computer system solves a problem by ________.

(2) The amount of effective work of a computer can be indicated by ________ directly.

(3) Computer systems consist of ________.

(4) Computer that can solve only one kind of problem is a ________.

(5) Computer that can solve many different kinds of problems is a ________.

(6) ________ instruct the hardware.

(7) ________ is difficult and expensive to change.

(8) We usually show the computer components in a ________.

a. general-purpose machine

b. hardware

c. accepting input, processing problem, and producing output

d. block diagram

e. software

f. storage size

g. special-purpose machine

h. hardware and software

1.2 WHAT IS A PROCESSOR

A processor is a functional unit that interprets and carries out instructions. Every processor comes with a unique set of operations such as ADD, STORE, or LOAD that represent the processor's instruction set. Computer designers are fond of calling their computers machines[1], so the instruction set is sometimes referred to as machine instructions and the binary language in which they are written is called machine language! You shouldn't confuse the processor's instruction set with the instructions found in high-level programming languages, such as BASIC or Pascal.

An instruction is made up of operations that specify the function to be performed and operands that represent the data to be operated on. For example, if an instruction is to perform the operation of adding two numbers, it must know (1) what the two numbers are and (2) where the two numbers are. When the numbers are stored in the computer's memory, they have their addresses to indicate where they are. So if an operand refers to data in the computer's memory it is called an address. The processor's job is to retrieve instructions and operands from memory and to perform each operation. Having done that, it signals memory to send it the next instruction.

This step-by-step operation is repeated over and over again at awesome speed. A timer called a clock releases precisely timed electrical signals that provide a regular pulse for the processor's work. The term that is used to measure the computer's speed is borrowed from the domain of electrical engineering and is called a megahertz (MHz), which means million cycles per second. For example, in an 8-MHz processor, the computer's clock ticks 8 million times to every 1 second tick of an ordinary clock.

A processor is composed of two functional units—a control unit and an arithmetic/logic unit—and a set of special workspaces called registers.

1. The Control Unit

The control unit is the functional unit that is responsible for supervising the operation of the entire computer system. In some ways, it is analogous to a telephone switch-board with intelligence because it makes the connections between various functional units of the computer system and calls into operation each unit that is required by the program currently in operation[2].

The control unit fetches instructions from memory and determines their types or decodes them. It then breaks each instruction into a series of simple small steps or actions. By doing this, it controls the step-by-step operation of the entire computer system.

2. The Arithmetic and Logic Unit

The arithmetic and logic unit (ALU) is the functional unit that provides the computer with logical and computational capabilities. Data are brought into the ALU by the control unit, and the ALU performs whatever arithmetic or logic operations are required to help carry out the instruction[3].

Arithmetic operations include adding, subtracting, multiplying, and dividing. Logic operations make a comparison and take action based on the results. For example, two numbers might be compared to determine if they are equal. If they are equal, processing will continue; if they are not equal, processing will stop.

3. Registers

A register is a storage location inside the processor. Registers in the control unit are used to keep track of the overall status of the program that is running. Control unit registers store information such as the current instruction, the location of the next instruction to be executed, and the operands of the instruction. In the ALU, registers store data items that are added, subtracted, multiplied, divided, and compared. Other registers store the results of arithmetic and logic operations.

An important factor that affects the speed and performance of a processor is the size of the registers. Technically, the term word size (also called word length) describes the size of an operand register, but it is also used more loosely to describe the size of the pathways to and from the processor. Currently, word sizes in general purpose computers range from 8 to 64 bits. If the operand registers of a processor are 16 bits wide, the processor is said to be a 16-bit processor.

NOTES

[1] be fond of doing ... 是短语"乐于……,喜欢……";call computers machines 意为"把计算机称为机器"。

[2] because 后的原因状语从句,由 makes 和 calls 带出的两个并列分句组成。calls into operation each unit 中的双宾语倒装,正常语序为 calls each unit into operation。原意为"传唤各部件进行

操作”，这里指微操作，实际上是指“完成微操作”。

[3] 这是一个 and 连接的并列句。后一个分句中的 whatever 是关系代词，引导后面的宾语从句。

KEYWORDS

instruction	指令	clock	时钟
instruction set	指令系统，指令集	megahertz (MHz)	兆赫
processor	处理器	control unit	控制器，控制部件
operation	操作、操作码、操作码指令	arithmetic and logic unit (ALU)	算术/逻辑部件
operand	操作数	word size (word length)	字长
register	寄存器	machine language	机器语言

