



教育部考试中心

National Computer Rank Examination  
全国计算机等级考试

# 三级教程

## ——数据库技术

(2012年版)



高等教育出版社  
HIGHER EDUCATION PRESS

# 全国计算机等级考试三级教程

## ——数据库技术

### (2012 年版)

Quanguo Jisuanji Dengji Kaoshi Sanji Jiaocheng  
——Shujuku Jishu

教育部考试中心



高等教育出版社·北京  
HIGHER EDUCATION PRESS BEIJING

## 内容提要

由教育部考试中心组织和实施的计算机等级考试是一种客观、公正、科学的专门测试计算机应用人员的计算机知识与技能的全国范围的等级考试，它面向社会，服务于社会。

本书由教育部考试中心组织，在全国计算机等级考试委员会指导下由有关专家执笔编写而成。本书按照《全国计算机等级考试大纲》中对三级数据库技术的要求而编写，内容包括：计算机基础知识、数据结构与算法、操作系统、数据库技术基础、关系数据库系统、关系数据库标准语言SQL、关系数据库的规范化理论与数据库设计、数据库系统实现技术、主流关系数据库管理系统、新一代数据库系统工具以及数据库技术的发展。

本书除了可以作为计算机等级考试教材外，还可作为学习计算机知识的参考书。

## 图书在版编目 (CIP) 数据

全国计算机等级考试三级教程：2012年版. 数据库技术/教育部考试中心编. --北京：高等教育出版社，2011.11 (2012.8重印)

ISBN 978-7-04-033900-0

I. ①全… II. ①教… III. ①电子计算机-水平考试-教材②数据库系统-水平考试-教材 IV. ①TP3

中国版本图书馆 CIP 数据核字 (2011) 第 238629 号

策划编辑 何新权 责任编辑 何新权 封面设计 陈方 责任校对 胡晓琪

---

出版发行	高等教育出版社	网 址	<a href="http://www.hep.edu.cn">http://www.hep.edu.cn</a>
社 址	北京市西城区德外大街4号		<a href="http://www.hep.com.cn">http://www.hep.com.cn</a>
邮政编码	100120	网上订购	<a href="http://www.landraco.com">http://www.landraco.com</a>
印 刷	保定市中国美凯印刷有限公司		<a href="http://www.landraco.com.cn">http://www.landraco.com.cn</a>
开 本	787mm×1092mm 1/16		
印 张	22.75		
字 数	570千字	版 次	2011年11月第1版
购书热线	010-58581118	印 次	2012年8月第2次印刷
咨询电话	400-810-0598	定 价	45.00元

---

本书如有缺页、倒页、脱页等质量问题，请到所购图书销售部门联系调换

版权所有 侵权必究

物料号 33900-00

# 大力推行全国计算机等级考试 为发展知识经济、信息产业和培养计算机 专门人才作出贡献

(代序)

中国科学院院士 北京大学信息与工程科学学部主任  
全国计算机等级考试委员会主任委员

杨芙清

当今，人类正在步入一个以智力资源的占有和配置，知识生产、分配和使用为最重要因素的知识经济时代，也就是小平同志提出的“科学技术是第一生产力”的时代。科教是经济发展的基础，知识是人类创新的源泉。基础研究的科学发现、应用研究的原理探索和开发研究的技术发明，三者之间的联系愈来愈紧密，转换周期日趋缩短。世界各国的竞争已成为以经济为基础、以科技（特别是高科技）为先导的综合国力的竞争。

在高科技中，信息科学技术是知识高度密集、学科高度综合，具有科学与技术融合特征的学科。它直接渗透到经济、文化和社会的各个领域，迅速改变着人们的观念、生活和社会的结构，是当代发展知识经济的支柱之一。

在信息科学技术中，微电子是基础，计算机硬件及通信设施是载体，计算机软件是核心。软件是人类知识的固化，是知识经济的基本表征，软件已成为信息时代的新型“物理设施”。人类抽象的经验、知识正逐步由软件予以精确地体现。在信息时代，软件是信息化的核心，国民经济和国防建设、社会发展、人民生活都离不开软件，软件无处不在。软件产业是增长最快的朝阳产业，是具有高额附加值、高投入/高产出、无污染、低能耗的绿色产业。软件产业的发展将推动知识经济的进程，促进从注重量的增长向注重质的提高的方向发展，是典型的知识型产业。软件产业是关系到国家经济安全和文化安全，体现国家综合实力，决定21世纪国际竞争地位的战略产业。

为了适应知识经济发展的需要，大力推动信息产业的发展，需要在全民中普及计算机的基本知识，广开渠道，培养和造就一批又一批能熟练运用计算机和软件技术的各行各业的专门人才。

1994年，原国家教委（现教育部）推出了全国计算机等级考试，它是一种重视应试人员对计算机和软件的实际掌握能力的考试。它不限制报考人员的学历背景，任何年龄段的人员都可以报考。这就为培养各行各业计算机的应用人才开辟了一条广阔的道路。

1994年是推出计算机等级考试的第一年，当年参加考试的有1万余人；而2010年，年报考人数已近507万人。截至2011年上半年，全国计算机等级考试共开考33次，考生人数累计

