

初级程序员级 软硬件知识

第二版

刘 英 主 编



清华大学出版社

中国计算机软件专业技术资格和水平考试统编辅导教材

初级程序员级软硬件知识

(第二版)

刘 英 主编

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书为中国计算机软件专业技术资格和水平考试统编教材,根据 1996 年初级程序员级考试大纲编写。全书分 12 章,内容包括:计算机基础知识,计算机硬件和软件基本知识,操作系统基本知识(包括 DOS 和 Windows 的基本使用),数据结构,文字处理基本知识,WPS 文字处理系统,中文 Word,QBASIC 语言,数据库管理系统 FoxBASE,程序设计基本方法,安全管理与日常维护,计算机病毒和计算机英语等。各章末给出了习题,书后附录中给出了习题参考答案,供读者练习参考。本书不包括初级程序员考试大纲要求的初等数学及 C 语言,这些部分请考生参考有关教材。

本书内容丰富,针对性强,可供准备参加全国初级程序员级技术资格和水平考试的计算机人员学习参考,也可供有关部门培训使用。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

初级程序员级软硬件知识/刘英主编. —2 版. —北京:清华大学出版社,1996
中国计算机软件专业技术资格和水平考试统编辅导教材
ISBN 7-302-02181-3

I. 初… I. 刘… III. 电子计算机-知识-教材 N. TP301

中国版本图书馆 CIP 数据核字(96)第 07565 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:北京昌平环球印刷厂

发行者:新华书店总店北京科技发行所

开 本:787×1092 1/16 印张:22.75 字数:535 千字

版 次:1996 年 5 月第 2 版 1996 年 8 月第 3 次印刷

书 号:ISBN 7-302-02181-3/TP·1042

印 数:30001—40000

定 价:29.00 元

JS320/17

出版说明

当今国际间的竞争是综合国力的竞争,关键又是科学技术的竞争,说到底人才的竞争。而电子信息技术又是当今国际间竞争的热点,它渗透和影响现代化社会生活的各个方面,而科学技术(包括电子信息技术)是要靠人去掌握、去应用、去发展,国家的强盛、民族的振兴靠人才,人才的培养靠教育,所以,党中央要求我们把经济建设转到依靠科技进步和提高劳动者素质的轨道上来。

培养人才要坚持多种形式、多种途径,大力开展岗位培训,不断提高职工队伍的技术和专业水平,计算机软件专业技术资格和水平考试制度,就是为了加速我国电子信息技术的广泛应用和软件事业的发展,科学考核和合理使用人才,促进计算机软件人才的国际交流与合作,进一步深化职称改革。这种考试是于1985年首先在上海、云南、四川三省市实行的,1987年发展为部分省、区、市的联合考试,到1988年全国已有31个省、区、直辖市和计划单列市参加程序员、高级程序员两个级别的联合考试,1989年发展为程序员级、高级程序员级、系统分析员级三个级别的联合考试。1990年在全国统一组织实施了软件专业技术职务任职资格(水平)考试。1991年3月在计算机软件专业技术资格和水平考试工作会议上,对考试《暂行规定》作了修改,1991年的软件专业技术资格和水平考试就是依照新的《暂行规定》进行的。实践证明这是一种严格的认定考试制度,给应试者提供了一次均等的机会。软件专业技术资格和水平考试,每年举行一次,实行全国统一组织、统一大纲、统一试题、统一评分标准。

考试和培训是相辅相成的,为了使全国参加统一考试的考生得到较好的辅导教材,编一套全国统一的辅导教材是完全必要的。1991年9月考委会在北京主持召开了辅导教材编审委员会会议,对统编辅导教材的编写、出版作了部署,成立了统编辅导教材编审委员会,编辑出版了《中国计算机软件专业技术资格和水平考试统编辅导教材》一套六册。1995年11月考试中心又在北京主持召开由教材编审委员会主编、副主编及各本教材主编、主审等有关专家会议。与会专家一致认为,1991年编辑出版的《中国计算机软件专业技术资格和水平考试统编辅导教材》,是一套深受读者欢迎的好教材,对培训计算机软件人才起到了良好的作用。根据4年来的培训实践及计算机技术的新发展并根据修订后的《考试大纲》对原书的内容进行部分补充、删改,并将原来一套六册,改成一套五册。他们是《初级程序员级软硬件知识》(主编:刘英 主审:吴克忠 沈林兴)、《程序员级高级程序员级软件知识》(主编:施伯乐 主审:吴立德)、《程序员级高级程序员级硬件知识》(主编:王爱英 主审:曲庭维)、《程序员级高级程序员级程序设计》(主编:张福炎 主审:郑国梁)、《1994年度—1995年度试题分析与解答(初级程序员级、程序员级、高级程序员级、系统分析员级)》(主编:王春森 主审:张然)。我们的本意是要把这套全国统编辅导教材编成为具有正确性、科学性、系统性,而且针对性强,在一段时间内相对稳定的好教材。由于编者水平所限及科技的快速发展,错误在所难免,请有关专家和读者给予指正。

