

计算机及应用专业 专科

全国高等教育自学考试

计算机组成原理习题详解

黄明 梁旭 编著



机械工业出版社
CHINA MACHINE PRESS



全国高等教育自学考试

计算机组成原理 习题详解

(计算机及应用专业 专科)

黄 明 梁 旭 编著



机械工业出版社

本书是根据“全国自学考试（计算机及应用专业 专科）考试大纲”以及历年考题编写的。本书共分 4 部分：第 1 部分是笔试应试指南；第 2 部分是笔试题解；第 3 部分是模拟试卷及参考答案；最后是附录，包括考试大纲和 2002 年下半年试卷。

本书紧扣考试大纲，内容取舍得当，叙述通俗易懂，附有大量与考试题型类似的习题及答案，以检查读者对考点的掌握程度。

本书适用于准备参加全国自学考试（计算机及应用专业 专科）的考生，也可作为大专院校和培训班的教学参考书。

图书在版编目（CIP）数据

计算机组成原理习题详解/黄明，梁旭编著. —北京：机械工业出版社，2004.3
（全国高等教育自学考试）

ISBN 7-111-14050-8

I. 计... II. ①黄...②梁... III. 计算机体系结构—高等教育—自学考试—解题 IV. TP303-44

中国版本图书馆 CIP 数据核字（2004）第 012668 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：胡毓坚

责任编辑：孙 业

责任印制：施 红

北京忠信诚胶印厂印刷·新华书店北京发行所发行

2004 年 3 月第 1 版·第 1 次印刷

787mm×1092mm 1/16·10.5 印张·250 千字

0001—5000 册

定价：17.00 元

凡购本图书，如有缺页、倒页、脱页，由本社发行部调换

本社购书热线电话（010）68993821、88379646

封面无防伪标均为盗版

出版说明

全国高等教育自学考试指导委员会推出面向社会的高等自学考试，经过 10 多年的实践，已建立起一整套较为完善的规章制度和操作流程，考试组织严密规范，考试纪律严格；坚持考试标准，实行教考分离，确保了毕业生的质量。它为没有机会进入高等学校的中国公民提供了接受高等教育的机会，并以严格的国家考试保证了毕业生的质量，获得了普遍赞誉。国家自考中心于 2002 年开始执行新的考试计划。新计划中开设的专业共 224 个，其中专科 141 个约占 63%，独立本科段 61 个约占 27%，专本衔接专业 22 个约占 10%。为帮助、指导广大自考生深入理解计算机及相关专业考试的基本概念，灵活运用基本知识，掌握解题方法和技巧，熟悉考试模式，进一步提高应试能力和计算机水平，特编写了以下专业的基础课与专业课主要课程的习题详解。

- ◇ 计算机及应用专业 独立本科段
- ◇ 计算机信息管理专业 独立本科段
- ◇ 计算机网络专业 独立本科段
- ◇ 计算机及应用专业 专科

丛书特点：

1. 以 2002 年最新考试大纲为基准。

本丛书是根据 2002 年最新考试大纲，为参加全国高等教育自学考试考生编写的一套习题详解教材。

2. 例题反映了历届考试中的难度和水平。

书中对大量的例题进行了分析，所选例题都是在对最近几年考题深入研究的基础上精心筛选的，从深度和广度上反映了历届考试中的难度和水平。

3. 作者经验丰富。

本丛书的作者都是多年从事全国高等教育自学考试辅导的高等院校的教师。

读者对象：

- ◇ 准备参加全国高等教育自学考试的考生。
- ◇ 计算机及相关专业的本专科生。

前言

自学考试是对自学者进行以学历考试为主的高等教育国家学历考试。为帮助、指导广大考生深入理解自学考试的基本概念，灵活运用基本知识，掌握解题方法和技巧，熟悉考试模式，进一步提高应试能力和计算机水平，特编写了本书。

全书共分 4 部分，即笔试应试指南、笔试题解、模拟试卷及参考答案和附录。书中所选试题均是在对历年真题深入研究的基础上精心设计的，从深度和广度上反映了考试的难度和水平。模拟试卷的题型分配与真题一致，这些题目是考试指导教师的多年积累，在辅导班中实际多次使用过的。

书中附录给出了“全国自学考试（计算机及应用专业 专科）计算机组成原理考试大纲”，以及“2002 年下半年全国自学考试（计算机及应用专业 专科）计算机组成原理试卷及参考答案”。

本书由黄明、梁旭编写。

由于编者水平有限，编写时间仓促，书中错误和不妥之处在所难免，请读者和专家批评指正。

读者在使用本书的过程中如有问题，可通过 E-mail 与我们联系：

dlhm@263.net

编 者

目 录

出版说明

前言

第 1 部分 笔试应试指南

1.1 笔试应试策略	2
1.2 笔试考点归纳	3
1.2.1 概论	3
1.2.2 数据编码和数据运算	5
1.2.3 存储系统	12
1.2.4 指令系统	16
1.2.5 控制器	19
1.2.6 系统总线	23
1.2.7 外围设备	25
1.2.8 输入输出系统	31