达 4 130 万，其中有 1 550 万人获得了不同级别的计算机等级证书。

事实说明，鼓励社会各阶层的人士通过各种途径掌握计算机应用技术，并运用等级考试对他们的才干予以认真的、有权威性的认证，是一种人才培养的有效途径，是比较符合我国具体情况的。等级考试也为用人单位录用和考核人员提供了一种测评手段。从有关公司对等级考试所做的社会抽样调查结果看，不论是管理人员还是应试人员，对该项考试的内容和形式都给予了充分的肯定。

计算机等级考试所取得的良好效果，也同全国各有关单位专家们在等级考试的大纲编写、试题设计、阅卷评分及效果分析等多项工作中所付出的大量心血和辛勤劳动密切相关，他们为这项工作的顺利开展作出了重要的贡献。

计算机与软件技术是一项日新月异的高新技术。计算机等级考试大纲有必要根据计算机与软件技术在近年的新发展，进行适当的修正，从而使等级考试更能反映当前计算机与软件技术的应用实际，使培养计算机应用人才的基础工作更健康地向前发展。

从面临知识经济的机遇与挑战这样一个社会大环境的背景出发，考察全国计算机等级考试，就会看到，这一举措是符合知识经济和信息产业的发展方向的，是值得大力推行的。

我们相信，在 21 世纪知识经济和信息产业加快发展的形势下，在教育部考试中心的精心组织领导下，在全国各有关专家们的大力配合下，全国计算机等级考试一定会以更新的面貌出现，从而为我国培养计算机应用专门人才的宏大事业作出更多的贡献。

2011 年 10 月

# 前 言

本书是根据教育部考试中心组织和实施的《全国计算机等级考试三级数据库技术考试大纲》编写的。

三级数据库技术教程的内容包括计算机基础知识、数据结构、操作系统、数据库系统的基本概念和基本原理、数据库设计和数据库应用系统开发的方法与工具以及关于数据库技术发展的简单介绍。要求三级数据库技术的合格考生应具备计算机的基础知识，了解和掌握数据结构、操作系统的基本原理和方法、数据库系统的基本原理和方法，熟悉数据库应用系统的开发方法和工具的使用，从而具备从事数据库应用系统项目开发和维护的基本能力。

本书在《全国计算机等级考试三级数据库技术教程（2011年版）》的基础上修订而成，对2011年版的内容进行了若干调整。修订的原则是，重视对计算机经典理论和基础概念的介绍，并注重反映计算机科学技术的新发展和新应用，以适应技术发展和应用需求变化对从事数据库应用系统开发和维护的人员所提出的新的要求。

本书由北京大学杨冬青教授主编，参加编写的人员有：吴功宜（第1章），陈向群（第1、3章），杨冬青（第2、7、8章），邵佩英（第4、5、6、9章），王文杰（第10、11章）。

由于编写时间仓促，教材涉及面较广，疏漏之处必然存在，望读者提出宝贵意见。

作者

# 目 录

<b>第 1 章 计算机基础知识</b> .....	1	<b>第 2 章 数据结构与算法</b> .....	49
1.1 计算机系统组成 .....	1	2.1 基本概念 .....	49
1.1.1 计算机的发展阶段 .....	1	2.1.1 数据结构的基本概念 .....	49
1.1.2 “存储程序”工作原理 .....	2	2.1.2 主要的数据存储方式 .....	50
1.1.3 计算机指令系统 .....	2	2.1.3 算法的设计与分析 .....	51
1.1.4 计算机的硬件组成 .....	4	2.2 线性表 .....	52
1.1.5 微处理器 .....	5	2.2.1 顺序表和一维数组 .....	53
1.1.6 总线 .....	5	2.2.2 链表 .....	54
1.1.7 计算机的软件组成 .....	6	2.2.3 栈 .....	56
1.1.8 计算机的配置和技术指标 .....	8	2.2.4 队列 .....	58
1.1.9 计算机的应用领域 .....	11	2.2.5 串 .....	60
1.2 计算机中的信息表示 .....	12	2.3 多维数组、稀疏矩阵和广义表 .....	60
1.2.1 计算机中的信息单位 .....	12	2.3.1 多维数组的顺序存储 .....	60
1.2.2 基本二进制运算 .....	13	2.3.2 稀疏矩阵的存储 .....	61
1.2.3 数据在计算机中的表示方式 .....	14	2.3.3 广义表的定义和存储 .....	63
1.2.4 计算机中的非数值信息 .....	15	2.4 树形结构 .....	64
1.3 计算机网络基础 .....	17	2.4.1 树的定义 .....	64
1.3.1 网络的基本概念 .....	17	2.4.2 二叉树的定义 .....	65
1.3.2 网络的分类 .....	18	2.4.3 树的二叉树表示 .....	66
1.3.3 Internet 基础 .....	21	2.4.4 二叉树和树的周游 .....	66
1.3.4 Internet 提供的服务 .....	25	2.4.5 二叉树的存储和线索二叉树 .....	67
1.3.5 Internet 的基本接入方式 .....	28	2.4.6 二叉树周游算法 .....	69
1.4 信息安全基础 .....	31	2.4.7 霍夫曼算法及其应用 .....	70
1.4.1 密码技术 .....	32	2.5 查找 .....	72
1.4.2 信息认证 .....	33	2.5.1 线性表查找 .....	73
1.4.3 恶意软件 .....	35	2.5.2 树形结构与查找 .....	77
1.4.4 计算机病毒 .....	37	2.6 排序 .....	84
1.4.5 防火墙技术 .....	39	2.6.1 插入排序 .....	84
1.4.6 网络安全 .....	40	2.6.2 选择排序 .....	86
1.4.7 操作系统安全 .....	44	2.6.3 交换排序 .....	90
1.4.8 数据库安全 .....	47	2.6.4 归并排序 .....	92
习题 1 .....	48	习题 2 .....	93