教材编审委员会

主 编：陈正清

副主编：华平澜 陈祥禄 宋小海 罗晓沛 施伯乐 吴立德

编 委：(按姓氏笔划排列)

方 裕 王勇领 王春森 王爱英 王 珊 吕文超

冯大本 刘 英 刘福滋 朱慧真 吴克忠 郑人杰

周明德 张 然 张公忠 张吉锋 张福炎 侯炳辉

钱士钧 徐国平 徐国定 徐洁磐 唐 敏

秘书长：邵祖英

秘书组：王 永 邓小敏 尹芳平 赵永红 吕跃军

秘书组联络地点：北京海淀区学院南路 55 号电子部计算机技术培训中心
(邮编:100081)

前 言

本书由中国计算机软件专业技术资格和水平考试统编教材编审委员会组织,根据1996年初级程序员级考试大纲的要求编写和审稿,供初级程序员级考生参考,亦可作为初级计算机技术人员培训教材。

按照初级程序员能够熟练掌握微型计算机操作,并能较熟练地使用常用软件,具有基本程序设计技能的要求,本书主要内容为:计算机基础知识,计算机硬件和软件基本知识,操作系统基本知识(包括DOS和Windows的基本使用),数据结构,文字处理基本知识,WPS文字处理系统,中文Word,QBASIC语言,数据库管理系统FoxBASE,程序设计基本方法,安全管理与日常维护,计算机病毒和计算机英语等。各章末给出了习题,书末附录中给出了习题参考答案,供读者练习参考。本书不包括初级程序员级考试大纲要求的初等数学及C语言程序设计,这些部分请考生参考有关教材。

本书由刘英主编,吴克忠、沈林兴主审。参加编写的有尚邦治(第1、2章)、沈林兴(第3章、第4.3节、第12.3节)、宋丹颖(第4.1、4.2节、第5章)、吴克忠(第6章)、刘英(第7、8、9、11章)、宋莉(第10章)、苏民生(第12.1、12.2节)。在本书编写过程中,电子工业部信息中心陈正清、王永同志、北京十二中庄燕文老师及清华大学出版社给予了很大帮助和支持,在此表示感谢。

由于时间紧迫,经验不足,疏漏和不足处请广大读者和专家给予指正。

编 者

1996年2月于北京

目 录

前言	1	习题	39
第1章 计算机基础知识	1	第3章 计算机软件基本知识	42
1.1 计算机系统组成	1	3.1 概述	42
1.2 计算机的运算基础	1	3.2 计算机软件系统的组成	43
1.2.1 数制及其转换	1	3.3 程序设计语言基本知识	45
1.2.2 算术运算和逻辑运算	4	习题	47
1.3 机内数据表示形式	8	第4章 操作系统基本知识和使用	50
1.3.1 原码、反码和补码	8	4.1 操作系统概述	50
1.3.2 定点数和浮点数	9	4.1.1 什么是操作系统	50
1.3.3 ASCII 码	10	4.1.2 操作系统的功能	50
1.3.4 奇偶校验	12	4.1.3 操作系统的类型	54
习题	12	4.2 DOS 操作系统的使用	56
第2章 计算机硬件基本知识	14	4.2.1 DOS 的基本结构	56
2.1 中央处理器	14	4.2.2 常用 DOS 命令的使用	63
2.1.1 运算器	14	4.2.3 DOS 6.x 新增命令和功能 介绍	73
2.1.2 控制器	15	4.3 Windows 简介	77
2.1.3 寄存器	15	4.3.1 概述	77
2.1.4 指令系统	17	4.3.2 安装和启动	80
2.1.5 指令的执行过程	19	4.3.3 中文 Windows 3.x 的基本组成和 基本操作	81
2.2 存储器系统	20	4.3.4 程序管理器	85
2.2.1 存储器的分类和作用	20	4.3.5 文件管理器	89
2.2.2 主存储器	21	习题	93
2.2.3 辅助存储器	22	第5章 数据结构	98
2.3 输入输出系统	25	5.1 基本概念	98
2.3.1 输入输出控制方式	25	5.2 线性表	98
2.3.2 系统总线	26	5.2.1 线性表的逻辑结构	99
2.3.3 I/O 接口卡	27	5.2.2 线性表的存储结构	99
2.3.4 输入输出设备	32	5.2.3 线性表的基本运算	100
2.4 网络	33	5.3 栈	101
2.4.1 网络概念和功能	33	5.3.1 栈的定义与结构	101
2.4.2 网络的组成和基本结构	34	5.3.2 栈的基本运算	102
2.4.3 网络协议	35		
2.4.4 网络硬件	36		
2.4.5 网络软件	39		

