



北京市高等教育精品教材立项项目

计算机通信国家级教学团队成果

微机原理与接口技术

周 锋 倪 晖 编著

高等学校工程创新型“十二五”规划计算机教材



Engineering Innovation



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

第1章 微机基础

1.1 计算机中的数和字符

1.1.1 进位制

进位计数制是最常用的数值表示方法。一个数由一定数目的数码排列在一起组成，每个数码的位置称为数位，每个数位上所能使用的数码的个数称为基数，每个数位上的数码所代表的数值是数码乘以该数位的位权，位权是基数的次幂。运算中，某一数位累计到基数以后向高数位进一。

数 N 使用 r 进制，即基数为 r 的进位计数制，可以表示为

$$(d_{n-1}d_{n-2}\cdots d_1d_0.d_{-1}\cdots d_{-m})_r$$

其数值是

$$N = d_{n-1} \times r^{n-1} + d_{n-2} \times r^{n-2} + \cdots + d_1 \times r^1 + d_0 \times r^0 + d_{-1} \times r^{-1} + \cdots + d_{-m} \times r^{-m}$$

式中， d_i 是 r 个数码 $0、1、\cdots、(r-1)$ 中任意一个； $i=n-1、n-2、\cdots、1、0、-1、\cdots、-m$ 是数位； r^i 为数位 i 的位权。

十进制是日常使用的计数制。十进制的基数是 10，数码是 0、1、2、3、4、5、6、7、8、9，遵循逢十进一的规则。例如：

$$100420 = 1 \times 10^5 + 0 \times 10^4 + 0 \times 10^3 + 4 \times 10^2 + 2 \times 10^1 + 0 \times 10^0$$

计算机中为便于存储以及计算的物理实现，普遍采用二进制。二进制的基数是 2，只有 0 和 1 两个数码，遵循逢二进一的规则。例如：

$$(101011)_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (43)_{10}$$

在计算机中，还经常使用十六进制。十六进制的基数是 16，数码是 0~15，其中 10、11、12、13、14、15 分别用 A、B、C、D、E、F 表示，遵循逢十六进一规则。例如：

$$(5AF)_{16} = 5 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 = (1455)_{10}$$

【例 1.1】 将十进制数 420 分别转换为二进制和十六进制。

十进制数转换为二进制或十六进制的方法是：用十进制数分别不断除以 2 或 16，记下每次相除时的余数，直到商 0 为止。过程如下：

2		214	余0= d_0
		107	余1= d_1
		53	余1= d_2
		26	余0= d_3
		13	余1= d_4
		6	余0= d_5
		3	余1= d_6
		1	余1= d_7
		0	

16		214	余6= d_0
		13	余13= d_1
		0	

所以， $(420)_{10}=(11010110)_2=(D6)_{16}$ 。

从上例可以看出，二进制的数码只有 0 和 1，数的二进制表示形式较长，在书写时不太方便，阅读也不直观。十六进制实际上可以看成是由于 $2^4=16$ 而从二进制衍生而来的，一个十六进制数码正好与 4 位二进制数码一一对应，其对应关系如表 1.1 所示。二进制和十六进制相互转换十分方便：将 4 位二进制码合在一起转为对应的十六进制数码或将一个十六进制数码转换为对应的 4 位二进制码。因此，在计算机领域经常使用十六进制的书写代替二进制的书写。在汇编语言里，十进制用后缀 D（默认为十进制）、二进制用后缀 B、十六进制用后缀 H 表示，不区分大小写。

表 1.1 十六进制数码与二进制数码对应关系

二进制	0000	0001	0010	0011	0100	0101	0110	0111
十六进制	0	1	2	3	4	5	6	7
二进制	1000	1001	1010	1011	1100	1101	1110	1111
十六进制	8	9	A	B	C	D	E	F

二进制、十六进制与十进制都遵循相同的进位计数制规则，差别只在于基数不同。所以，它们的运算方法和我们已知的十进制运算是类似的，只需用“逢二进一”、“逢十六进一”代替“逢十进一”即可。

【例 1.2】 计算 $298H+72H=?$ ， $298H\times 6CH=?$

$$\begin{array}{r} 298H \\ + 72H \\ \hline 30AH \end{array}$$

注意：8+2=10 没有进位，9+7=16 才有进位。

$$\begin{array}{r} 298H \\ \times 6CH \\ \hline 1F20 \\ + F90 \\ \hline 11820H \end{array}$$

1.1.2 数的机器表示

计算机中的数是用二进制表示的。数码 0 和 1 由二值器件的两个稳态来表示，我们称之为二进制位（bit），它是计算机中信息存储的最小单位，一般用 b 表示；8 个相邻的二进制位构成一个字节（Byte），一般用 B 表示。由于机器资源总是有限的，决定了数的机器表示和数学表示是不同的，我们把前者称为机器数，后者称为真值。