<b>第 3 章 操作系统</b> .....	95	3.6.6 磁盘调度 .....	152
3.1 操作系统概述 .....	95	习题 3 .....	154
3.1.1 操作系统的基本概念 .....	95	<b>第 4 章 数据库技术基础</b> .....	156
3.1.2 操作系统的功能 .....	96	4.1 数据库技术基本概念 .....	156
3.1.3 操作系统的分类 .....	97	4.1.1 信息、数据与数据处理 .....	156
3.1.4 操作系统与用户的接口 .....	99	4.1.2 数据管理技术的发展和数据库方法的特征 .....	157
3.1.5 操作系统的结构 .....	100	4.1.3 数据库、数据字典、数据库管理系统、数据库系统 .....	158
3.2 操作系统的硬件环境 .....	101	4.1.4 数据库技术的研究领域 .....	160
3.2.1 中央处理器 (CPU) .....	101	4.2 数据模型 .....	161
3.2.2 存储体系 .....	102	4.2.1 数据模型的概念 .....	161
3.2.3 中断系统 .....	102	4.2.2 数据模型的要素 .....	161
3.2.4 I/O 控制方式 .....	104	4.2.3 数据模型的分类 .....	162
3.2.5 时钟 .....	104	4.2.4 概念数据模型——ER 模型 .....	163
3.3 进程管理 .....	105	4.2.5 常用的逻辑数据模型 .....	166
3.3.1 进程的基本概念 .....	105	4.3 数据库系统的模式结构 .....	169
3.3.2 进程控制 .....	107	4.3.1 数据库系统中的模式、实例和数据库状态 .....	169
3.3.3 线程的基本概念 .....	108	4.3.2 数据库系统的三级模式结构 .....	172
3.3.4 进程 (线程) 调度 .....	109	4.3.3 数据库的二层映象与数据独立性 .....	173
3.3.5 进程同步机制 .....	111	习题 4 .....	174
3.3.6 进程间通信 .....	115	<b>第 5 章 关系数据库系统</b> .....	175
3.3.7 死锁 .....	117	5.1 关系数据库系统和关系数据模型 .....	175
3.4 存储管理 .....	121	5.1.1 关系数据库系统的发展历史 .....	175
3.4.1 存储管理概述 .....	121	5.1.2 关系数据模型 .....	176
3.4.2 可变分区存储管理方案 .....	123	5.2 关系模型的数据结构 .....	177
3.4.3 页式存储管理方案 .....	126	5.2.1 关系模型的数据结构和基本术语 .....	177
3.4.4 虚拟页式存储管理方案 .....	129	5.2.2 关系的形式定义和关系数据库对关系的限定 .....	179
3.4.5 虚拟存储管理的性能问题 .....	132	5.2.3 关系数据库中常用的表示法 .....	181
3.5 文件管理 .....	133	5.3 关系模型的完整性约束 .....	182
3.5.1 文件管理概述 .....	133	5.3.1 关系模型的完整性约束分类 .....	182
3.5.2 文件结构和存取方式 .....	134	5.3.2 域完整性约束 .....	182
3.5.3 文件目录及实现 .....	137	5.3.3 实体完整性约束 .....	183
3.5.4 文件的操作及实现 .....	139	5.3.4 参照完整性约束 .....	183
3.5.5 文件系统的实现 .....	140	5.3.5 关系模型数据完整性约束的检查 .....	184
3.5.6 文件系统的安全 .....	143		
3.5.7 文件系统的性能 .....	145		
3.6 设备管理 .....	146		
3.6.1 设备管理概述 .....	146		
3.6.2 通道技术 .....	147		
3.6.3 缓冲技术 .....	149		
3.6.4 虚拟设备技术 (SPOOLing 技术) .....	149		
3.6.5 设备分配及处理 .....	150		