5.3.3 栈的应用	103	7.2.2 文件编辑	143
5.4 队列	104	7.2.3 文件模拟显示与打印输出	154
5.4.1 队列的定义与结构	104	7.3 WPS 文字处理系统视窗版介绍	157
5.4.2 队列的基本运算	104	7.3.1 特点	157
5.4.3 队列的应用	104	7.3.2 WPS 视窗版基本操作	158
5.5 数组	104	习题	166
5.5.1 数组定义和存储结构	105	第8章 中文 Word	168
5.5.2 数组的基本操作	106	8.1 中文 Word 简介	168
5.5.3 数组的应用	106	8.1.1 中文 Word 操作环境	168
5.6 记录	106	8.1.2 中文 Word 安装与启动	169
5.6.1 记录的定义和结构	107	8.2 Word 基本操作	169
5.6.2 记录的基本操作	107	8.2.1 窗口操作	169
5.6.3 记录的应用	107	8.2.2 命令与菜单的使用	170
5.7 字符串	108	8.2.3 文件编辑	173
5.7.1 串的定义和存储结构	108	8.2.4 简单表格使用	182
5.7.2 串的运算	108	习题	184
5.7.3 串的应用	109	第9章 QBASIC 语言	187
习题	109	9.1 概述	187
第6章 文字处理基本知识	112	9.1.1 QBASIC 语言的特点	187
6.1 汉字操作系统基本知识	112	9.1.2 启动 QBASIC	187
6.1.1 汉字的基本特性与汉字代码	112	9.1.3 QBASIC 的基本操作	188
6.1.2 汉字操作系统	116	9.1.4 QBASIC 语法概述	193
6.2 汉字输入方法	119	9.2 QBASIC 的程序控制结构	198
6.2.1 汉字流水码	119	9.2.1 条件转移与分支结构	198
6.2.2 汉字拼音编码	119	9.2.2 循环语句与循环结构	203
6.2.3 汉字拼形编码	120	9.2.3 数组	209
6.2.4 汉字音形码	121	9.2.4 子程序与自定义函数	211
6.2.5 五笔字型输入法	121	9.3 QBASIC 的文件系统	218
6.3 文字排版基本知识	124	9.3.1 程序文件	218
6.3.1 常用名词术语	124	9.3.2 数据文件	218
6.3.2 字号、字体与字型	127	9.4 在 QBASIC 中使用 GWBASIC	
6.3.3 文字版排版	128	程序	228
6.3.4 表格排版	129	习题	230
6.3.5 标点符号的排法	131	第10章 数据库管理系统	236
6.4 校对基本知识	131	10.1 数据库管理系统概述	236
习题	132	10.1.1 数据库的基本概念	236
第7章 WPS 文字处理系统	136	10.1.2 数据库管理系统中的数据	
7.1 WPS 系统简介	136	模型	238
7.2 WPS 基本操作	137	10.2 FoxBASE+ 系统	240
7.2.1 菜单使用	137		

10.2.1 FoxBASE+简介	240	11.2.2 排序	296
10.2.2 FoxBASE+的系统配置		11.2.3 查找	300
文件	241	11.2.4 常用的几个算法	304
10.3 FoxBASE+基本语法规则	242	11.2.5 字符串处理	307
10.3.1 FoxBASE+的命令	242	11.3 程序设计试题常见题型及解题	
10.3.2 常量、变量、表达式及函数	243	方法	308
10.4 FoxBASE+的基本操作	248	习题	313
10.4.1 数据库文件操作	248		
10.4.2 数据库记录操作	252	第12章 综合知识	318
10.4.3 多数据库操作	258	12.1 安全管理和日常维护	318
10.5 FoxBASE+程序设计	261	12.1.1 计算机安全和计算机犯罪	318
10.5.1 交互式命令语句	262	12.1.2 计算机实体安全	320
10.5.2 控制语句及常用程序设计		12.1.3 计算机信息安全和信息加密	
方法	263	保护	321
10.5.3 程序调用	267	12.1.4 计算机日常维护	322
10.5.4 数组的使用	270	12.2 计算机病毒防护	323
10.5.5 自定义函数	271	12.2.1 计算机病毒的基本概念	323
10.5.6 程序文件的调试	273	12.2.2 病毒防护方法	327
习题	274	12.3 计算机英语	332
		12.3.1 计算机系统	332
第11章 程序设计基本方法	286	12.3.2 计算机产品规格	334
11.1 程序设计和流程图	286	12.3.3 计算机操作	338
11.1.1 程序设计基本概念和步骤	286	12.3.4 DOS/WINDOWS 的	
11.1.2 程序流程图	287	使用	340
11.1.3 程序控制结构基本知识	290	12.3.5 计算机屏幕信息	343
11.1.4 结构程序设计	294	习题	345
11.2 基本算法	295		
11.2.1 算法概念及其表示	295	附录 参考答案	349