EXERCISES

1. Match the following terms to the appropriate definition.

(1) ________ Processor.
(2) ________ Instruction set.
(3) ________ Clock.
(4) ________ Machine language.
(5) ________ Operation.
(6) ________ Operand.
(7) ________ Megahertz (MHz).
(8) ________ Control unit.
(9) ________ Arithmetic and logic unit (ALU).
(10) ________ Register.
(11) ________ Word size.

a. The part of an instruction that specifies the function that is to be performed.
b. The binary language in which a computer's instruction set is written.
c. A timer in a processor that releases precisely timed signals that provide a pulse for the processor's work.
d. A functional unit that interprets and carries out instructions.
e. A unique set of operations that comes with every processor.
f. The part of an instruction that tells where data that are operated on are located.
g. Million cycles per second.
h. The function unit that is responsible for supervising the operation of the entire computer system.
i. A function unit that provides the computer with logical and computational capability.
j. The term used to describe the size of operand registers and buses.
k. A storage location inside the processor.

2. Fill in the blanks with appropriate words or phrases.

(1) We usually call our computers ________.
(2) An instruction set can sometimes be referred to as ________.
(3) The binary language is called ________.
(4) We don't confuse the processor's instruction set with the instructions of ________.
(5) An instruction consists of ________.
(6) An operand that refers to data in the memory is called an ________.
(7) A timer can give precisely timed ________.
(8) A processor includes two functional units; they are ________.
(9) The ways by which the control unit works are analogous to ________.
(10) The control unit takes out the ________ from memory.

a. address
b. high-level programming languages
c. instructions
d. machines
e. electrical signals
f. machine instructions
g. a telephone switch-board with intelligence
h. operations and operands

i. machine language j. control unit and ALU

1.3 MEMORY SYSTEMS

1. Memory System Desiderata

The memory system has three desiderata.

(1) Size: infinitely large, no constraints on program or data set size.

(2) Speed: infinitely fast, latency equal to the fastest memory technology available[1].

(3) Cost: the per bit cost should approach the lowest-cost technology available.

Clearly these specifications cannot all be achieved as they are mutually exclusive. However, with the semiconductor and magnetic memory technology of today, these specifications are closely approximated.

2. Hierarchical Memory

In this section it is shown how designers implement a practical memory that approaches the performance of an ideal memory at reasonable cost. This memory system has a hierarchy of levels: The memory closest to the processor is fast and relatively small, but has a high cost per bit. This level is called the cache; The real memory, sometimes known as main memory, is slower, larger, and has a lower cost per bit than the cache; The lowest level in the hierarchy is usually a magnetic disk that has the longest latency and the lowest bandwidth, however, it can be very large and has a very low cost per bit. This hierarchy is illustrated in Fig. 1-3.

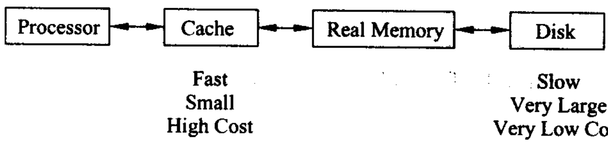

Fig. 1-3 Hierarchical memory

Note that Fig. 1-3 does not include the processor register file in the memory hierarchy.[2] The register file is a program-managed cache and is generally not included in the memory system. Also, there can be more than one cache in the hierarchy.

3. Paged Virtual Memory

Paged virtual memory provides the solution to the first desideratum of a very large memory's being available to the processor. Because of the importance of this desideratum, the relationship between virtual and real memory is discussed first. With virtual memory, the processor addresses the disk, the large, slow end of the hierarchy.[3] The memories between the processor and the disk are there to increase the effective performance (reduced latency and increased bandwidth) of the very slow disk. If every instruction and data reference were made to the disk, the processor performance would be slow indeed.

Then why is virtual memory so important? Large memory is needed for large programs and data sets. Early computers with small real memory required that the transfer of data between real memory and disk be managed explicitly by the operating system or the user. Virtual memory provides for automatic management of this portion of the memory hierarchy through a combination of hardware and software aids.

The virtual memory interface is shown in Fig. 1-4. A real memory of 16M bytes and a virtual memory of 2G bytes are shown for illustration; many modern virtual-memory systems are much larger than this. Virtual-memory space is divided into equal-sized groups called pages. A page in a modern computer is 1K, 2K, or 4K bytes. Real memory is also divided into the same equal-sized groups, called page frames. When information is moved between virtual-memory space and real-memory space, a complete page is moved.