5.4 关系代数 .....	185	<b>第7章 关系数据库的规范化理论</b>	<b>与数据库设计</b> .....	225
5.4.1 关系代数操作的分类 .....	185	7.1 “不好”的关系模式中存在的	问题 .....	225
5.4.2 基于传统集合论的关系代数		7.2 函数依赖 .....		226
操作 .....	186	7.2.1 函数依赖的定义 .....		226
5.4.3 一元的专门关系操作 .....	188	7.2.2 函数依赖的逻辑蕴含 .....		227
5.4.4 二元的专门关系操作 .....	190	7.2.3 码 .....		227
5.4.5 扩展的关系操作 .....	193	7.2.4 函数依赖的公理系统 .....		227
习题5 .....	195	7.3 1NF, 2NF, 3NF, BCNF .....		228
<b>第6章 关系数据库标准语言</b>		7.3.1 第一范式(1NF)及进一步	规范化 .....	228
<b>SQL</b> .....	196	7.3.2 第二范式(2NF) .....		229
6.1 SQL概述 .....	196	7.3.3 第三范式(3NF) .....		229
6.1.1 SQL的标准化历程 .....	196	7.3.4 Boyce-Codd范式(BCNF) .....		230
6.1.2 SQL的功能特点 .....	196	7.4 多值依赖和4NF .....		230
6.1.3 SQL的基本组成 .....	197	7.4.1 多值依赖 .....		230
6.1.4 SQL的数据类型 .....	199	7.4.2 第四范式(4NF) .....		232
6.2 SQL的数据定义 .....	200	7.5 关系模式的分解 .....		232
6.2.1 SQL的模式 .....	200	7.5.1 模式分解的等价标准 .....		233
6.2.2 SQL的基本表 .....	201	7.5.2 关于模式分解的几个事实 .....		234
6.2.3 SQL的索引 .....	203	7.6 数据库设计 .....		235
6.2.4 SQL的域 .....	204	7.6.1 设计过程概览 .....		235
6.3 SQL的数据查询 .....	204	7.6.2 概念结构设计 .....		235
6.3.1 简单查询 .....	205	7.6.3 逻辑结构设计 .....		237
6.3.2 连接查询 .....	207	7.6.4 物理结构设计 .....		238
6.3.3 嵌套查询 .....	208	7.7 规范化理论在数据库设计中	的应用 .....	239
6.3.4 集合查询 .....	209	习题7 .....		239
6.3.5 SQL中的连接表和外连接 .....	209	<b>第8章 数据库系统实现技术</b> .....		241
6.4 SQL的数据修改 .....	211	8.1 数据库管理系统概述 .....		241
6.4.1 SQL的插入操作 .....	211	8.1.1 数据库管理系统的基本功能 .....		241
6.4.2 SQL的删除操作 .....	211	8.1.2 数据库管理系统的主要成分和	工作流程 .....	242
6.4.3 SQL的更新操作 .....	212	8.2 存储管理 .....		243
6.5 SQL的视图 .....	212	8.2.1 物理存储介质简介 .....		243
6.5.1 视图的概念 .....	212	8.2.2 数据存储组织 .....		244
6.5.2 视图的查询 .....	214	8.2.3 缓冲区管理 .....		245
6.5.3 视图的修改 .....	215	8.2.4 数据字典 .....		245
6.5.4 视图的作用 .....	216	8.2.5 索引结构 .....		246
6.6 SQL的数据控制 .....	217	8.3 查询处理 .....		247
6.6.1 授予权限 .....	217			
6.6.2 收回权限 .....	219			
6.7 嵌入式SQL和动态SQL .....	219			
6.7.1 嵌入式SQL .....	219			
6.7.2 动态SQL .....	222			
习题6 .....	223			

8.3.1 查询处理概述 .....	247	和发展趋势 .....	285
8.3.2 查询执行 .....	248	10.2 系统开发工具的选择 .....	287
8.3.3 查询优化 .....	249	10.2.1 当前应用开发对工具的总需求 .....	287
8.4 事务管理 .....	251	10.2.2 目前应用开发工具存在的问题 .....	288
8.4.1 事务的概念和特性 .....	251	10.3 基于浏览器/服务器结构的软件开发 .....	288
8.4.2 故障恢复 .....	253	10.3.1 信息系统的层次结构划分 .....	288
8.4.3 并发控制 .....	255	10.3.2 浏览器/服务器系统开发工具 .....	289
习题 8 .....	262	10.4 数据库建模工具——PowerDesigner .....	292
<b>第 9 章 主流关系数据库管理系统</b> .....	263	10.4.1 PowerDesigner 功能介绍 .....	293
9.1 关系数据库管理系统综述 .....	263	10.4.2 PowerDesigner 主要模块 .....	294
9.1.1 关系 DBMS 的发展历史 .....	263	10.4.3 利用 PowerDesigner 进行数据库设计 .....	295
9.1.2 新的应用需求对关系 DBMS 的挑战 .....	264	10.5 可视化程序开发工具 Delphi .....	297
9.1.3 关系 DBMS 的选择 .....	265	10.5.1 Delphi 的主要特点 .....	297
9.1.4 关系 DBMS 的发展趋势 .....	266	10.5.2 Delphi 可视化编程环境 .....	298
9.2 SQL Server .....	267	10.5.3 Delphi 对数据库应用的开发 .....	298
9.2.1 SQL Server 概述 .....	267	10.6 应用开发工具 PowerBuilder .....	300
9.2.2 SQL Server 2000 的体系结构 .....	267	10.6.1 PowerBuilder 的主要特点 .....	300
9.2.3 SQL Server 2000 的功能特点 .....	268	10.6.2 PowerBuilder 的数据窗口 .....	301
9.2.4 SQL Server 2000 多版本支持及操作系统的选择 .....	269	10.7 Windows 平台应用程序开发工具 Visual Studio .....	302
9.2.5 SQL Server 2000 数据库 .....	270	10.7.1 Visual Studio 的历史 .....	302
9.2.6 从 Internet 访问 SQL Server 2000 数据库 .....	271	10.7.2 Visual Studio 的主要特点 .....	303
9.3 Oracle .....	272	10.7.3 Visual Studio 2008 的新功能 .....	303
9.3.1 Oracle 系统概述 .....	272	习题 10 .....	306
9.3.2 Oracle 体系结构 .....	272	<b>第 11 章 数据库技术的发展</b> .....	308
9.3.3 Oracle 数据库服务器功能及其特色 .....	274	11.1 数据库技术发展阶段 .....	308
9.3.4 Oracle 的工具及其功能 .....	275	11.1.1 第一代数据库系统 .....	308
9.3.5 Oracle 的数据仓库和 Internet 解决方案 .....	276	11.1.2 第二代数据库系统 .....	309
9.3.6 Oracle 的对象—关系特性 .....	277	11.1.3 第三代数据库系统 .....	309
9.4 SQL Server 和 Oracle 的安全性 .....	279	11.2 数据库系统体系结构 .....	310
习题 9 .....	282	11.2.1 集中式数据库系统体系结构 .....	310
<b>第 10 章 新一代数据库系统工具</b> .....	284	11.2.2 分布式数据库系统体系结构 .....	311
10.1 新一代数据库系统工具概述 .....	284	11.2.3 客户机/服务器数据库系统体系结构 .....	313
10.1.1 数据库系统工具的分类 .....	284	11.2.4 面向 Web 应用的数据库系统体系结构 .....	314
10.1.2 新一代数据库系统工具的特征			