第 1 章 计算机基础知识

1.1 计算机系统组成

计算机系统由软件系统和硬件系统组成。软件系统包括系统软件和应用软件。系统软件有操作系统、程序设计语言、编译系统、通用数据库管理系统等。应用软件有文字处理软件、表处理软件、应用数据库管理系统等。硬件系统包括主机、存储器、输入设备和输出设备。主机主要由中央处理器(CPU)和主存储器(内存储器)组成,此外还包括时钟电路、中断控制电路、DMA 电路、总线、附属电路等。CPU 用于执行运算和控制整个计算机的工作,内存储器用于暂时存储运行中的程序和数据,时钟电路用于产生计算机工作时所必须的时间控制信号,中断控制电路用于中断过程的硬件控制,DMA 电路提供 DMA 过程的硬件控制,总线用于计算机中各部件之间的信息传递。主机是硬件系统最基本的部分,存储器有内存储器和外存储器之分,内存储器包括只读存储器(ROM)、高速缓冲存储器(CACHE)和随机存储器(RAM),外存储器包括软盘存储器、硬盘存储器、磁带存储器、光盘存储器等,输入设备有键盘、鼠标、扫描仪等,输出设备有显示器、打印机、绘图仪等。

1.2 计算机的运算基础

1.2.1 数制及其转换

1. 十进制、二进制、十六进制

(1) 十进制数:十进制数有 0、1、2、3、4、5、6、7、8、9 十个符号,计数时按逢十进一的规则进行。一个十进制数可以写成以 10 为基数按权展开的形式。

[例 1.1] $1234.67 = 1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 + 6 \times 10^{-1} + 7 \times 10^{-2}$

式中 10^3 、 10^2 、 10^1 、 10^0 、 10^{-1} 、 10^{-2} 等称为权,式中某一位置上的符号(0, 1, 2, ..., 8, 9)与权相乘的积表示该位置数值的大小。

(2) 二进制数:由于使用电子元器件表示两种物理状态(电压的高和低,或开关的开和关)容易实现,所以计算机中使用二进制数表示信息和进行运算。

二进制数有 0、1 两个符号,计数时按逢二进一的规则进行。一个二进制数可以写成以 2 为基数按权展开的形式。

[例 1.2] $(10110.11)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2}$

式中 2^4 、 2^3 、 2^2 、 2^1 、 2^0 、 2^{-1} 、 2^{-2} 等称为权,式中某一位置上的符号与权相乘的积表示该位置数值的大小。

(3) 十六进制数:由于二进制数的位数较长,不便记忆,而十六进制数与二进制数转换方便并且位数较少,所以习惯用十六进制数表示二进制数。

十六进制数有 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F 等十六个符号,计数时按逢十六进一的规则进行。一个十六进制数可以写成以 16 为基数按权展开的形式。

【例 1.3】 $(1A3F.C7)_{16} = 1 \times 16^3 + A \times 16^2 + 3 \times 16^1 + F \times 16^0 + C \times 16^{-1} + 7 \times 16^{-2}$
 式中 16^3 、 16^2 、 16^1 、 16^0 、 16^{-1} 、 16^{-2} 等称为权,式中某一位置上的符号与权相乘的积表示该位置数值的大小。

2. 十进制数与二进制数的转换

(1)十进制数转换成二进制数:十进制数转换为二进制数时,整数部分和小数部分分别进行转换,然后把整数部分和小数部分拼接起来形成一个二进制数。整数部分转换的方法是:

- ① 用十进制数除以 2,得到商和余数(0 或 1);
 - ② 再用该商除以 2,又得到商和余数;
 - ③ 重复步骤②,直到商是 1 为止;
 - ④ 最后一次除以 2 得到的商和以前历次除以 2 得到的余数组成转换后的二进制数。
- 权的确定原则是把第一次得到的余数作为二进制数的最低位,最后一次得到的商作为二进制数的最高位。

