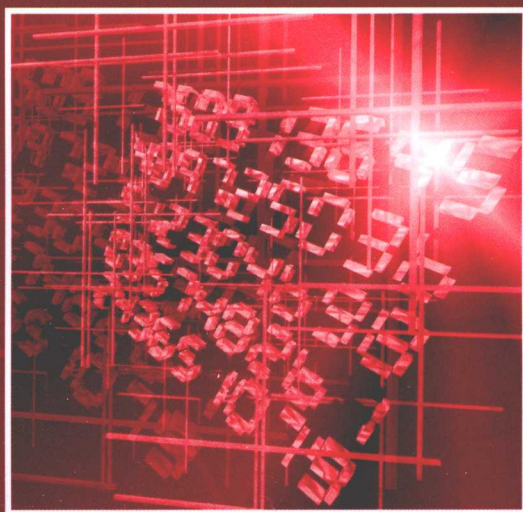




普通高等教育“十一五”计算机类规划教材

# 汇编语言 与计算机系统组成

● 李心广 张建民 潘智刚 等编著



免费  
电子课件

 机械工业出版社  
CHINA MACHINE PRESS

普通高等教育“十一五”计算机类规划教材

# 汇编语言与计算机 系统组成

李心广 张建民 潘智刚 罗海涛 姚敏锋 编 著  
赖声礼 主 审



机械工业出版社

本书将“汇编语言程序设计”、“计算机组成原理”及“计算机体系结构”有机地结合于一体。在保证必要的经典内容的同时,力求反映近代理论和先进技术;在理论与应用关系上,力求实用,以应用为主。

本书共分4篇:第1篇为计算机系统组成基础,内容包括计算机系统概论、计算机中的信息表示与运算方法。第2篇为计算机系统分层结构,内容包括CPU组织、指令系统层、80x86汇编语言程序设计。第3篇为存储系统与输入/输出系统,内容包括存储系统介绍、输入/输出系统、计算机I/O设备。第4篇为计算机系统部件设计。

本书可作为高等院校计算机类、自动控制及电子技术应用等专业的教材;也可作为其他理工电气信息类专业教材;还可供从事相关专业的工程技术人员作为参考书。为了方便教学,本书配有教学课件,欢迎选用本书作为教材的老师登录 [www.cmpedu.com](http://www.cmpedu.com) 下载或发邮件到 [llm7785@sina.com](mailto:llm7785@sina.com) 索取。

## 图书在版编目(CIP)数据

汇编语言与计算机系统组成/李心广等编著. —北京:机械工业出版社, 2009.5

普通高等教育“十一五”计算机类规划教材  
ISBN 978-7-111-26868-0

I. 汇… II. 李… III. ①汇编语言-程序设计-高等学校-教材②计算机体系结构-高等学校-教材 IV. TP313 TP303

中国版本图书馆CIP数据核字(2009)第058730号

机械工业出版社(北京市百万庄大街22号 邮政编码100037)

策划编辑:刘丽敏 责任编辑:郭娟 版式设计:霍永明

责任校对:李婷 封面设计:张静 责任印制:洪汉军

北京市朝阳区展望印刷厂印刷

2009年7月第1版第1次印刷

184mm×260mm·25.5印张·627千字

标准书号:ISBN 978-7-111-26868-0

定价:44.00元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

销售服务热线电话:(010)68326294

购书热线电话:(010)88379639 88379641 88379643

编辑热线电话:(010)88379726

封面无防伪标均为盗版

# 前 言

随着科学技术的不断发展,各学科在教学过程中,都会将本学科最新技术的发展成果增加到教学体系之中。近年来,计算机技术的飞速发展,必然导致与之相关的学科教学内容做出较大幅度的调整。而考虑到以加强学生自主学习、提高学生创新能力为目的的素质教育要求,就必然要减少课堂教学时数。为此,本教材就是为适应这一形势发展而编写的。“汇编语言程序设计”、“计算机组成原理”及“计算机系统结构”是计算机专业的主干课程。这三门课程有相互依赖关系,因此独立开课时难免有重复现象。如果整合则可节省大量的课时。三门课程整合后形成一门课程——“汇编语言与计算机系统组成”。由于该课程涵盖了计算机专业的三门主干课程的教学内容,因此合并决不是简单地将三门课程合起来,而是将合并的原则确定为,既要保证学科的基本知识(保证足够的知识储备,为学生更深入学习该学科提供基础),又要保证知识前后衔接,同时又要将最新的技术融入教学内容中(将大规模集成电路应用技术引入到课程教学之中)。

本书的主要特点如下。

(1) 汇编语言程序设计、计算机组成原理及计算机体系结构三门课程的有机结合。学习汇编语言应理解计算机原理,而学习计算机组成原理应懂得汇编语言,三门课程分开教学,势必造成一些教学内容的重复。将这三门课程有机地组合后,可节省教学课时。

(2) 将原计算机体系结构课程中的两项主要教学内容:“流水线技术”、“存储层次结构”分散于本书的第3章微体系结构层——CPU的构成、第6章存储系统中。避免了计算机组成及计算机体系结构课程内容的重复,这也方便了相关内容的教学。

(3) 将汇编语言程序设计归于计算机系统分层结构中,体现原三门课程的紧密联系。在课程中增加计算机组成部件在大规模集成电路中实现的设计方法,使学生真正体会到现代计算机部件的设计思想。计算机指令集的发展有CISC复杂指令集及RISC精简指令集两个方向。RISC指令集的实现采用组合逻辑电路,而现代组合逻辑电路的实现一般采用VLSI超大规模集成电路实现,因此在教材中加入了在FPGA实现计算机部件的内容。