11.3 面向对象数据库系统 .....	315	11.7.3 数据仓库与数据挖掘的结合 .....	335
11.3.1 面向对象的基本概念 .....	315	11.8 Web 挖掘 .....	336
11.3.2 面向对象数据库管理系统 .....	317	11.8.1 Web 信息的特点 .....	336
11.3.3 面向对象数据库系统模型和 其他模型的简单比较 .....	322	11.8.2 Web 挖掘的流程和分类 .....	337
11.4 移动数据库系统 .....	322	习题 11 .....	338
11.4.1 移动环境的特征 .....	322	<b>附录 1 全国计算机等级考试三级数 数据库技术考试大纲 (2007 年版) .....</b>	<b>339</b>
11.4.2 移动数据库的数据管理 .....	323	<b>附录 2 2011 年 3 月全国计算机等级 考试三级笔试试题及参考答案 ——数据库技术 .....</b>	<b>341</b>
11.5 多媒体数据库系统 .....	324	<b>附录 3 习题答案 .....</b>	<b>349</b>
11.5.1 多媒体数据库的基本概念 .....	324	<b>参考文献 .....</b>	<b>351</b>
11.5.2 多媒体数据库的数据管理 .....	324		
11.5.3 多媒体数据库系统的重要 问题 .....	325		
11.6 数据仓库和联机分析处理 .....	326		
11.6.1 数据仓库的基本概念 .....	326		
11.6.2 数据仓库的数据模型 .....	328		
11.6.3 数据仓库的体系结构 .....	330		
11.6.4 联机分析处理的功能 .....	331		
11.7 数据挖掘 .....	333		
11.7.1 知识发现与数据挖掘 .....	333		
11.7.2 数据挖掘的目标和任务 .....	334		

# 第 1 章

---

## 计算机基础知识

### 1.1 计算机系统组成

一般而言，计算机系统由两个基本部分组成，这两个部分是硬件系统和软件系统，它们构成一个完整的计算机系统。

计算机硬件是组成计算机的物理设备的总称，它们由各种器件和电子线路组成，是计算机完成计算工作的物质基础。

计算机软件是在计算机硬件设备上运行的各种程序及其相关资料的总称。程序则是由计算机最基本的操作指令组成的。计算机所有指令的组合称为机器的指令系统。没有软件的计算机系统通常称为“裸机”，而裸机是无法工作的。因此，如果将硬件比喻为“舞台”，是系统的物质基础，则软件可比喻为“剧目”，是系统的灵魂，二者缺一不可，即硬件和软件的相互依存才能构成一个可用的计算机系统。

#### 1.1.1 计算机的发展阶段

计算机是一种专门进行计算的工具，属于人类发明的工具中的一种。不过，计算机与其他人类发明的工具有点不相同，其他多数工具是人类四肢和五官的延伸，而计算机则可以看做是人的头脑的延伸，能帮助完成一些原本需要耗费人的脑力才能完成的工作。

在第二次世界大战期间，由于国防军事工程和武器设计的迫切需要，对计算提出了巨大需求。英国数学家阿兰·图灵提出了一种自动计算机器的模型，即图灵机；冯·诺依曼提出了现代计算机的基本原理：存储程序控制原理。相关的计算理论和计算机构想逐渐成熟，导致了世界上第一台电子数字计算机问世。

计算机的发展经历了四个发展阶段。

##### 1. 第一代：电子管计算机（1946—1957年）

这一代计算机的运算速度约为每秒几千次至几万次，体积大，成本高，可靠性低。在此期间，开始形成计算机的基本体系，确定了程序设计的基本方法，数据处理机开始得到应用。这一代计算机的支撑软件是机器语言和汇编语言。