【例 1.4】把十进制数 44 转换成二进制数。

解:先用 44 多次除以 2,得到余数和最后的商

$$\begin{array}{r}
 2 \overline{) 44} \cdots \text{余 } 0 \\
 2 \overline{) 22} \cdots \text{余 } 0 \\
 2 \overline{) 11} \cdots \text{余 } 1 \\
 2 \overline{) 5} \cdots \text{余 } 1 \\
 2 \overline{) 2} \cdots \text{余 } 0 \\
 1 \cdots \text{商 } 1
 \end{array}$$

按照权的确定原则把商和余数写成二进制数形式,十进制数 44 转换成二进制数是:
 101100

小数部分转换成二进制数的方法是:

- ① 把十进制小数乘以 2,得到积,把积的整数部分(1 或 0)提出;
- ② 再用所得积的小数部分乘以 2,得到积,把积的整数部分再提出;
- ③ 重复步骤②;
- ④ 乘以 2 过程中提出的各个整数部分组成转换后的二进制小数。权的确定原则是最先提出的整数(0 或 1)是二进制小数的最高位。

十进制小数转换成二进制小数时有可能二进制小数位数较多,这种情况下一般根据需要只保留小数点后若干位,其余位舍弃不要。

【例 1.5】把十进制小数 0.38 转换成二进制小数,精确到小数点后 3 位。

解：先多次用 2 乘以 0.38，产生积和整数部分

$$\begin{array}{r}
 0.38 \\
 \times) \quad 2 \\
 \hline
 0.76 \cdots \cdots \text{整数是 } 0 \\
 0.76 \\
 \times) \quad 2 \\
 \hline
 1.52 \cdots \cdots \text{整数是 } 1 \\
 0.52 \\
 \times) \quad 2 \\
 \hline
 1.04 \cdots \cdots \text{整数是 } 1
 \end{array}$$

再把所得整数部分按权的确定原则写成二进制小数形式，即：

$$(0.38)_{10} \approx (0.011)_2$$

十进制数转换二进制数还有一种简便方法，思路是把一个十进制数写成按二进制数权的大小展开的式子，再按二进制数权的大小写成二进制数的形式。

【例 1.6】把 $(894)_{10}$ 转换成二进制数。

解：用简便方法把十进制数 894 转换成二进制数时，第一步找小于 894 的最大的 2^n ，从表 1.1 中找到 $2^9=512$ ， $894-512=382$ 。第二步找小于 382 的最大的 2^n ，从表 1.1 中找到 $2^8=256$ ， $382-256=126$ 。第三步找小于 126 的最大的 2^n ，从表 1.1 中找到 $2^6=64$ ， $126-64=62$ 。第四步找小于 62 的最大的 2^n ，从表 1.1 中找到 $2^5=32$ ， $62-32=30$ 。第五步找小于 30 的最大的 2^n ，从表 1.1 中找到 $2^4=16$ ， $30-16=14$ 。从表 1.1 中查出，14 对应的二进制数是 1110，所以：

$$(894)_{10} = 2^9 + 2^8 + 2^6 + 2^5 + 2^4 + 14 = (110111110)_2$$

【例 1.7】 $(1364.75)_{10} = 2^{10} + 2^8 + 2^6 + 2^4 + 2^2 + 2^{-1} + 2^{-2} = (101010100.11)_2$ 。

【例 1.8】把十进制数 300.4 转换成二进制数，保留小数点后三位。

$$(300.4)_{10} = 2^8 + 2^5 + 2^3 + 2^2 + 2^{-2} + 2^{-3} \approx 100101100.011$$

从上述举例可以看出，使用这种简便方法的基础是要记住常用 2^n 的取值 (n 是可正、可负的整数)。十进制数与二进制数、十六进制数对照表见表 1.1。

表 1.1 常用数制对照表

十进制数	二进制数	十六进制数	十进制数	二进制数	十六进制数
0	0	0	7	111	7
1	1	1	8	1000	8
2	10	2	9	1001	9
3	11	3	10	1010	A
4	100	4	11	1011	B
5	101	5	12	1100	C
6	110	6	13	1101	D
14	1110	E	$2^7=128$	10000000	80
15	1111	F	$2^8=256$	100000000	100

续表

十进制数	二进制数	十六进制数	十进制数	二进制数	十六进制数
$2^4=16$	10000	10	$2^8=256$	100000000	100
$2^5=32$	100000	20	$2^{10}=1024$	10000000000	400
$2^6=64$	1000000	40			

(2) 二进制数转换成十进制数:

二进制数转换成十进制数采用对二进制数各位按权相加的方法。

$$[\text{例 } 1.9] \quad (101101.01)_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = 32 + 8 + 4 + 1 + 0.25 = 45 + 0.25 = (45.25)_{10}$$