本书内容结构采用4篇9章的方式:第1篇为计算机系统组成基础,含第1章计算机系统概论,第2章计算机中的信息表示;第2篇为计算机系统分层结构,含第3章微体系结构层——CPU的构成,第4章指令系统层,第5章汇编语言层;第3篇为存储系统与输入/输出系统,含第6章存储系统,第7章输入/输出系统,第8章I/O设备;第4篇为计算机系统部件设计,含第9章现代计算机系统部件设计。

本书可作为高等院校计算机科学与技术、软件工程、网络工程等计算机类专业、自动控制专业以及其他相关专业的教材,也可供从事相关专业的工程技术人员和科研人员作为参考书。

本教材的参考教学学时数为90学时左右,可根据各学校的具体情况调整部分教学内容。

本书编写分工如下:李心广编写第1章,姚敏锋编写第2章,张建民编写第3、7、8章,潘智刚编写第4、5章,罗海涛编写第6、9章。全书由李心广负责修改、统稿、定稿,



张晶负责全书版式的检查。感谢华南理工大学赖声礼教授主持全书的审稿工作。

计算机技术的发展日新月异，教学改革任重道远，编者的能力与这两方面所提出的要求相比，还有很大差距。恳请同行及使用者批评指正，以便再版时修正。

电子教案可与出版社联系，相关资料也可与作者联系，联系邮箱：[lxggu@163.com](mailto:lxggu@163.com)。

作者

# 目 录

## 前言

## 第 1 篇 计算机系统组成基础

第 1 章 计算机系统概论	2
1.1 计算机的基本概念	2
1.1.1 存储程序的工作方式	2
1.1.2 信息的数字化表示	2
1.1.3 计算机体系结构、组成与实现	4
1.2 计算机系统的硬、软件组成	5
1.2.1 计算机硬件系统	5
1.2.2 计算机软件系统	7
1.3 层次结构模型	8
1.3.1 从语言功能角度划分层次结构	8
1.3.2 软、硬件在逻辑上的等价	10
1.4 计算机的工作过程	10
1.4.1 处理问题的步骤	10
1.4.2 计算机的解题过程	12
1.5 微型计算机的主要技术指标	15
1.6 计算机的发展与应用	16
1.6.1 电子计算机的诞生	17
1.6.2 第一代计算机	18
1.6.3 第二代计算机	18
1.6.4 第三代计算机	18
1.6.5 第四代计算机	19
1.6.6 新一代计算机	21
1.6.7 我国计算机的发展	21
思考题与习题	22
第 2 章 计算机中的信息表示	23
2.1 无符号数和有符号数	23
2.1.1 无符号数	23
2.1.2 有符号数	23
2.2 数的定点表示和浮点表示	30
2.2.1 定点表示	31
2.2.2 浮点表示	31
2.2.3 定点数和浮点数的比较	32
2.2.4 举例	33
2.2.5 IEEE 754	34

2.3 定点运算	35
2.3.1 移位运算	35
2.3.2 加法与减法运算	37
2.3.3 乘法运算	40
2.3.4 除法运算	47
2.4 浮点四则运算	50
2.4.1 浮点加减运算	50
2.4.2 浮点乘除运算	55
2.4.3 浮点运算所需的硬件配置	57
2.5 算术逻辑单元	58
2.5.1 ALU 电路	58
2.5.2 快速进位链	60
2.6 字符的表示	65
2.6.1 ASCII 码	65
2.6.2 Unicode 编码	67
2.6.3 汉字编码简介	68
2.7 指令信息的表示	70
2.7.1 指令格式	71
2.7.2 常用的寻址方式	72
2.7.3 指令类型	74
2.7.4 Pentium 指令格式	76
2.8 校验技术	78
2.8.1 奇偶校验码	79
2.8.2 循环冗余校验码	79
思考题与习题	82

## 第 2 篇 计算机系统分层结构

### 第 3 章 微体系结构层——CPU

的构成	86
3.1 CPU 的组成和功能	86
3.1.1 CPU 的组成	86
3.1.2 CPU 的功能	87
3.1.3 指令的执行过程	88
3.2 CPU 模型机的数据通路及 指令流程分析	88
3.2.1 单总线结构	88
3.2.2 双总线结构	90
3.2.3 三总线结构	92