##### 2. 第二代：晶体管计算机（1958—1964年）

这一代计算机的运算速度提高到每秒几万次至几十万次，可靠性提高，体积缩小，成本降低。在此期间，工业控制机开始得到应用。这一代计算机的支撑软件是算法语言、管理程序，操作系统的雏形开始出现。

##### 3. 第三代：集成电路计算机（1965—1970年）

这一代计算机的运算速度是每秒几十万到几百万次，可靠性进一步提高，体积进一步缩小，成本进一步下降。在此期间，机种逐渐多样化、生产逐渐系列化、使用逐渐系统化，小型

计算机开始出现，操作系统基本成熟。

#### 4. 第四代：大规模集成电路计算机（1971年以后）

这一代计算机的运算速度提高到每秒几百万次、几千万次至每秒几千亿次甚至更高，可靠性更进一步提高，体积更进一步缩小，成本更进一步降低。

### 1.1.2 “存储程序”工作原理

1945年6月，冯·诺依曼（John von Neumann）等人发表了一篇长达101页纸的报告，即计算机史上著名的“101页报告”。该报告明确提出了计算机应该具有五大部件，程序和由该程序处理的数据应存放在同样的存储器里，并建议用二进制替代十进制运算。这个报告的革命意义在于“存储程序”，即计算机自动依次执行指令。冯·诺依曼的这一设计思想被誉为计算机发展史上的里程碑，标志着计算机时代的真正开始。

冯·诺依曼奠定了现代计算机工作原理的基础，其特点是：

(1) 在执行程序和处理数据时必须将程序和数据装入存储器中，然后才能使计算机在工作时能够自动地从存储器中取出指令并加以执行。

(2) 用二进制形式表示数据和指令。

(3) 对计算进行集中的顺序控制。

(4) 计算机系统由运算器、存储器、控制器、输入设备、输出设备等五大部件组成。

冯·诺依曼“存储程序”工作原理的核心包含两层含义：首先，将编写好的程序和原始数据存储在计算机的存储器中，即“存储程序”；其次，计算机按照存储的程序逐条取出指令加以分析，并执行指令所规定的操作，即“程序控制”。

到目前为止，世界上所有投入使用的电子数字计算机都采用“存储程序”工作原理，人们后来把这种采用“存储程序”工作原理的机器统称为“诺依曼机”。

### 1.1.3 计算机指令系统

计算机所能执行的操作称为指令。一台计算机所能执行的各种不同类型指令的总和，称为该计算机的指令系统。

计算机指令系统是表征一台计算机性能的关键要素之一。计算机指令系统的格式与功能不仅直接与计算机硬件结构有关，而且也直接影响到系统软件和应用软件的设计和使用。

#### 1. 计算机指令系统的发展

在20世纪50年代，计算机指令系统比较简单，通常只有定点加减、逻辑运算、数据传送、转移等十几至几十条指令。

到了20世纪60年代，随着计算机硬件和其他软件技术的发展，计算机的指令系统也逐渐复杂起来，出现了乘除运算、浮点运算、十进制运算、字符串处理等指令。有的计算机指令数目多达一二百条，计算机指令的寻址方式出现多样化的趋势。

在20世纪60年代后期，开始出现系列计算机、复杂指令系统计算机以及精简指令系统计算机等。

#### (1) 系列计算机

所谓系列计算机是指在一个系列的计算机中，各个型号计算机的基本指令系统相同，而且新推出的计算机型号的指令系统一定包含该系列原有机种的全部指令，实现“向下兼容”。这样，原有旧机型上运行的各种软件可以不加任何修改便在新机型上运行，大大减少了软件开发费用。

## (2) 复杂指令系统计算机

随着计算机技术的发展，有的计算机指令系统中的指令多达几百条，人们把这种计算机称为复杂指令系统计算机（Complex Instruction Set Computer, CISC）。

如此庞大的指令系统是难以保证正确性的，而且也不易调试维护，有时还造成硬件资源的浪费。

## (3) 精简指令系统计算机

精简指令系统计算机（Reduced Instruction Set Computer, RISC）是一种计算机系统结构的设计思想，其目的是要改变复杂指令系统计算机指令过多的状态，希望用较少的指令实现类似的功能，从而达到简化硬件结构和程序设计的目标。

显然，指令的种类少了，那么有些原先需要一条指令完成的操作，就可能需要多条指令完成。所以，究竟是复杂指令系统优越，还是精简指令系统优越，目前尚没有定论。

## 2. 指令类型

不同计算机的指令系统包含的指令种类和数目是不同的。一般的指令系统均包含三类指令：数据传送类指令、算术逻辑类指令和判定控制类指令等。

### (1) 数据传送类指令

这类指令主要用于在计算机的各种存储装置之间传送数据。比如将内存存储单元的数据装入 CPU 的寄存器中有加载（load）指令，而将 CPU 的某个寄存器中的数据保存到一个内存存储单元中要用存储（store）指令。

### (2) 算术逻辑类指令

算术逻辑类指令完成在 CPU 的寄存器和内存存储单元之间的各种算术和逻辑类操作。比如可以完成在某个 CPU 的寄存器和另一个 CPU 的寄存器之间进行加法操作，并将结果保存在第三个寄存器中。