3. 二进制数与十六进制数的转换

(1) 二进制数转换成十六进制数: 从表 1.1 可以看出一位十六进制数可以表示四位二进制数。二进制数转换成十六进制数的方法是对一个二进制数以小数为界, 向左向右每四位二进制数用一位十六进制数代替, 分组时前后不足四位的部分分别补零

$$[\text{例 } 1.10] \quad (1\ 0101\ 0100.1100)_2 = (154.C)_{16}$$

(2) 十六进制数转换成二进制数: 十六进制数转换成二进制数的方法是每位十六进制数用四位二进制数代替。

$$[\text{例 } 1.11] \quad (C9)_{16} = (1100\ 1001)_2$$

1.2.2 算术运算和逻辑运算

1. 算术运算

计算机中用二进制数的形式表示信息, 算术运算按二进制运算规则进行。二进制运算规则如下:

(1) 二进制加法运算规则:

$$0+0=0; 0+1=1; 1+0=1; 1+1=10(\text{有进位 } 1)$$

[例 1.12] $001111+00011$ 按二进制加法运算规则:

$$\begin{array}{r} 00111 \\ +) 00011 \\ \hline 01010 \end{array}$$

(2) 二进制减法运算规则:

$$0-0=0; 0-1=1(\text{向高位借 } 1) 1-0=1; 1-1=0$$

[例 1.13] $01101-00011$ 按二进制减法运算规则:

$$\begin{array}{r} 01101 \\ -) 00011 \\ \hline 01010 \end{array}$$

(3) 二进制乘法运算规则:

$$0 \times 0 = 0, 0 \times 1 = 0, 1 \times 0 = 0, 1 \times 1 = 1$$

[例 1.14] 110×11 按二进制乘法运算规则:

$$\begin{array}{r} 110 \\ \times) 011 \\ \hline 110 \\ +) 110 \\ \hline 10010 \end{array}$$

2. 逻辑运算

计算机中用两种状态表示二进制的 0 和 1, 也可以表示逻辑值的“是”与“非”。对逻辑值的运算称为逻辑运算, 逻辑运算的一个特点是按位进行运算, 即逻辑运算中没有进位问题。一般用二进制的“1”表示逻辑的“是”或“真”, 用二进制的“0”表示逻辑的“非”或“假”。逻辑运算中可以和普通代数一样使用字母表示变量, 称为逻辑变量, 逻辑变量的取值只有两种, 取值“0”或取值“1”。

(1) 基本运算: 逻辑代数有三种基本运算。

① 逻辑乘: 也称“与”运算, 运算符为“ $\cdot$ ”或“ $\wedge$ ”。运算规则:

$$0 \cdot 0 = 0 \quad 0 \cdot 1 = 0 \quad 1 \cdot 0 = 0 \quad 1 \cdot 1 = 1$$

使用逻辑变量进行“与”运算时, $A \cdot B$ 可简写成 AB 。

例如 $A=1, B=0$, 则 $A \cdot B = 1 \cdot 0 = 0$ 。

② 逻辑和: 也称“或”运算, 运算符为“ $+$ ”或者“ $\vee$ ”。运算规则:

$$0+0=0 \quad 0+1=1 \quad 1+0=1 \quad 1+1=1$$

③ 逻辑非: 也称“反”运算, 运算符是在逻辑值或变量符号上加“ $-$ ”。运算规则:

$$\bar{0}=1 \quad \bar{1}=0 \quad \text{若 } A=0 \text{ 则 } \bar{A}=1$$

(2) 基本公式

① 0, 1 律:

$$A \cdot 0 = 0 \quad A \cdot 1 = A \quad A + 0 = A \quad A + 1 = 1$$

② 交换律:

$$A+B=B+A \quad A \cdot B=B \cdot A$$

③ 结合律:

$$A+B+C=(A+B)+C=A+(B+C) \quad A \cdot B \cdot C=(A \cdot B) \cdot C=A \cdot (B \cdot C)$$

④ 分配律: $A \cdot (B+C) = A \cdot B + A \cdot C$

⑤ 重叠律:

$$A+A=A \quad A+A+A+\cdots+A=A \quad A \cdot A=A \quad A \cdot A \cdot \cdots \cdot A=A$$

⑥ 互补律:

$$A+\bar{A}=1 \quad A \cdot \bar{A}=0$$

⑦ 吸收律:

$$A + A \cdot B = A \quad A \cdot (A + B) = A \quad A + \bar{A} \cdot B = A + B \quad A \cdot (\bar{A} + B) = A \cdot B$$