3.3 组合逻辑控制器原理 .....	94	5.2 汇编语言语句类型及格式 .....	167
3.3.1 模型机的指令系统 .....	94	5.2.1 语句类型 .....	167
3.3.2 模型机的时序系统与控制方式 .....	96	5.2.2 语句格式 .....	167
3.3.3 模型机主要组成部分的门级设计 及控制信号 .....	98	5.3 80x86 宏汇编伪指令 .....	170
3.3.4 指令流程及控制信号序列 .....	101	5.3.1 符号定义伪指令 .....	170
3.3.5 组合逻辑控制器的设计步骤 .....	103	5.3.2 数据定义伪指令 .....	171
3.4 微程序控制器原理 .....	105	5.3.3 指令集选择伪指令 .....	175
3.4.1 微程序控制的基本概念 .....	105	5.3.4 段结构伪指令 .....	175
3.4.2 微指令的编码方式 .....	107	5.3.5 定位伪指令 .....	177
3.4.3 微程序控制器中的分支控制 .....	108	5.3.6 过程定义伪指令 .....	178
3.4.4 微指令格式 .....	110	5.3.7 标题伪指令 .....	178
3.4.5 模型机的微指令格式设计及 微程序编写 .....	110	5.3.8 结束伪指令 .....	178
3.5 精简指令系统计算机 .....	111	5.3.9 包含伪指令 .....	179
3.5.1 RISC 与 CISC 的概念 .....	112	5.3.10 简化段定义伪指令 .....	179
3.5.2 精简指令系统计算机的 技术特点 .....	112	5.4 宏指令 .....	180
3.6 指令流水技术 .....	113	5.5 汇编语言程序的设计步骤 .....	185
3.6.1 流水线的基本概念 .....	113	5.6 汇编语言程序设计的基本技术 .....	188
3.6.2 DLX 流水线 .....	113	5.6.1 顺序结构程序设计 .....	188
3.6.3 流水线的效率 .....	115	5.6.2 分支结构程序设计 .....	190
3.6.4 流水线中的相关 .....	115	5.6.3 循环结构程序设计 .....	201
思考题与习题 .....	118	5.6.4 子程序设计 .....	210
<b>第4章 指令系统层</b> .....	120	5.6.5 系统功能的调用 .....	226
4.1 概述 .....	120	思考题与习题 .....	229
4.2 80x86 CPU .....	122	<b>第3篇 存储系统与输入/输出系统</b>	
4.2.1 8086/8088 微处理器 .....	122	<b>第6章 存储系统</b> .....	234
4.2.2 80386 微处理器 .....	123	6.1 概述 .....	234
4.2.3 Pentium 微处理器 .....	124	6.1.1 存储器的分类 .....	234
4.2.4 Pentium IV 微处理器 .....	125	6.1.2 主存的主要技术指标 .....	236
4.3 基本执行环境 .....	127	6.2 存储原理 .....	237
4.3.1 操作模式 .....	127	6.2.1 半导体存储器的存储原理 .....	237
4.3.2 存储管理 .....	128	6.2.2 磁表面存储器的存储原理 .....	243
4.3.3 80x86 CPU 的寄存器结构 .....	133	6.2.3 光存储器的存储原理 .....	247
4.4 80x86 CPU 的指令系统 .....	136	6.3 主存储器的组织 .....	249
4.4.1 80x86 数据类型 .....	136	6.3.1 主存储器的逻辑设计 .....	249
4.4.2 80x86 指令格式 .....	138	6.3.2 主存储器与 CPU 的连接 .....	256
4.4.3 80x86 寻址方式 .....	138	6.3.3 Pentium CPU 的存储器组织 .....	260
4.4.4 80x86 CPU 指令的分类 .....	141	6.3.4 高级 DRAM .....	260
思考题与习题 .....	162	6.4 高速缓冲存储器 .....	267
<b>第5章 汇编语言层</b> .....	166	6.4.1 Cache 的基本原理 .....	267
5.1 概述 .....	166	6.4.2 Cache 的组织结构 .....	271
		6.4.3 Cache 的性能分析 .....	272
		6.4.4 改进 Cache 性能的方法 .....	273



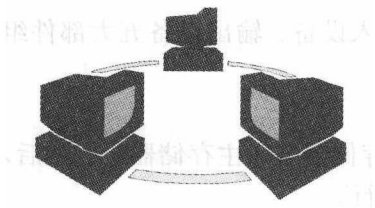
6.4.5 Pentium II CPU 的 Cache 组织结构 .....	283	传送过程 .....	326
6.5 外部存储器 .....	284	7.5.3 DMA 接口的类型 .....	328
6.5.1 硬磁盘存储器 .....	284	7.6 通道与 IOP .....	330
6.5.2 U 盘存储器 .....	288	7.6.1 通道 .....	330
6.5.3 光盘存储器 .....	290	7.6.2 IOP 与外围处理机 .....	333
6.5.4 磁带存储器 .....	291	7.7 系统总线 .....	334
6.6 物理存储系统的组织 .....	293	7.7.1 总线的概念与特性 .....	334
6.6.1 存储系统的层次结构 .....	293	7.7.2 总线的分类 .....	335
6.6.2 磁盘冗余阵列 .....	295	7.7.3 总线的性能指标与总线标准 .....	336
6.6.3 并行存储技术 .....	299	7.7.4 总线使用的控制 .....	338
6.7 虚拟存储系统的组织 .....	303	7.8 典型的外设接口 .....	340
6.7.1 虚拟存储器 .....	303	7.8.1 ATA 接口 .....	340
6.7.2 虚拟存储器的组织方式 .....	304	7.8.2 USB 接口 .....	341
6.7.3 Pentium CPU 支持的虚拟 存储器 .....	307	思考题与习题 .....	342
思考题与习题 .....	308	<b>第 8 章 I/O 设备</b> .....	345
<b>第 7 章 输入/输出系统</b> .....	311	8.1 概述 .....	345
7.1 I/O 系统概述 .....	311	8.1.1 I/O 设备的分类 .....	345
7.1.1 I/O 系统的组成 .....	311	8.1.2 主机对 I/O 控制的 4 个层次 .....	346
7.1.2 设计 I/O 系统应考虑的三个 要素 .....	311	8.2 键盘 .....	346
7.1.3 主机与 I/O 设备间的连接方式 .....	312	8.2.1 键开关与键盘类型 .....	346
7.2 I/O 接口 .....	313	8.2.2 编码键盘的工作原理 .....	347
7.2.1 I/O 接口的功能和组成 .....	313	8.2.3 非编码键盘的工作原理 .....	348
7.2.2 I/O 端口的编址方式 .....	314	8.3 显示器 .....	349
7.2.3 接口的分类 .....	315	8.3.1 CRT 显示器 .....	349
7.3 直接程序控制方式 .....	316	8.3.2 液晶显示器 .....	353
7.3.1 立即程序传送方式 .....	316	8.3.3 等离子显示器 .....	355
7.3.2 程序查询方式 .....	316	8.4 打印机 .....	356
7.3.3 直接程序控制方式适应的 场合及缺点 .....	318	8.4.1 针式打印机 .....	356
7.4 程序中中断方式 .....	318	8.4.2 激光打印机 .....	358
7.4.1 程序中中断方式的基本思想 和作用 .....	318	8.4.3 喷墨打印机 .....	360
7.4.2 I/O 接口中的中断逻辑及中断 驱动程序流程 .....	319	思考题与习题 .....	364
7.4.3 中断请求和中断判优 .....	321	<b>第 4 篇 计算机系统部件设计</b>	
7.4.4 中断响应 .....	322	<b>第 9 章 现代计算机系统部件设计</b> .....	366
7.4.5 多重中断与中断屏蔽 .....	323	9.1 现代计算机设计技术 .....	366
7.5 DMA 方式 .....	325	9.1.1 EDA 技术 .....	366
7.5.1 DMA 接口组成 .....	325	9.1.2 可编程逻辑器件 .....	367
7.5.2 DMA 的数据传送方式与		9.1.3 硬件描述语言 .....	368
		9.2 计算机组成部件设计 .....	368
		9.2.1 算术逻辑运算部件的设计 .....	368
		9.2.2 简单计算机结构设计 .....	378
		思考题与习题 .....	396
		<b>参考文献</b> .....	397