1. 定点数

定点数是指小数点位置固定。在微机中，小数点位置固定在最低位，即机器中所有的定点数都是纯整数。

在微机中，对于定点数存在无符号数和有符号数的概念。无符号数表达的是大于等于 0 的数，在机器表示时不需要表示符号信息。有符号数存在符号信息，在机器表示时最高位用来表示符号，“1”表示负，“0”表示正。在计算机中，常用像 8 位、16 位、32 位、64 位等字节整数倍的位数进行定点数的表示，位数越多数的表示范围越大，如表 1.2 所示是无符号数和有符号数的表示方法和表示范围。

表 1.2 无符号数和有符号数的表示方法和表示范围

位 数	无 符 号 数		有 符 号 数	
8 位	8 位全部参与数的表示		最高位表示符号，其余 7 位表示绝对值	
	表示方法	表示范围	表示方法	表示范围
	00000000	0～（2 ⁸ –1=255）	0 0000000	0～（2 ⁷ –1=127）
			0 1111111	
	11111111		1 0000001	–1～（–(2 ⁷ –1)=–127）
			1 1111111	
16 位	16 位全部参与数的表示		最高位表示符号，其余 15 位表示绝对值	
	表示方法	表示范围	表示方法	表示范围
	0000000000000000	0～（2 ¹⁶ –1=65535）	0 000000000000000	0～（2 ¹⁵ –1=32767）
			0 111111111111111	
	1111111111111111		1 000000000000001	–1～（–(2 ¹⁵ –1)=–32767）
			1 111111111111111	

在表 1.2 中，有符号数的表示方法是：最高位是符号位，其余位表示绝对值。这种表示方法称为原码。原码的表示方法常用来描述真值。在计算机中，有符号数的机器表示是补码，补码的表示方法为：

$$[X]_{\text{补}} = \begin{cases} X & X \geq 0 \\ 2^n + X = 2^n - |X| & X < 0, n \text{ 是位数} \end{cases}$$

通俗的描述是：0 和正数的补码就是本身，负数的补码是绝对值求反加 1。

【例 1.3】求 1，46，-1，-46 的 8 位补码。

正数的补码就是本身，所以 1 的补码就是[00000001]_补。

同理，46 的补码是[00101110]_补。

负数的补码是绝对值求反加 1，所以-1 的补码就是[11111111]_补（00000001 求反为 11111110，再加 1 得到 11111111）。

同理，-46 的补码是[11010010]_补（00101110 求反为 11010001，再加 1 得到 11010010）。

我们都知道，时钟转一周是 12 小时，加 1 小时是将时针顺时针拨 1 格，而减 1 小时可以将时针逆时针拨 1 格，也可以顺时针拨 11（即 12-1）格，这样，在时钟上-1 和+11 是一样的。与时钟类似，如果使用 8 位来表示有符号数，它的一周就是 2⁸=256，所以-1 就可以用 255（即 2⁸-1）来表示，这就是补码的含义。

再次强调：微机中有符号数的机器表示是补码。在微机中，将“求反加 1”操作称为求补操作，又称为算术求反（在数的前面加上负号）。补码具有以下特性：

$$[-X]_{\text{补}} = [X]_{\text{补}} \text{ 的求补}$$

$$[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}}$$

$$[X-Y]_{\text{补}} = [X]_{\text{补}} + [-Y]_{\text{补}}$$

由第三个补码特性可以看出，补码的减法运算可以变为加法运算。这样，CPU 中就可以用加法器直接实现减法，而不再需要再设置专门实现减法的部件了。

【例 1.4】用 8 位补码计算 32-7=?

0010 0000
- 0000 0111

0010 0000
+ 1111 1001

进位的1被丢弃 ① 0001 1001

计算中，最高位（符号位）产生的进位超出了机器资源被丢弃。
所以，32-7=25。

8 位补码和 16 位的表示范围如表 1.3 所示。一般来说， n 位补码的表示范围是：

$$-2^{n-1} \leq N \leq 2^{n-1}-1$$

表 1.3 补码表示范围

8 位补码表示范围		16 位补码表示范围	
0 1111111	127	0 111111111111111	$2^{15}-1=32767$
⋮	⋮	⋮	⋮
0 0000000	0	0 000000000000000	0
1 1111111	-1	1 111111111111111	-1
⋮	⋮	⋮	⋮
1 0000000	-128	1 000000000000000	$-2^{15}=-32768$

注意：有符号数在进行位数扩展（8 位扩展到 16 位、16 位扩展为 32 位等）时，正数（符号位为 0）高位补 0，负数（符号位为 1）高位应补 1，称为符号扩展。例如：

$$[00000001]_{补}=[00000000\ 00000001]_{补}=1, [11111111]_{补}=[11111111\ 11111111]_{补}=-1$$