⑧ 对合律:

$$\overline{\overline{A}} = A$$

⑨ 德·摩根定理:

$$\overline{(A + B)} = \bar{A} \cdot \bar{B} \quad \overline{A \cdot B} = \bar{A} + \bar{B}$$

(3) 真值表:是逻辑变量之间逻辑运算关系的一种表达形式。在遇到一个逻辑问题时,首先要根据逻辑运算关系列出真值表,然后再根据真值表列出逻辑表达式。常用逻辑运算的真值表如表 1.2 所示。

表 1.2 常用逻辑运算真值表

A	B	$A \cdot B$	$A + B$	$\bar{A} \cdot B + A \cdot \bar{B}$
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

上述真值表中 A 和 B 是逻辑变量; AB 、 $A+B$ 、 $\bar{A}B + A\bar{B}$ 是 A、B 的逻辑函数,逻辑函数指明了逻辑变量之间的逻辑运算。逻辑函数的取值是根据逻辑变量的值以及逻辑运算规则而产生。

(4) 逻辑表达式:是由逻辑变量、逻辑常量以及逻辑运算符组成的表达式,它是逻辑变量的函数。给出逻辑表达式可以列出其真值表,但是实际问题中常需要根据真值表列出逻辑表达式。列出逻辑表达式时只需把逻辑函数中那些取值为“1”的行分别对应一项,各项之间用“或”运算,而每一项是各逻辑变量或其“非”的“与”运算。取逻辑变量还是取其“非”值取决于真值表该行中它的值是“1”还是“0”。以表 1.2 中逻辑函数 $A+B$ 为例,从表中可以看出,在 $AB=01$ 或 $AB=10$ 或 $AB=11$ 时逻辑函数 $A+B=1$ 。 $AB=01$ 的逻辑“与”运算写成 $\bar{A} \cdot B$, $AB=10$ 的逻辑“与”运算写成 $A \cdot \bar{B}$, $AB=11$ 的逻辑“与”运算写成 $A \cdot B$ 。列出的逻辑表达式为: $F = \bar{A} \cdot B + A \cdot \bar{B} + A \cdot B$ 。它可以化简为 $A+B$ 。

(5) 逻辑表达式化简:目的是通过逻辑化简使得硬件上减少逻辑门电路的数量或减少逻辑门输入端的个数;其原则是尽量减少逻辑表达式中的项数或每个项中的变量个数;化简常用的方法有:利用公式化简和利用卡诺图化简等。考试时考生任选自己熟悉的方法进行化简。

公式化简是利用上述基本公式,对逻辑函数进行等值变换,达到减少表达式项数或减少每一项中变量个数的目的。

卡诺图是逻辑表达式的另一种真值表,也是一种有效的化简工具。卡诺图化简是利用卡诺图尽量合并最小项,最终达到化简的目的。三个变量的卡诺图如下所示(只要给出了三个变量的逻辑表达式,就可以写出图中各个小方块中的值):

		AB							
		C							
		0 0		0 1		1 1		1 0	
0		m0		m2		m6		m4	
1		m1		m3		m7		m5	

卡诺图中每一小方块对应一个最小项,所谓最小项就是真值表的一行中各逻辑变量或其“非”的逻辑乘积项,其中: $m_0=\bar{A}\bar{B}\bar{C}$ 、 $m_1=\bar{A}\bar{B}C$ 、 $m_2=\bar{A}B\bar{C}$ 、 $m_3=\bar{A}BC$ 、 $m_4=AB\bar{C}$ 、 $m_5=ABC$ 、 $m_6=AB\bar{C}$ 、 $m_7=ABC$ 。利用卡诺图化简就是把卡诺图中的最小项按合并规则尽量合并,以达到减少表达式项数或每项中变量个数的目的。三个变量的情况下,最小项合并规则简单地讲是:相邻的两个或四个最小项可以合并为一项,例如最小项 ABC 和 ABC 可合并为一项 AB ;中间间隔两个最小项的最小项可以合并,例如 $\bar{A}\bar{B}\bar{C}$ 和 $A\bar{B}\bar{C}$ 可合并;中间间隔一个最小项的两个最小项或对角的两个最小项不能合并,例如 $\bar{A}\bar{B}\bar{C}$ 和 $AB\bar{C}$ 不能合并, $\bar{A}BC$ 和 ABC 也不能合并。

[例 1.15] 有一真值表(见右),请化简。

解:写出 F 的逻辑表达式。根据真值表,把真值表中逻辑函数 F 的值等于 1 的那些最小项列出,即 $F=\bar{A}\cdot\bar{B}+A\cdot\bar{B}+A\cdot B$,利用公式对 F 的逻辑表达式进行化简:

A	B	F
0	0	1
0	1	0
1	0	1
1	1	1

$$\begin{aligned}
 F &= \bar{A} \cdot \bar{B} + A \cdot \bar{B} + A \cdot B \\
 &= \bar{A} \cdot \bar{B} + A(\bar{B} + B) \quad (\text{利用分配律}) \\
 &= \bar{A} \cdot \bar{B} + A \quad (\text{利用互补律以及 0、1 律}) \\
 &= A + \bar{B} \quad (\text{利用吸收律})
 \end{aligned}$$

也可用卡诺图对 F 的逻辑表达式进行化简:

F 的逻辑表达式: $F=\bar{A}\cdot\bar{B}+A\cdot\bar{B}+A\cdot B$

用卡诺图进行化简时先把表达式中的每个最小项按其在卡诺图中的位置标在卡诺图中,标出后如下所示:

		A	
		0	1
0	B	✓	✓
1	B		✓

再根据最小项合并规则对卡诺图中的最小项进行合并,从卡诺图上可以看出 $\bar{A}\cdot\bar{B}$ 项和 $A\cdot\bar{B}$ 项可以合并为 $\bar{B}$, $A\cdot\bar{B}$ 项和 AB 项可以合并为 A ,最后列出逻辑表达式 $F=A+\bar{B}$ 。

3. 逻辑运算的应用

程序设计中经常需要使用逻辑运算对数据进行处理。常用的逻辑运算有“与”运算、“或”运算、“非”运算、“异或”运算($A\oplus B=A\bar{B}+\bar{A}B$)等。为了下面讲述方便,设变量 A 的

内容是 $d_7, d_6, d_5, d_4, d_3, d_2, d_1, d_0$ 八位二进制数, 其中 d_7 为最高位。

(1) 位操作运算: 使用逻辑运算可以对二进制的指定位进行操作, 即使用逻辑运算将某一个二进制位置“1”或置“0”。

【例 1.16】将变量 A 的 d_6 位清“0”。

解: 这种情况可利用“逻辑与”的特点(一个数和 1 相“与”, 该数保持不变; 一个数和 0 相“与”, 该数清 0)把指定位清 0, 即 $A \cdot (11011111) \rightarrow A$ 。

【例 1.17】将变量 A 的 d_6 位置“1”。

这种情况可利用“逻辑或”的特点(一个数和 0 相“或”, 该数保持不变; 一个数和 1 相“或”, 该数恒置 1)。

(2) 字节操作: 使用逻辑运算对一个字节进行操作。

【例 1.18】把变量 A 的内容全部清“0”。

解: 全部清“0”有两种方法, 一种是利用“逻辑与”的特点清“0”, 第二种是利用“异或”特点清“0”(一个变量自身进行“异或”操作, 其结果是把该变量内容清“0”), 即 $A \oplus A \rightarrow A$ 。

【例 1.19】把变量 A 的各位置“1”。

解: 可利用“逻辑或”的特点实现变量 A 置“1”, 即 $A + (11111111) \rightarrow A$ 。

【例 1.20】把变量 A 的各位一律求反。

解: 求反有两种方法, 一种是利用“逻辑非”运算, 即 $\bar{A} \rightarrow A$; 另一种是利用“异或”运算求反(二进制数各位与“1”相“异或”, 其结果是把它求反), 即 $A \oplus (11111111) \rightarrow A$ 。

(3) 部分字节操作: 部分字节操作就是使用逻辑运算对一个字节内的几位进行操作。

【例 1.21】把变量 A 的低四位清“0”, 高四位保持不变。

解: 使用“与”运算, 不变部分“与”1, 清“0”部分“与”0, 即 $A \cdot (11110000) \rightarrow A$ 。

【例 1.22】设有变量 A 和 B, 要求把 A 的高四位和 B 的低四位拼成一个字节送给变量 C。

实现: $[A \cdot (11110000)] + [B \cdot (00001111)] \rightarrow C$

1.3 机内数据表示形式

1.3.1 原码、反码和补码

计算机中参加运算的数有正负之分, 计算机中数的正负号用二进制数表示。用二进制数表示符号的数称为机器码, 常用的机器码有原码、反码和补码。

1. 原码

求原码的简单方法: 设 X; 若 $X \geq 0$, 则符号位(原码最高位)为 0, X 其余各位取值照抄; 若 $X \leq 0$, 则符号位为 1, X 其余各位取值照抄。

【例 1.23】 $X = +1001001$ $[X]_{\text{原}} = 01001001$

【例 1.24】 $X = -1001001$ $[X]_{\text{原}} = 11001001$