# 第 1 篇

## 计算机系统组成基础

本篇介绍计算机系统概论，计算机中信息的表示与运算方法。



# 第 1 章 计算机系统概论

本章介绍计算机系统组成的基本概念，计算机中的信息表示，计算机系统的硬、软件组成，计算机的层次结构，计算机的工作过程，数字计算机的特点与性能指标，计算机的发展与应用。

## 1.1 计算机的基本概念

电子计算机是一种不需要人工直接干预，能够自动、高速、准确地对各种信息进行处理和存储的电子设备。电子计算机从总体上来说可以分为两大类：电子模拟计算机和电子数字计算机。电子模拟计算机中处理的信息是连续变化的物理量，运算的过程也是连续的；而电子数字计算机中处理的信息在时间上是离散的数字量，运算的过程是不连续的。通常所说的计算机都是指电子数字计算机。

### 1.1.1 存储程序的工作方式

计算机系统由硬件系统和软件系统两大部分组成。美籍匈牙利科学家冯·诺依曼（John von Neumann）奠定了现代计算机的基本结构，其特点是：

- 1) 使用单一的处理部件来完成计算、存储以及通信的工作。
- 2) 存储单元是定长的线性组织。
- 3) 存储空间的单元是直接寻址的。
- 4) 使用低级机器语言，指令通过操作码来完成简单的操作。
- 5) 对计算进行集中的顺序控制。
- 6) 计算机硬件系统由运算器、存储器、控制器、输入设备、输出设备五大部件组成，并规定了它们的基本功能。
- 7) 以二进制形式表示数据和指令。
- 8) 在执行程序和处理数据时必须将程序和数据从外存储器装入主存储器中，然后才能使计算机在工作时能够自动从存储器中取出指令并加以执行。

这就是存储程序概念的基本原理。

### 1.1.2 信息的数字化表示

计算机除了处理数字信息外，还必须处理用于组织、控制或表示数据的字母、符号、汉字以及控制符号等。

#### 1. 以二进制表示数据

电子计算机处理的信息，都是仅用“0”与“1”两个简单数字表示的信息，或者是用这种数字进行了编码的信息。这种数制叫做二进制。

二进制的优点是：二进制只有“0”和“1”两个数字，很容易表示。电压的高和低、

晶体管的截止与饱和、磁性材料的磁化方向等都可以表示为“0”和“1”两种状态。

二进制数的每一位只有0和1两种状态，只需要两种设备就能表示，所以二进制数节省设备。由于状态简单，所以它抗干扰能力强，可靠性高。

## 2. 字符编码

计算机中的大多数I/O信息是非数字的字符，须按特定的规则用二进制编码在计算机中表示。国际上普遍采用了一些标准代码，如ASCII、EBCDIC（扩展的BCD）等。

ASCII（美国标准信息交换码）由7位二进制数编码字符集组成，共有128个编码，包括大小写字母（各26个）、十进制数码字（10个）、运算符和标识符（33个，如“空格”、“\*”等）、控制符（33个，如“LF”、“CR”等）。例如，字符“Yes OK”可由ASCII表示为“59 65 73 20 4F 48 H”（十六进制）或“89 101 115 32 79 75”（十进制）。又例如，使打印机的打印头或显示器的显示光标返回到第一列的控制符“CR”（0DH），而控制符“LF”（0AH）为换行到下一行。上述控制符表示设备的控制功能，还有部分控制字符用于数据通信的连接。但不同的计算机系统对这些控制字符可分别定义为不同的功能。