2. 浮点数

任意一个带符号的二进制数 N 使用科学计数法可以表示为 $N=(-1)^S \times 2^E \times M$ ，式中， S 是数的符号； E 是指数（阶码）； M 是尾数。计算机用存储尾数和指数的方式来表达实数，这就是浮点表示法。浮点数利用指数达到了浮动小数点的效果，从而可以灵活地表达更大范围的实数。

浮点数由阶码（指数）和尾数及其符号位组成。其中，尾数用定点小数表示，它给出了实数的有效数字，尾数的位数决定了浮点数的表示精度；阶码用整数形式表示，它指明小数点在数据中的位置，阶码位数决定了浮点数的表示范围。微机使用有限的连续字节保存浮点数，单精度浮点数（32 位）和双精度浮点数（64 位）的格式定义如图 1.1 所示。

IEEE单精度浮点数 $(-1)^S \times (1.M) \times 2^{E-127}$

符号S	阶码E	尾数M
1位	8位	23位

IEEE双精度浮点数 $(-1)^S \times (1.M) \times 2^{E-1023}$

符号S	阶码E	尾数M
1位	11位	52位

图 1.1 浮点数标准格式

1.1.3 字符编码

现代计算机不仅处理数值领域的问题，而且处理大量非数值领域的问题。非数值信息必然需要引入数字、字母、专用符号等抽象符号，这些抽象符号统称为字符。字符在计算机内部也需要以二进制形式表示，称为字符的二进制内码。字符编码规定了字符和其二进制内码的对应关系。

ASCII 码（美国国家信息交换标准字符码）是国际上普遍采用的字符编码。ASCII 码出现于 20 世纪 60 年代末，是为保证人和设备、设备和计算机之间能进行正确的信息交换所编制的统一的信息代码。微机的字符外设一般常采用 ASCII 码。ASCII 码包括 32 个通用控制字符、10 个阿拉伯数字、52 个英文大小字母及 34 个运算符、标点符号和其他专用符号，共 128 个字符，二进制编码需 7 位（占一字节，最高位置 0）。表 1.4 是标准 ASCII 码表。

表 1.4 标准 ASCII 码表

高 3 位 低 4 位	000	001	010	011	100	101	110	111
0000	NUL	DLE	(Space)	0	@	P	‘	p
0001	SOH	DC1		1	A	Q	a	q
0010	STX	DC2	`	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(8	H	X	h	x
1001	HT	EM)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[k	{
1100	FF	FS	,	<	L	“	l	
1101	CR	GS	-	=	M]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

1.2 微机的基本结构

1.2.1 冯·诺依曼结构

虽然微型计算机发展迅速，但至今为止其硬件结构仍采用冯·诺依曼建立的经典结构。这种结构的主要特点是：微型计算机系统的硬件由五大部分组成，如图 1.2 所示。这五部分是：运算器、控制器、存储器、输入设备和输出设备。其中运算器和控制器合称为微处理器 CPU，输入/输出由多个 I/O 接口和外部设备组成。微机的各个组成部分通过总线连接起来。总线由数据总线 DB（Data Bus）、地址总线 AB（Address Bus）和控制总线 CB（Control Bus）

组成。

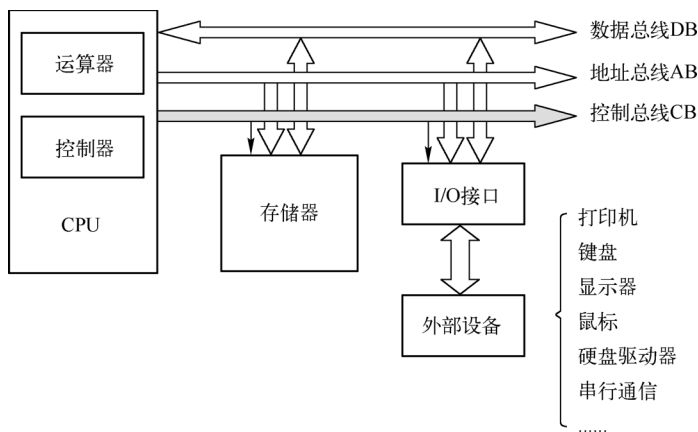


图 1.2 微机硬件结构

1.2.2 微机的硬件组成

1. 中央处理器

中央处理器（Central Processing Unit，CPU）由运算器和控制器组成，是计算机系统核心部件。控制器是整个计算机系统的指挥中心，负责对指令进行分析，并根据指令的要求，有序、有目的地向各个部件发出控制信号，使计算机的各部件协调一致地工作。运算器是对数据进行加工处理的部件，它在控制器的作用下与存储器交换数据，负责进行各类基本的算术运算、逻辑运算和其他操作。