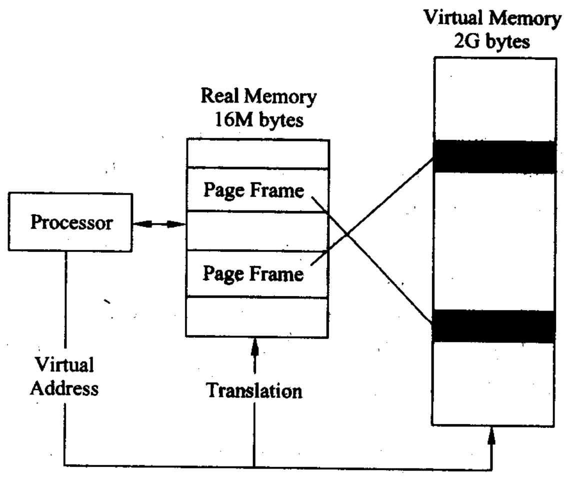

Fig. 1-4 Page allocation and address translation

4. Caches

Section 3 discussed how virtual memory extends the address space of a processor. However, the latency of real memory is too long to support high-performance processors. Even with the high-speed DRAMs used today for real memory, something must be done to overcome this latency problem.

The solution to this performance problem is to add another level to the memory hierarchy, called a cache, shown in Fig. 1-3. The allocation of spaces in a three-level memory is shown in Fig. 1-5. As discussed in Section 3, the virtual-address space is divided into equal-sized pages. These pages are placed in real-memory frames of the same size. Because a page can be placed in any vacant frame, there is no particular order to the allocation of pages to frames. With the addition of a cache, blocks of 16-32 bytes are taken from the page frames in real memory and placed into a block slot for access by the processor. For modern processors the cache usually has a latency of one processor clock, so that instructions and data may be fetched without delay except when the referenced item is not in the cache.

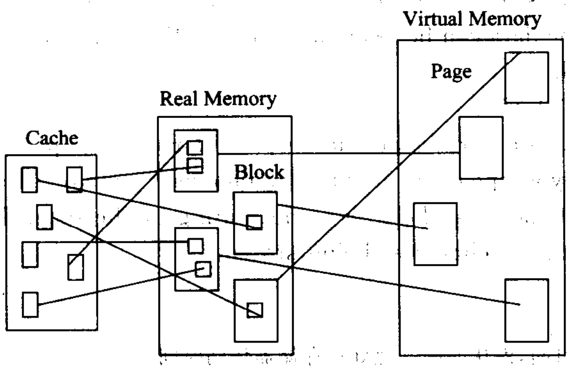

Fig. 1-5 Real memory and cache allocation

5. Memory Devices

1) RANDOM-ACCESS MEMORY

Random-access memory, or RAM, is the kind of memory we usually refer to when we speak of computer memory. It is the most widely used type, and consists of rows of chips with locations established in tables maintained by the control unit[4].

As the name suggests, items stored in RAM can be gotten (accessed) both easily and in any order (randomly) rather than in some sequence. RAM relies on electric current for all its operations; moreover, if the power is turned off or interrupted, RAM quickly empties itself of all your hard work. Thus, we say RAM is volatile, or nonpermanent.

2) READ-ONLY MEMORY

Read-only memory, or ROM, typically holds programs. These programs are manufactured, or "hard-wired" in place on the ROM chips. For example, a microcomputer has a built-in ROM chip (sometimes called ROM BIOS, for ROM basic input/output system) that stores critical programs such as the one that starts up, or "boots", the computer. ROM is "slower" than RAM memory, and as a result, items in ROM are transferred to RAM when needed for fast processing.

Items held in ROM can be read, but they cannot be changed or erased by normal input methods. New items cannot be written into ROM. The only way to change items in most forms of ROM is to change the actual circuits.

3) MAGNETIC DISKS

The magnetic disk is a circular platter with a smooth surface and a coating that can be magnetized. Data is stored on it as magnetized spots. The reading and recording device, the disk drive, spins the disk past read/write heads that detect or write the magnetized spots on the disk.

4) CD-ROMS

Optical disks need thin beams of concentrated light to store and read data. It is a form of laser storage, called CD-ROM.