一般计算机用一个字节表示一个字符，对7位ASCII，最高位一般取0，但也可以在该位设置特殊位，或编码附加的非ASCII字符（如德、法、俄文字母、制表符号、图形等），或作奇偶校验位，这取决于系统和程序。

EBCDIC编码是用8位二进制数表示一个字符，可以表示256个符号，包括控制符、运算符、标识符、大小写字母、数字等。IBM公司在它的各类大型机上广泛采用EBCDIC编码。例如，0~9的EBCDIC编码为“F0~F9 H”，A为0C1H，a为81H，“LF”为25H。

## 3. 十进制数编码

十进制数可转化成二进制数进行处理，但不适用于大量I/O、存储及其处理的场合。这就提出了用二进制数表示一位十进制数字的编码方法，即BCD码（Binary Coded Decimal）表示法，它既具有二进制的形式，又具有十进制的特点。常用的是从4位二进制数的16种不同状态中选出10个表示十进制数0~9，可有多种选择的方法。

(1) 8421码（BCD）。8421码是一种有权码，从高位开始位权分别是8、4、2、1。特点为：

1) 与ASCII码之间转换简单。

0~9的8421BCD是0~9ASCII的低4位，两者相差30H。

例：“234”转换为对应的BCD码为234。

(ASCII) 00110010 00110011 00110100 → (BCD) 0010 0011 0100

2) 加减运算要修正运算结果。

若运算结果产生进位/借位或出现>9的数字（A~FH）时，要对结果进行加6/减6处理。例：9+7=16

$$\begin{array}{r}
 1001 \\
 +0111 \\
 \hline
 10\ 000 \quad \text{有进位, 加6修正} \\
 +110 \\
 \hline
 1\ 0110
 \end{array}$$



8421 BCD 是最常见的一种 BCD，我们常常直接把 8421 码称为 BCD。现在许多计算机都设计有 BCD 修正指令。

(2) 其他有权 BCD。其他有权 BCD 有 2421 码、5211 码、8421 码和 4311 码等。其特点是各种编码的二进制数位的位权是固定的。例如，5211 码的位权分别为 5、2、1、1，即数 8 的 5211 码为 1101。

### 1.1.3 计算机体系结构、组成与实现

计算机体系结构 (Computer Architecture) 这个词目前已被广泛使用。Architecture 本来用在建筑方面，译为“建筑学、建筑术、建筑样式、构造、结构”等。这个词被引入计算机领域后，最初的译法也各有不同，后来趋向译为“体系结构”，但关于它的定义仍未统一。

经典的“计算机体系结构”定义是 1964 年 C. M. Amdahl 在介绍 IBM 360 系统时提出的：计算机体系结构是程序员所看到的计算机的属性，即概念性结构与功能特性。

按照计算机系统的多级层次结构，不同级的程序员所看到的计算机具有不同的属性。例如，传统机器语言程序员所看到的计算机主要属性是该机指令集的功能特性。而高级语言虚拟机程序员所看到的计算机主要属性是该机所配置的高级语言所具有的功能特性。显然，不同的计算机系统，从传统机器级程序员或汇编语言程序员来看，是具有不同属性的。但是，从高级语言（如 Visual Basic）程序员来看，它们就几乎没有什么差别，是具有相同属性的。或者说，这些传统机器级所存在的差别是高级语言程序员所“看不见”的，也是不需要他们知道的。在计算机技术中，对这种本来是存在的事物或属性，但从某种角度看又好像不存在概念称为透明性 (Transparency)。通常，在一个计算机系统中，低层机器的属性对高层机器的程序员往往是透明的，如传统机器级的概念性结构和功能特性，对高级语言程序员来说是透明的。由此看出，在层次结构的各个级上都有它的体系结构。Amdahl 提出的体系结构是指传统机器级的体系结构，即一般所说的机器语言程序员所看到的传统机器级所具有的属性。这些属性是机器语言程序设计者（或者编译程序生成系统）为使其所设计（或生成）的程序能在机器上正确运行，所需遵循的计算机属性，包含其概念性结构和功能特性两个方面。目前，对于通用寄存器型机器来说，这些属性主要是指：

- (1) 数据表示（硬件能直接辨认和处理的数据类型）。
- (2) 寻址规则（包括最小寻址单元、寻址方式及其表示）。
- (3) 寄存器定义（包括各种寄存器的定义、数量和使用方式）。
- (4) 指令集（包括机器指令的操作类型和格式、指令间的排序和控制机构等）。
- (5) 中断系统（中断的类型和中断响应硬件的功能等）。
- (6) 机器工作状态的定义和切换（如管态和目态等）。
- (7) 存储系统（主存容量、程序员可用的最大存储容量等）。
- (8) 信息保护（包括信息保护方式和硬件对信息保护的支持）。
- (9) I/O 结构（包括 I/O 连接方式、处理机/存储器与 I/O 设备间数据传送的方式和格式以及 I/O 操作的状态等）。

这些属性是计算机系统中由硬件或固件完成的功能，程序员在了解这些属性后才能编出在传统机器上正确运行的程序。因此，经典的计算机体系结构概念的实质是计算机系统中

软、硬件界面的确定，其界面之上是软件的功能，界面之下是硬件和固件的功能。

这里比较全面地介绍了经典的计算机体系结构概念。随着计算机技术的发展，计算机体系结构所包含的内容是不断变化和发展的。目前经常使用的是广义的计算机体系结构的概念，它既包括了经典的计算机体系结构的概念范畴，又包括了对计算机组成和计算机实现技术的研究。