### (3) 判定控制类指令

判定控制类指令指挥程序完成一些非数据操作类型的执行活动。比如无条件跳转（jump）指令，可以强制程序直接转移到一个指定的地址上，去执行该地址中的指令。又比如分支（branch）指令，可以让程序依据出现的条件的不同，转移到不同的地址上。

## 3. 指令系统的寻址方式

指令中如何提供操作数或操作数地址的方式称为寻址方式。在不同的计算机系统中，寻址方式的名称和分类并不统一，这里介绍几种比较简单的寻址方式。

(1) 立即寻址（立即数寻址）：指令中直接给出操作数，操作数紧跟在操作码之后，作为指令的一部分存放在代码段里，在取出指令的同时也就取出了操作数，立即有操作数可用，所以称为立即寻址。

(2) 寄存器寻址：操作数在寄存器中，指令中指明寄存器代号，这种寻址方式称为寄存器寻址。

(3) 直接寻址：在直接寻址方式中，指令中直接给出操作数所在的地址。

(4) 寄存器间接寻址：在采用寄存器间接寻址方式时，指令中给出寄存器代号，被指定的寄存器中存放着操作数的地址。

(5) 寄存器相对寻址：操作数的地址是一个基址寄存器的内容与指令中指定的 8 位或 16 位位移量之和。

### 1.1.4 计算机的硬件组成

以“存储程序”原理为基础的冯·诺依曼结构的计算机，一般都由五大功能部件组成，它们是：运算器、控制器、存储器、输入设备和输出设备。图 1.1 给出计算机各功能部件的关系图，图中粗箭头线代表数据传输线路，单箭头线代表控制信号传输线路。

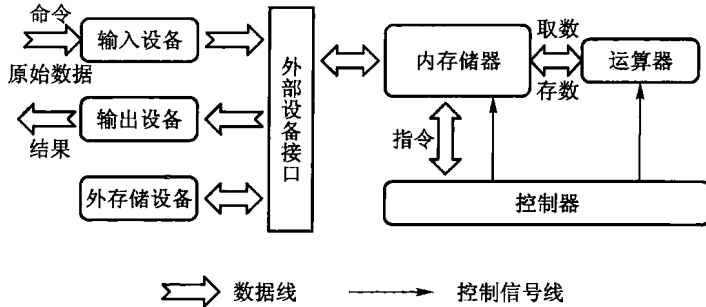


图 1.1 计算机硬件系统基本组成框图

#### (1) 运算器

运算器是用于对数据进行加工的部件，它可对数据进行算术运算和逻辑运算。

算术运算包括加、减、乘、除及它们的复合运算。逻辑运算包括一般的逻辑判断和逻辑比较，如比较、移位、逻辑加、逻辑乘、逻辑反等操作。

#### (2) 控制器

控制器是计算机的控制部件，它控制计算机各部分自动协调地工作，完成对指令的解释和执行。它每次从存储器读取一条指令，经分析译码，产生一串操作命令发向各个部件，控制各部件动作，实现该指令的功能；然后再取下一条指令，继续分析、执行，直至程序结束，从而使整个机器能连续、有序地进行工作。

#### (3) 存储器

存储器是计算机的记忆装置，它的主要功能是存放程序和数据。程序是计算机操作的依据，数据是计算机操作的对象。存储器可以划分为内存储器和外存储器。现代内存储器由半导体高速存储部件构成，外存储器主要由硬盘、磁带等大容量存储部件构成。

#### (4) 输入设备

输入设备是外部向计算机传送信息的装置，其功能是将数据、程序及其他信息从人们熟悉的形式转换成计算机能接受的信息形式，输入到计算机内部。

常见的输入设备有键盘、鼠标、光笔、纸带输入机、模/数转换器、声音识别输入设备等。

#### (5) 输出设备

输出设备的功能是将计算机内部二进制形式的信息转换成人们所需要的或其他设备能接受和识别的信息形式。

常见的输出设备有打印机、显示器、绘图仪、数/模转换器、声音合成输出设备等。

有的设备兼有输入、输出两种功能，如磁盘机、磁带机等，它们既是输入设备，也是输出设备。

通常将运算器和控制器合称为中央处理器（Central Processor Unit, CPU）；中央处理器和内存储器合称为主机。中央处理器能够直接访问的存储设备只有内存储器。输入设备、输出设备和外存储器合称为外部设备，又称为 I/O 设备。外部设备通过 I/O 总线与主机相连。

## 1.1.5 微处理器

### 1. 什么是微处理器

在早期的计算机系统中，运算器和控制器都是由体积巨大的电子真空管部件组成的。随着半导体技术的发展，中央处理器的体积逐渐缩小，集成在一块单独的半导体器件上，被称为微处理器。可见，微处理器就是具有中央处理器功能的大规模集成电路器件。

现在，微处理器已经无处不在，在各种计算机系统中，从巨型计算机、大型计算机、服务器到台式计算机、笔记本计算机和嵌入式计算机，都有大量的高性能微处理器。在工业控制和国防军事领域中也离不开各类的微处理器，甚至在电视机、洗衣机、移动电话等家电产品中也需微处理器。微处理器已经成为各种数字化智能设备的关键部件。