图 1.3 是一个简单 CPU 的模型。CPU 一般都包括控制逻辑、运算器（ALU）、寄存器、暂存器和内部总线等部件。

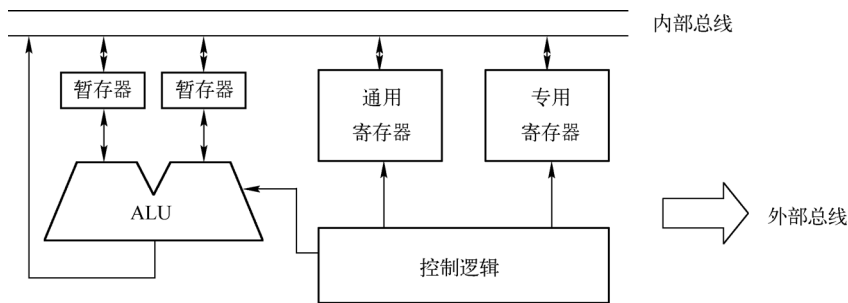


图 1.3 简单 CPU 的模型

CPU 可以完成的一个基本操作称为指令，CPU 中用来计算和控制计算机系统的一套指令的集合称为指令集或指令系统。所以，不管 CPU 内部是如何实现的，它的功能都是由指令集体现的。CPU 可以执行的指令愈多，在一定程度上代表其功能越强大，但也会带来控制复杂等负面影响。

寄存器也是 CPU 的一个重要组成部分，它是 CPU 内部的临时存储单元（与 CPU 速度

相同)。寄存器既可以存放数据和地址，也可以存放控制信息或 CPU 工作的状态信息。为了减少 CPU 访问内存的次数和降低指令控制的复杂度，CPU 的许多指令都是依赖寄存器的，所以 CPU 的寄存器结构也是 CPU 功能的一个重要体现。

字是计算机内部进行数据处理和数据传递的基本单位，其所包含的二进制位数称为字长。字长一般来说和运算器、寄存器、总线宽度是一致的，它是计算机处理精度和运算能力的反映。例如，80386 CPU 字长是 32 位，即 CPU 一次就可以处理 32 位的数据，其能力高于字长是 16 位的 8086 CPU，后者一次仅可以处理 16 位的数据。

运算速度是衡量 CPU 性能的一项重要指标。通常所说的计算机运算速度（平均运算速度），是指每秒钟所能执行的指令条数，一般用“百万条指令 / 秒”（Million Instruction Per Second，MIPS）来描述。

2. 存储器

存储器是计算机系统内最主要的记忆装置。在微机中，CPU 需要执行的程序与需要处理的数据都存放在存储器中。

不管存储器如何组织和实现，从逻辑上可以将其看做一个由存储单元构成的线性表，如图 1.4 所示。通过指定从 0 开始的序号，就可访问相应存储单元。存储单元的序号就是存储单元的地址。

在微机系统中，存储器是按字节编址的，即一个存储单元存储一个字节信息。

存储器的容量和速度对微机系统性能影响是很大的。存储容量是指存储器所能容纳二进制信息的总量，它反映了微机即时存储信息的能力。容量大，能存储的字数就多，能直接接纳和存储的程序就长，计算机的解题能力和规模就大。另一方面，计算机中大量的操作都需要与存储器打交道，因此，存储器的速度往往又是影响计算机数据处理速度的一个主要因素。存取速度通常用存取时间来衡量，指从启动一次存储器操作到完成该操作所经历的时间。

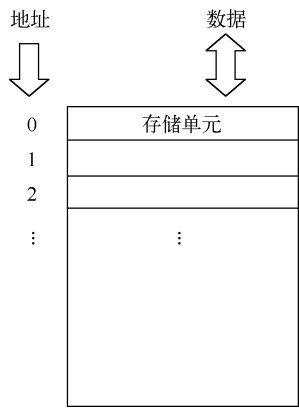


图 1.4 存储器模型

3. I/O 系统

微机的 I/O 系统负责与人或其他系统进行信息交换，由 I/O 接口和外部设备构成。

外部设备包括输入设备和输出设备。常用的输入设备是键盘、鼠标、数字扫描仪及模/数转换器等，它们的作用是获取外部信息，在获取过程中需要把外部信息变换为机器内部所能接收和识别的信息形式。输出设备的作用是把计算机处理的结果变换为人或其他机器设备所能接收和识别的信息形式，常用的输出设备有显示器和打印机等。外部设备，特别是负责与人打交道的人机设备（它们需要识别或展现人所容易理解的信息形式），在功能和实现上存在很大的差异，各式各样，种类繁多。

由于种类繁多且速度各异，所以外部设备不是直接与高速工作的主机（CPU 和存储器）相连接，而是通过 I/O 接口与主机相联系。I/O 接口负责外部设备与主机通信的转换与缓冲，屏蔽外部设备的差异。