因此，计算机体系结构是程序设计者所看到的计算机系统的属性，是计算机的外特性、概念性结构和功能特性，其研究计算机系统的硬、软件的功能划分及接口关系。

计算机组成是指计算机各功能部件的内部构造和相互之间的联系（部件配置、相互连接和作用），强调各功能部件的性能参数相匹配，实现机器指令级的各种功能和特性，是计算机系统结构的逻辑实现。

计算机组成的物理实现，即把一台完成逻辑设计的计算机真正地制作出来，解决各部件的物理结构、器件选择、电源供电、通风与冷却、装配与制造工艺等各个方面的问题。

设计一种新型计算机系统首先面临的问题是什么呢？我们会列举出很多因素，其中最主要的有新型计算机的主要特点和性能。它们具体包括指令集设计、功能组织、逻辑设计、实现技术等。实现技术包括集成电路设计、制造和封装技术、系统制造、供电、冷却技术等。另外，我们往往要求在限定的造价范围内，使新型计算机具有最高的性能。如何采用先进的计算机体系结构和生产技术，制造出具有高性能价格比的计算机系统，是所有通用计算机设计的共同目标。

## 1.2 计算机系统的硬、软件组成

一个完整的计算机系统应包括硬件系统和软件系统。计算机的硬件是计算机的物质基础。软件是发挥计算机功能，使计算机能正常工作的程序。

### 1.2.1 计算机硬件系统

硬件通常是指构成计算机的设备实体。一台计算机的硬件系统应由五个基本部分组成：运算器、控制器、存储器、输入和输出设备。这五大部分通过系统总线完成指令所传达的操作。当计算机在接收指令后，由控制器指挥，将数据从输入设备传送到存储器存放，再由控制器将需要参加运算的数据传送到运算器，由运算器进行处理，处理后的结果由输出设备输出。

#### 1. 中央处理器

中央处理器（Central Processing Unit, CPU）由控制器、运算器和寄存器组成，通常集成在一块芯片上，是计算机系统的核心设备。计算机以 CPU 为中心，输入和输出设备与存储器之间的数据传输和处理都通过 CPU 来控制执行。微型计算机的中央处理器又称为微处理器。

#### 2. 控制器

控制器是对输入的指令进行分析，并统一控制计算机的各个部件完成一定任务的部件。它一般由指令寄存器、状态寄存器、指令译码器、时序电路和控制电路组成。计算机的工作方式是执行程序，程序就是为完成某一任务所编制的特定指令序列，各种指令操作按一定的



时间关系有序安排。控制器产生各种最基本的不可再分的微操作的命令信号，即微命令，以指挥整个计算机有条不紊地工作。当计算机执行程序时，控制器首先从指令指针寄存器中取得指令的地址，并将下一条指令的地址存入指令寄存器中，然后从存储器中取出指令，由指令译码器对指令进行译码后产生控制信号，用以驱动相应的硬件完成指定操作。简言之，控制器就是协调指挥计算机各部件工作的元件，它的基本任务就是根据各种指令的需要综合有关的逻辑条件与时间条件产生相应的微命令。

### 3. 运算器

运算器又称为算术逻辑单元 (Arithmetic Logic Unit, ALU)。运算器的主要任务是执行各种算术运算和逻辑运算。算术运算是指各种数值运算，比如加、减、乘、除等。逻辑运算是进行逻辑判断的非数值运算，比如与、或、非、比较、移位等。计算机所完成的全部运算都是在运算器中进行的，根据指令规定的寻址方式，运算器从存储器或寄存器中取得操作数，进行计算后，送回到指令所指定的寄存器中。运算器的核心部件是加法器和若干个寄存器，加法器用于运算，寄存器用于存储参加运算的各种数据以及运算后的结果。

### 4. 存储器

存储器分为内存储器 (也称为内存或主存) 和外存储器 (也称为外存或辅存)。外存储器一般也可作为输入/输出设备。计算机把要执行的程序和数据存入内存中，内存一般由半导体存储器构成。半导体存储器可分为三大类：随机存储器、只读存储器、特殊存储器。

(1) 随机存储器。随机存储器 (Random Access Memory, RAM) 的特点是可以读写，存取任一单元所需的时间相同，通电时存储器内的内容可以保持，断电后，存储的内容立即消失。RAM 可分为动态 (Dynamic RAM) 和静态 (Static RAM) 两大类。所谓动态随机存储器是用 MOS 电路和电容来作存储元件的。由于电容会放电，所以需要定时充电以维持存储内容的正确，例如每隔 2ms 刷新一次，因此称之为动态存储器。所谓静态随机存储器是用双极型电路或 MOS 电路的触发器来作存储元件的，它没有电容放电造成的刷新问题。只要有电源正常供电，触发器就能稳定地存储数据。DRAM 的特点是集成密度高，主要用于大容量存储器。SRAM 的特点是存取速度快，主要用于高速缓冲存储器。

(2) 只读存储器 (Read Only Memory, ROM) 只能读出原有的内容，不能由用户再写入新内容。原来存储的内容是由厂家一次性写入的，并永久保存下来。ROM 可分为可编程 (Programmable) ROM、可擦除可编程 (Erasable Programmable) ROM、电擦除可编程 (Electrically Erasable Programmable) ROM。例如，EPROM 存储的内容可以通过紫外光照射来擦除，这使它的内容可以反复更改。