### 2. 微处理器的分类

依据微处理器的应用领域，可以将微处理器大致分为三类：通用微处理器、嵌入式微处理器（微控制器）和数字信号处理器等。

一般而言，通用微处理器追求高性能，它们用于运行通用软件，配备完备、复杂的操作系统。

嵌入式微处理器的功能相对简单，重视成本，主要用于各种嵌入式系统中，如工业控制、移动电话、彩色电视机等众多数字智能设备领域中，一些简单的嵌入式微处理器称为微控制器。

数字信号处理器则不仅具有一般微处理器的功能，还在内部设计有能够高速处理多路数字信号的电路，主要用在需要快速处理大量复杂数字信号的领域，如通信设备、雷达、数字图像处理设备、数字音视频设备中。

### 3. 多核微处理器

最初的 Intel 首款微处理器 4004 的主频为 108 kHz，后来的奔腾 4 处理器的主频已经达到了 1.5 GHz。而在 2000 年，Intel 公司开启了多核时代的大幕。

所谓多核，就是在一块单一的集成电路芯片上同时集成两个甚至两个以上的微处理器。依据不同的设计理念，可以设计出结构不同的多核微处理器。

向多核结构的转变是微处理器的一次重大革新。以往微处理器的体系结构改变几乎没有对软件的体系结构造成什么大的影响。然而在多核微处理器中，各 CPU 会并行执行指令，软件设计人员如果不考虑多核微处理器的并行处理功能编写程序，就不能充分发挥多核微处理器的运算性能。

另外，与单核微处理器软件相比，多核微处理器软件的程序结构与调试都变得极为复杂。这是对软件设计人员的一个挑战。

## 1.1.6 总线

### 1. 总线的概念

在计算机内，CPU、内存、I/O 设备等部件之间必须相互连接，以保证有关的各类信号和数据能正确及时地传送。连接一般有两种方式，一种是专用电路连接，这在早期计算机系统中是很平常的处理方式；另一种则是构建公用的信号或数据传输通道，这种在各个计算机部件之间可共享连接的信息传输通道称为总线。总线技术早在 20 世纪 70 年代的计算机系统中就已出现。在微型计算机出现之后，总线技术有了飞速的发展。

总线可以分为数据总线、地址总线、控制总线等。显然，总线不仅可以用在 I/O 设备上，



也可以用在 CPU 内部。事实上,在现代微处理器中设计有各类内部总线,不过这些 CPU 内部使用的总线不在我们的讨论范围之内。

采用总线方式组织计算机结构的主要优势是灵活性和低成本。由于总线是共享的,所以只要新的设备遵循总线的标准,就很容易增加新设备。而多种计算机设备可以共用一套总线,很明显具有成本上的优势。

总线在传统上分类为 CPU—内存总线和 I/O 总线。CPU—内存总线短,速度快,其传输带宽要与内存的存取速度相匹配,以便实现主存与 CPU 之间的最大数据传输速率。I/O 总线所需要连接的设备类型多,所以总线长,通常会遵循某个总线标准。

## 2. 常见的 I/O 总线类别

目前常见的 I/O 总线有如下几种。

**PCI (Peripheral Component Interconnect) 总线:**这是一种不依附于某个具体处理器的局部总线,能支持多种外设,并能在高时钟频率下保持高性能。

**USB (Universal Serial Bus) 通用串行总线:**这是一种连接 I/O 串行设备的技术标准。USB 定义了一类通用硬件接口,使得在计算机上添加任何 I/O 串行设备都变得相当规范并且操作非常容易。USB 具有即插即用的功能,允许外部设备的热插拔,降低了设备成本与使用过程中技术支持与服务的成本,还可以主动为外部设备提供电源。

**1394 总线:**起源于 Apple 公司,又名 FireWire,是为家用电器研制的一种高速串行总线标准,其目的是解决对速度要求很高的宽带设备的传输问题,所以,1394 总线在数字视频设备(比如数字便携式摄像机)中得到了普遍应用。

## 1.1.7 计算机的软件组成

计算机软件是计算机程序和进行数据处理所必需的各种手续、规则和文件的总称。计算机软件可以划分为系统软件和应用软件两大类。

### 1. 系统软件

系统软件包含程序设计语言处理程序、操作系统、数据库管理系统以及各种通用服务程序等。

#### (1) 计算机语言与语言处理程序

##### ① 计算机语言

计算机语言是一类面向计算机的人工语言,它是进行程序设计的工具,又称为程序设计语言。现有的程序设计语言一般可分为三类。

##### (a) 机器语言

机器语言是最初级且依赖于硬件的计算机语言。用机器语言编写程序,程序人员必须熟悉机器指令的二进制符号代码,记忆指令代码能完成的操作,还应指出这一操作对象的位置,即记忆指令的操作码和地址码。

用机器语言编写的程序称为机器语言程序,它全部(包括数据)都是二进制代码的形式,不易被人识别,但可以被计算机直接执行。由于机器语言直接依赖于机器,所以对于不同型号的计算机,其机器语言是不同的。即在一种类型计算机上编写的机器语言程序,不能在另一种不同的机器上运行。

由于机器语言程序是直接在计算机上执行,所以效率比较高,能充分发挥计算机的高速计算的能力。在计算机发展的初期,人们都是使用机器语言直接编写程序。但机器语言不易记忆