外部设备的速度和 CPU 速度存在巨大的差距，由其引发的 I/O 瓶颈一直是微机系统需

要面对的一个重要问题。

4. 总线

总线是连接处理器、存储器、I/O 接口的公共通路。总线最终是一组传递数字信号的物理信通道，可以将其抽象为传送数据、地址、控制三类信息的通路，即总线由数据总线（DB）、地址总线（AB）和控制总线（CB）组成。

以 CPU 访问存储器为例，CPU 将地址送到地址总线上，存储器用地址总线上的地址信号选定存储单元，CPU 再通过控制总线向存储器发出读/写等控制信号，存储器将存储单元的数据送到数据总线上（存储器读）或将 CPU 送到数据总线上的数据写入存储单元（存储器写）。由此可以看出，地址总线的宽度决定了存储空间的大小，即存储单元的数目，如 20 位地址总线决定了存储空间是 $2^{20}=1\text{MB}$ 。

总线每秒能够传送的最大字节数称为总线的传输速率，它是总线工作能力的衡量。不难想象，总线的性能是微机系统中的一个瓶颈问题，因为 CPU 与存储器、I/O 接口的信息交换最终都需要经过总线这一公共通路。

1.3 微机的软件系统

1.3.1 指令和指令系统

指令是 CPU 可以完成的基本操作的抽象，例如传送操作、算术运算操作等，CPU 的设计者一般就把这样的基本操作规定为指令，例如传送指令、算术运算指令等；另一方面，指令是 CPU 可以接受的命令，计算机使用者把要求 CPU 完成的任务组织为一条条的指令，CPU 逐条接受每个命令，完成其相对应的操作，即执行指令。

指令的形式是二进制代码，通常包含两个部分：一部分代码表示操作码；另一部分代码表示操作数。操作码指明指令所要完成的操作的类型或性质，如数据传送、加法运算或数据输出等。操作数则指明操作的数据对象，可以是数据本身也可以是数据所在的存储单元地址。

CPU 所能执行的全部指令就是 CPU 的指令系统，它是 CPU 全部功能的逻辑表现。指令系统可以说是整个计算机体系中最重要概念之一，它是计算机软件系统和硬件系统的“分水岭”。对硬件系统来说，指令系统是硬件实现的全部功能；对软件系统而言，指令系统是软件系统的基础，指令是软件实现的“因子”。在 CPU 的设计者看来，指令系统的设计确定了系统的软件、硬件功能分配，即哪些功能由软件完成，哪些功能由硬件完成，是整个计算机系统设计的出发点。站在软件开发角度，指令系统是自上而下观察而获得的硬件系统的功能模型，是程序设计的基础。

1.3.2 程序和语言

在计算机中，区别于硬件的概念是软件。软件从概念上说是指解决问题所使用的思想、方法、逻辑等无形的东西，这些无形的东西只要在计算机上实现就需要有一定的形式，即程序。

程序是计算机使用者交给计算机执行的软件载体。因此，程序必然体现出两面性，一方面要表现软件思想，即程序设计者的思想；另一方面，它又能为计算机所识别和执行。

所以程序就需要用一种语言表达，这种语言能够让人和计算机进行沟通，我们称为计算机语言。

因为计算机只能识别二进制数码，所以计算机执行的程序也只能是由二进制的指令代码和二进制的数数据所构成，我们称这样的程序为机器语言程序。直接使用指令二进制代码的计算机语言就是机器语言。在计算机发展的初期，程序就是由机器语言编写的，属于机器语言阶段。但是机器语言是由一连串的 0 和 1 组成的，没有什么明显特征，既无法理解也无法记忆，因此编写程序是十分困难而且十分繁琐的工作。

计算机语言的第二个阶段是汇编语言阶段。为了减轻使用机器语言编程的痛苦，人们开始使用助记符来替代指令的二进制代码。助记符一般是指令功能的英文缩写，例如，“传送”使用 MOV（move 的缩写），加法使用 ADD，转移使用 JMP 等。这样，每条指令就有了明显的特征，易于理解和记忆。使用指令助记符构成的计算机语言就是汇编语言。虽然汇编语言仍然直接使用指令编写程序，但毕竟已经跨入了符号语言，它是计算机语言发展史上的一大进步。

1954 年，FORTRAN 的问世标志了高级语言的诞生。高级语言是接近于数学语言或人类自然语言的形式语言，不依赖于特定的计算机结构，具有良好的通用性和可移植性。高级语言的出现极大地拓展了计算机的功能和应用范围。至今，高级语言经历了从早期语言到结构化程序设计语言、从面向过程程序设计到面向对象程序设计的发展过程，出现了几百种高级语言，其中影响较大、使用较普遍的有 FORTRAN，ALGOL，COBOL，BASIC，Pascal，C，C++，Java 等。

如图 1.5 所示展示了程序员使用机器语言、汇编语言和高级语言编程的过程。汇编语言同机器语言一样都是使用指令编写程序的，它们都依赖于具体的计算机，都要求编程者对特