(3) 特殊存储器包括电荷耦合存储器、磁泡存储器、电子束存储器等，它们多用于特殊领域内的信息存储。

此外，描述内、外存储容量的常用单位如下。

1) 位/比特 (bit)：这是内存中最小的单位，二进制数序列中的一个 0 或一个 1 就是一个比特，在计算机中，一个比特对应着一个晶体管。

2) 字节 (Byte, B)：它是计算机中最常用、最基本的存储单位。一个字节等于 8 个比特，即 1 Byte = 8bit。

3) 千字节 (Kilo Byte, KB)：计算机的内存容量都很大，一般都是以千字节作单位来表示。1KB = 1024B。

4) 兆字节 (Mega Byte, MB): 20 世纪 90 年代流行的微机的硬盘和内存等一般都是以兆字节为单位。1MB = 1024KB。

5) 吉字节 (Giga Byte, GB): 目前市场流行的微机的硬盘已经达到 80GB、120GB、250GB 等规格。1GB = 1024MB。

6) 太字节 (Tera Byte, TB): 1TB = 1024 GB。

### 5. 输入/输出设备

输入设备用来接收用户输入的原始数据和程序, 并将它们变为计算机能识别的二进制存入到内存中。常用的输入设备有键盘、鼠标、扫描仪、光笔等。

输出设备用于将存储在内存中的由计算机处理的结果转变为人们能接受的形式输出。常用的输出设备有显示器、打印机、绘图仪等。

### 6. 总线

总线是一组为系统部件之间传送数据的公用信号线。它具有汇集与分配数据信号、选择发送信号的部件与接收信号的部件、总线控制权的建立与转移等功能。一般按信号类型将总线分为三组, 即地址总线 (Address Bus, AB)、数据总线 (Data Bus, DB) 和控制总线 (Control Bus, CB)。

## 1.2.2 计算机软件系统

软件是指计算机系统中使用的各种程序及其文档。程序是对计算任务的处理对象和规则的描述, 文档是为了便于了解程序所需要的阐述性资料。

### 1. 软件的作用

计算机的工作是由存储在其内部的程序指挥的, 这是冯·诺依曼计算机系统的重要特色, 因此程序或者软件质量的好坏将极大地影响计算机性能的发挥, 特别是并行处理技术以及 RISC 计算机的出现使软件显得更加重要。软件的具体作用如下:

(1) 它在计算机系统中起着指挥和管理的作用。计算机系统中有各种各样的软、硬件资源, 必须由相应的软件 (特别是操作系统) 来统一管理和指挥。

(2) 它是计算机用户和硬件的接口界面。用户要使用计算机, 必须编制程序使用软件, 用户主要通过软件与计算机进行交流。

(3) 它是计算机体系结构设计的主要依据。为了方便用户, 使计算机系统具有较高的总体效率, 在设计计算机时必须考虑软件和硬件的结合, 以及用户对软件的要求。

### 2. 软件的发展过程

软件的发展受计算机硬件发展和应用的推动与制约, 其发展过程大致分三个阶段。从第一台计算机上的第一个程序出现到实用的高级语言出现为第一阶段 (1946—1956)。这时计算机的应用以科学计算为主, 计算量较大, 但输入/输出量不大。机器以 CPU 为中心, 存储器较小, 编制程序工具为机器语言, 突出问题是程序设计与编制工作复杂、繁琐、易出错。因此重点考虑程序本身, 使它占用内存小, 节省运行时间, 从而提高效率。这时尚未出现“软件”一词。

从实用的高级程序设计语言出现到软件工程出现以前为第二阶段 (1956—1968)。这时除了科学计算外, 出现了大量数据处理问题, 计算量不大, 但输入/输出量较大。机器结构转向以存储器为中心, 出现了大容量存储器, 输入/输出设备增加。为了充分利用这些资源,



出现了操作系统；为了提高程序人员工作效率，出现了高级语言；为了适应大量的数据处理，出现了数据库及其管理系统。这时人们也认识到文档的重要性，出现了“软件”一词。随着软件复杂性的不断提高，甚至出现了人们难以控制的局面，即所谓软件危机。为了克服危机，人们采取了多种方法，特别值得一提的是“软件工程”方法的出现。

软件工程出现以后迄今一直为第三阶段（1968—）。对于一些复杂的大型软件，采用个体或者合作的方式进行开发不仅效率低、可靠性差，且很难完成，必须采用工程方法才能适用。为此，从20世纪60年代末开始，软件工程得到了迅速的发展，还出现了“计算机辅助软件”、“软件自动化”实验系统等。目前，人们除了研究改进软件传统技术外，还在着重研究以智能化、自动化、集成化、并行化、以及自然化为标志的软件新技术。

### 3. 软件的分类

按功能分，软件大致可以分为三类：系统软件、支撑软件和应用软件。

(1) 系统软件：包括操作系统和各类语言的编译程序，它位于计算机系统中最接近硬件的层，其他软件只有通过系统软件才能发挥作用，系统软件与具体应用无关。

1) 操作系统：管理整个计算机系统的软、硬件资源，包括对它们进行调度、管理、监视、服务等，以改善人机界面，并提供对应用软件的支持。按功能分，操作系统可以分成多种类型，包括单用户操作系统和批处理操作系统；分时操作系和实时操作系统；网络操作系统、分布式操作系统和并行操作系统等。