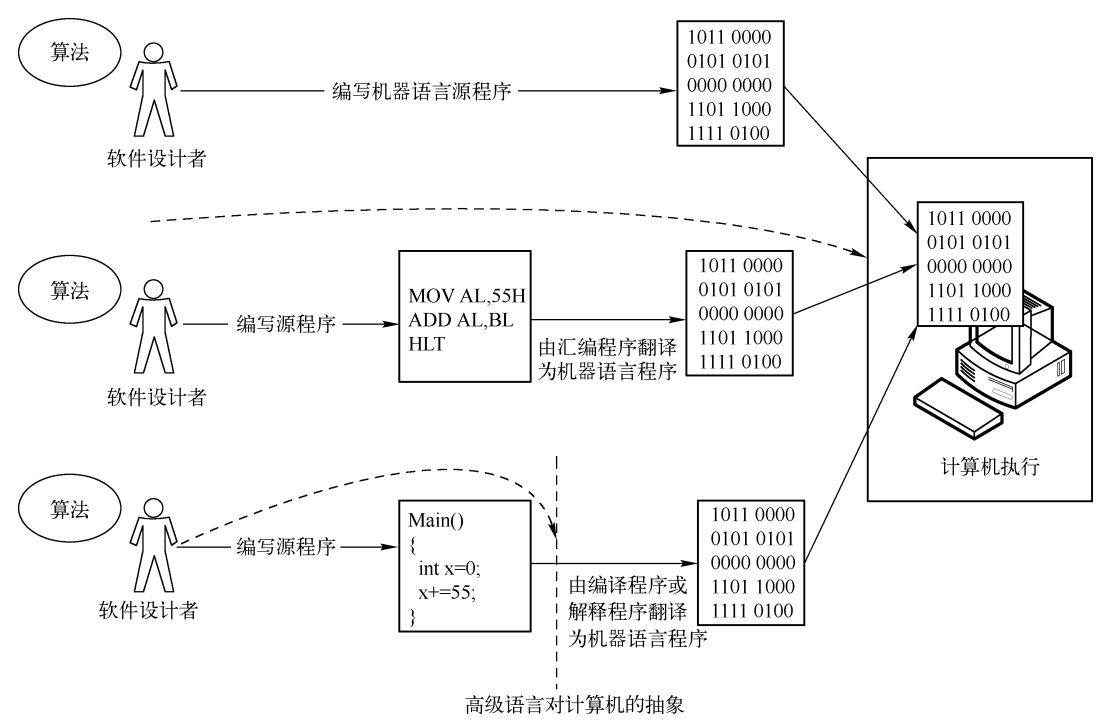


图 1.5 使用机器语言、汇编语言和高级语言编程的过程

定计算机的指令系统、存储结构等具体的机器特性有所了解，所以汇编语言又称为低级语言。使用高级语言的编程者所面对的不是特定的计算机，而是由高级语言对计算机抽象的虚拟计算机，高级语言程序必须由编译程序或解释程序将其翻译为机器语言。

1.3.3 微机软件系统

计算机系统是由硬件系统和软件系统组成的一个复杂的整体。硬件系统和软件系统互相依赖，不可分割。硬件系统是计算机的“躯干”，是物质基础。软件系统则是建立在这个“躯干”上的“灵魂”。按照传统的观点，软件又可分为系统软件和应用软件，如图 1.6 所示。