2) 编译程序：把由程序人员编写的各类高级语言书写的程序翻译成能与之等价的、可执行的机器语言代码。

(2) 支撑软件：它是支撑其他软件的开发与维护的软件。数据库管理系统、各类子程序库以及网络软件等均为支撑软件。20世纪70年代中、后期发展起来的软件开发环境则是支撑软件的代表，它主要包括环境数据库、各类接口软件和工具组。

(3) 应用软件：各类用户为满足各自的需要开发的各种应用程序。例如，为进行数据处理、科学计算、事务管理、工程设计以及过程控制所编写的各类应用程序。

软件是人类开发的各种程序和书写的文档。它是智力产品，而且随着硬件技术的不断发展和应用要求的日益提高，软件产品越来越复杂、庞大，如何来保证软件的准确性、友善性、高效率以及智能化，是软件工作者始终努力的目标。

## 1.3 层次结构模型

### 1.3.1 从语言功能角度划分层次结构

现代计算机系统是由软件和硬/固件组成的十分复杂的系统。为了对这个系统进行描述、分析、设计和使用，人们从不同的角度提出了观察计算机的观点和方法。本节从计算机语言的角度，把计算机系统按功能划分成多级层次结构。

随着计算机系统的发展，人们已设计出一系列语言。从面向机器的语言（如机器语言、汇编语言），到各种高级程序设计语言（如Java、C/C++），到各种面向问题的语言或者叫应用语言（如面向数据库查询的SQL语言，面向数字系统设计的VHDL语言，面向人工智能的Prolog语言）。计算机语言就是这样由低级向高级发展，高级语言的语句相对于低级语



言功能更强,更便于应用,但又都是以低级语言为基础的。

计算机语言可分成一系列的层次(Level)或级,最低层语言的功能最简单,最高层语言的功能最强。对于用某一层语言编写程序的程序员来说,他一般不管其程序在机器中是如何执行的,只要程序正确,他总能得到预期的结果。这样,对这层语言的程序员来说,他似乎有了一种新的机器,这层语言就是这种机器的机器语言,该机器能执行用该层语言编写的全部程序。因此计算机系统就可以按语言的功能划分成多级层次结构。每一层以一种不同的语言为特征。这样,可以把现代计算机系统画成如图 1.1 所示的层次结构。

图中第四级以上完全由软件实现。我们称由软件实现的机器为虚拟机器(Virtual Machine),以区别于由硬件或固件实现的实际机器。第一级是微程序机器级,这一级的机器语言是微指令集,程序员用微指令编写的微程序一般是直接由硬件解释实现的。

第二级是一般机器级。这一级的机器语言是该机的指令集,程序员用机器指令集编写的程序可以由微程序进行解释。这个解释程序运行在第一级上。微程序解释指令集又称为仿真(Emulation),实际上,在第一级可以有一个或数个能够在它上面运行的解释程序,每一个解释程序都定义了一种指令集。因此,可以通过仿真在一台机器上实现多种指令集。

计算机系统中也可以没有微程序机器级。在这些计算机系统中是用硬件直接实现传统机器的指令集,而不必由任何解释程序进行干预。目前使用的 RISC 技术就是采用这样的设计思想,处理器的指令集全部用硬件直接实现以提高指令的执行速度。

第三级是操作系统虚拟级。从操作系统的基本功能来看,一方面它要直接管理传统机器中的软、硬件资源,另一方面它又是传统机器的引申。它提供了传统机器所没有的某些基本操作和数据结构,如文件结构与文件管理的基本操作、存储体系和多道程序以及多重处理所用的某些操作、设备管理等。

第四级是汇编语言虚拟级。这一级的机器语言是汇编语言,用汇编语言编写的程序,首先翻译成第三级和第二级语言,然后再由相应的机器执行。完成汇编语言翻译的程序就叫做汇编程序。

第四级上出现了一个重要变化。通常的第一、二和三级是用解释(Interpretation)方法实现的,而第四级或更高级则经常是用翻译(Translation)方法实现。

翻译和解释是语言实现的两种基本技术。它们都是以执行一串  $N$  级指令来实现  $N+1$  级指令,但二者仍存在着差别:翻译技术是先把  $N+1$  级程序全部变换成  $N$  级程序后,再去执行新产生的  $N$  级程序,在执行过程中  $N+1$  级程序不再被访问。而解释技术是每当一条  $N+1$  级指令被译码后,就直接去执行一串等效的  $N$  级指令,然后再去取下一条  $N+1$  级的指令,依此重复进行。在这个过程中不产生翻译出来的程序,因此解释过程是边变换边执行的过程。在实现新的虚拟机器时,这两种技术都被广泛使用。一般来说,解释执行比翻译花的时间多,但存储空间占用较少。

第五级是高级语言虚拟级。这一级的机器语言就是各种高级语言,目前高级语言已达数百种。用这些语言所编写的程序一般是由称为编译程序的翻译程序翻译到第四级或第三级

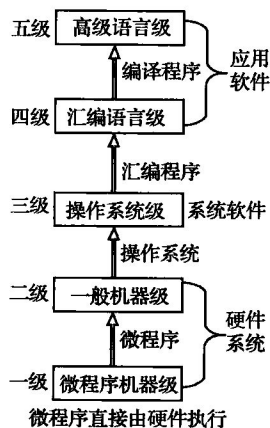


图 1.1 计算机层次结构