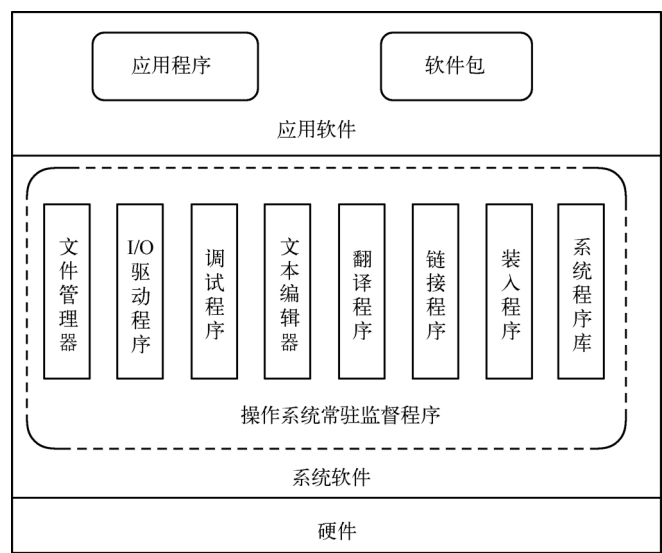


图 1.6 计算机系统的组成

系统软件是指控制计算机的运行，管理计算机的各种资源，并为应用软件提供支持和服

- 务的一类软件。系统软件通常包括以下几种。
- (1) 操作系统
 - 操作系统（Operating System, OS）是管理计算机的各种资源、自动调度用户的各种作业程序、处理各种中断的软件。它是计算机硬件的第一级扩充，是用户与计算机之间的桥梁，是软件体系中最基础和最核心的部分。
 - (2) 语言处理程序
 - 语言处理程序是对各种计算机语言进行汇编或解释、编译的程序，它将计算机不能直接执行的非机器语言源程序，翻译成能直接执行的机器语言。
 - (3) 系统程序库
 - 系统程序库是为进一步扩充计算机功能和方便用户使用而设置的标准子程序库。
 - (4) 工具软件
 - 工具软件是在系统开发和系统维护时使用的工具，完成一些与管理计算机系统资源及文件有关的任务，包括编辑程序、链接程序、计算机测试和诊断程序等。这种程序需要操作系统的支持，而它们又支持软件的开发和维护。
 - 应用软件是用户利用计算机以及各种系统软件编制的解决用户各种实际问题的程序。

一些应用软件经过标准化、模块化，逐步形成为解决某些典型问题的应用程序组合，称为软件包。

由图 1.6 还可以看出软件系统和硬件的关系，根据这种关系，我们通常把一个完整的计算机系统划分成为 4 个层次，从底层到高层分别为硬件层、操作系统层、应用支撑层和应用层。在这种层次结构中，高层可以理解为对低层资源的使用、管理和扩充，低层可以理解为为高层提供支持和服务；每个层次都是相对独立的，具有本层次特定的功能模型和特点。

1.4 微机系统的发展

1.4.1 微机的诞生

1976 年，乔布斯和沃兹尼克成功设计了他们的第一台微型计算机，装在一个木盒子里，它有一块较大的电路板，8KB 的存储器，能发声，且可以显示高分辨率图形。1977 年，沃兹尼克设计了世界上第一台真正的个人计算机——Apple II，如图 1.7 所示。1978 年年初，他们又为 Apple II 增加了磁盘驱动器。

1981 年 8 月 12 日，IBM 正式推出 IBM 5150，如图 1.8 所示，它的 CPU 是 Intel 8088，主频为 4.77MHz，主机板上配置 64KB 存储器，另有 5 个插槽供增加内存或连接其他外部设备用。它还装备了显示器、键盘和两个软磁盘驱动器，而操作系统是微软的 DOS 1.0。IBM 将 5150 型计算机称为 Personal Computer（个人计算机），不久，“个人计算机”的缩写“PC”成为所有个人计算机的代名词。



图 1.7 Apple II 个人计算机



图 1.8 IBM 5150 型个人计算机

1.4.2 微处理器的发展

微型计算机的发展主要表现在其核心部件——微处理器的发展上，每当一款新型的微处理器出现时，就会带动微机系统的其他部件相应发展，如微机体系结构的进一步优化、存储器存取容量的不断增大、存取速度的不断提高、外围设备性能的不断改进以及新设备的不断出现等。

1978 年和 1979 年，Intel 公司先后推出了 8086 芯片和 8088 芯片，它们都是 16 位微处理器，内含 29 000 个晶体管，时钟频率为 4.77MHz，地址总线为 20 位，可使用内存为 1MB。它们的内部数据总线都是 16 位，外部数据总线 8088 芯片是 8 位，8086 芯片是 16 位。1981 年 8088 芯片首次用于 IBM PC 中，开创了全新的微机时代。

1982 年 Intel 公司推出了 80286 芯片，该芯片含有 13.4 万个晶体管，时钟频率由最初的

6MHz 逐步提高到 20MHz。其内部和外部数据总线皆为 16 位，地址总线为 24 位，可寻址 16MB 内存。80286 芯片有两种工作方式：实模式和保护模式。

1985 年 Intel 公司推出了 80386 芯片，它是 80x86 系列中的第一种 32 位微处理器，内含 27.5 万个晶体管，时钟频率为 12.5MHz，后提高到 20MHz、25MHz、33MHz。其内部和外部数据总线都是 32 位，地址总线也是 32 位，可寻址 4GB 内存。它除具有实模式和保护模式外，还增加了一种叫作虚拟 86 的工作方式，可以通过同时模拟多个 8086 处理器来提供多任务能力。

1989 年 Intel 公司推出了 80486 芯片，这种芯片突破了以往 100 万个晶体管的界限，集成了 120 万个晶体管。其时钟频率从 25MHz 逐步提高到 33MHz、50MHz。80486 处理器将 80386 处理器和数学协处理器 80387 以及一个 8KB 的高速缓存集成在一个芯片内，并且在 80x86 系列中首次采用了 RISC 技术，可以在一个时钟周期内执行一条指令。它还采用了突发总线方式，大大提高了与内存数据交换的速度。

Intel 公司于 1993 年又推出了 80586 芯片，其正式名称为 Pentium，即奔腾处理器。Pentium 含有 310 万个晶体管，时钟频率最初为 60MHz 和 66MHz，后提高到 200MHz。Pentium 引起的轰动尚未结束，1995 年 Intel 公司又推出了新一代微处理器——P6。P6 含有 550 万个晶体管，时钟频率为 133MHz，处理速度几乎是 100MHz 的 Pentium 的 2 倍。P6 的一级（片内）缓存为 8KB 指令和 8KB 数据。值得注意的是在 P6 的一个封装中除 P6 芯片外还包括有一个 256KB 的二级缓存芯片，两个芯片之间用高频宽的内部通信总线互连。P6 最引人注目的是具有一项称为“动态执行”的创新技术，这是继 Pentium 在超标量体系结构上实现突破之后的又一次飞跃。

1997 年 Intel 公司又推出了 Pentium II。Pentium II 使用 MMX（多媒体扩展的新指令集，有 57 条新指令，用于高效地处理图形、视频、音频数据）和 AGP（加速图形端口，是由 Intel 公司开发的图形总线技术）技术，其系统总线速度达到 66MHz，一级 Cache 含 16KB 指令 Cache 和 16KB 数据 Cache，二级 Cache 为 512KB，采用了 0.35 μm 的工艺。Pentium II 采用了新的单边接触盒（Single Edge Contact, SEC）封装。

1999 年 2 月，Intel 公司发布了 Pentium III 处理器。Pentium III 的内核与 Pentium II 大致一样，只有新增加了 70 条单指令对数据流扩展（Streaming SIMD Extensions, SSE）指令集，使 CPU 的浮点运算能力得到增强，提高了 CPU 对浮点运算密集型应用程序的执行效率。

Intel 公司于 2002 年推出了 Pentium IV，使得在与 AMD 的竞争中重新获得领先地位。Pentium IV 采用了全新设计的架构，系统总线速度等效于 400MHz，CPU 带宽相比 Pentium III 就有几倍的提升。在指令集方面，Intel 为 Pentium IV 特意设计了 144 条全新的指令，称其为 SSE2。早期的 Pentium IV 依然使用 0.18 μm 工艺生产，后来过渡到 0.13 μm 工艺，主频速度也从 0.18 μm 工艺时的最高 2GHz 提高到拥有最新超线程（Hyper-Threading）技术的 3.06GHz。

微处理器的出现是一次伟大的工业革命，从 20 世纪 70 年代至今，微处理器的发展日新月异。可以说，人类的其他发明的发展速度都没有微处理器那么快、影响也没那么深远。

1.4.3 微机系统的发展

20 世纪 70 年代末出现了微处理器，它比大型机和小型机集成度更高，因而促进了集成

电路技术的发展，这又进一步推动了计算机性能的提高——计算机性能以大约每年 35% 的速度提高。这一发展速度，再加上微处理器批量生产的成本优势，使得计算机产业中以微处理器为基础的部分迅速增长。

此外，计算机市场的两个重大变化使新的计算机系统结构比以往更容易取得商业运作上的成功。其一，实际上人们已经不再使用汇编语言编程，这就降低了对目标代码兼容性的要求。其二，与硬件厂商无关的标准操作系统的出现，降低了推出新系统结构的成本和风险。

这种令人难以置信的高速发展带来了双重效果。

一方面，它极大地增强和完善了计算机提供给用户的功能。现代最高性能的微处理器对很多应用程序的处理效果远远超过了以前的巨型机。

另一方面，这种飞速发展使得以微处理器为基础的计算机在整个计算机设计领域占据了统治地位。工作站和个人计算机已经成为计算机工业的主要产品。由逻辑电路或门阵列制成的小型机已经被微处理器构建而成的服务器取代。大型机也在慢慢地被由流行的微处理器组成的多处理器系统取代，甚至高端产品——巨型机，也可以由多个微处理器构成。

进入 21 世纪后，桌面电脑、服务器、嵌入式系统成为了微机发展的 3 个主要的方向和市场。

习题 1

1. 完成下列数的进制转换。

(1) 133D=? B (2) 01101100B=? H=? D (3) 97D=? H

2. 写出下列各整数的原码和补码 (8 位)。

(1) -99 (2) 33 (3) -1 (4) -128

3. 已知字符 0、A 和 a 的 ASCII 码，如何求出各数字字符和各英文大小写字母字符的 ASCII 码？

4. 通常计算机硬件由_____、输入设备和输出设备五部分组成。

- | | |
|----------------|----------------|
| A. 寄存器、存储器、运算器 | B. 运算器、控制器、存储器 |
| C. 控制器、寄存器、存储器 | D. 控制器、运算器、寄存器 |

5. 微机系统主要的性能指标有哪些？

6. 什么是指令？什么是指令集？

7. 指令系统对微机系统有何意义？

8. 什么是计算机语言？什么是程序？它们和计算机软件是什么关系？

9. 高级语言为什么具有通用性？

10. 简述计算机系统的层次结构。