第 2 部分 笔 试 题 解

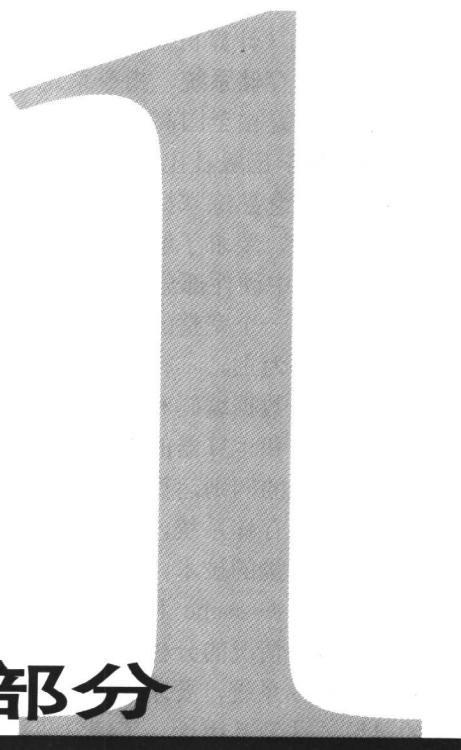
2.1 概论	38
2.1.1 单项选择题	38
2.1.2 改错题	41
2.1.3 名词解释题	41
2.1.4 简答题	41
2.1.5 习题	42
2.2 数据编码和数据运算	43
2.2.1 单项选择题	43
2.2.2 改错题	47
2.2.3 名词解释题	48
2.2.4 简答题	48
2.2.5 计算题	49
2.2.6 设计题	51
2.2.7 习题	51
2.3 存储系统	54
2.3.1 单项选择题	54
2.3.2 改错题	58
2.3.3 名词解释题	59

2.3.4	简答题	59
2.3.5	计算题	60
2.3.6	设计题	61
2.3.7	习题	62
2.4	指令系统	65
2.4.1	单项选择题	65
2.4.2	改错题	68
2.4.3	名词解释题	69
2.4.4	简答题	70
2.4.5	分析题	71
2.4.6	习题	72
2.5	控制器	76
2.5.1	单项选择题	76
2.5.2	改错题	79
2.5.3	名词解释题	80
2.5.4	简答题	81
2.5.5	分析题	82
2.5.6	习题	82
2.6	系统总线	85
2.6.1	单项选择题	85
2.6.2	改错题	88
2.6.3	名词解释题	89
2.6.4	简答题	89
2.6.5	计算题	90
2.6.6	习题	91
2.7	外围设备	94
2.7.1	单项选择题	94
2.7.2	改错题	96
2.7.3	名词解释题	97
2.7.4	简答题	98
2.7.5	计算题	99
2.7.6	习题	100
2.8	输入输出系统	103
2.8.1	单项选择题	103
2.8.2	改错题	107
2.8.3	名词解释题	109
2.8.4	简答题	109
2.8.5	计算题	111
2.8.6	分析题	111

2.8.7 习题	111
2.9 习题参考答案	114

第3部分 模拟试卷及参考答案

3.1 模拟试卷一及参考答案	130
3.1.1 模拟试卷一	130
3.1.2 参考答案	132
3.2 模拟试卷二及参考答案	134
3.2.1 模拟试卷二	134
3.2.2 参考答案	137
附录	141
附录 A 全国自学考试（计算机及应用专业 专科） 计算机组成原理自学考试大纲	142
附录 B 2002 年下半年全国自学考试 计算机组成原理试卷及参考答案	151
参考文献	157



第1部分

笔试应试指南

笔试应试策略

笔试考点归纳

1.1 笔试应试策略

全国自学考试(计算机及应用专业 专科)计算机组成原理考试大纲涵盖了概论、数据编码和数据运算、存储系统、指令系统、控制器、系统总线、外围设备、输入输出系统 8 章内容。使用的教材是由全国高等教育自学考试指导委员会组编,胡越明编著的《计算机组成原理》,由经济科学出版社出版。考试复习的过程中要紧紧围绕大纲的知识点,首先对大纲涉及的各章基本概念熟练掌握。

第 1 章为概论,要求了解计算机组成原理的基本知识。了解整个计算机系统由硬件和软件两部分构成,其中硬件部分包括运算器、控制器、存储器、输入输出设备等五大功能部件,通过总线相互连成一个完整的硬件系统;软件部分包括系统软件、应用软件和程序设计语言。本章以识记性内容为主。

第 2 章内容是数据编码和数据运算,要求掌握二进制数据编码的方法,掌握编码数据的加减乘除运算方法和运算器的构成,能够按步骤进行原码和补码的加减乘除四则运算,能够按步骤进行浮点数的四则运算。

第 3 章内容是存储系统,要求掌握存储器芯片的类型和各主要存储器芯片的工作原理;掌握扩展存储器容量的技术,能够用给定的存储器芯片按要求设计主存,从而深刻理解存储器的构成原理;掌握 cache 和虚拟存储器的构成原理,能够分析 cache 和虚拟存储器的命中情况。通过对命中情况的分析,对 cache 和虚拟存储器的工作原理有深入的理解。

第 4 章是指令系统,要求理解指令中应包含的信息,掌握常用指令及指令格式,深入理解常用的寻址方法及其用途,了解常见指令的种类和功能。

第 5 章是控制器,要求掌握 CPU 的功能及组成方式,掌握控制器的控制方式,掌握组合逻辑控制器和微程序控制器的工作原理和设计方法,深入了解计算机中一条指令的完整执行过程。

第 6 章是系统总线,要求掌握总线的基本概念、总线的类型、总线控制器和总线接口的结构原理;理解总线在计算机系统中的作用,深入了解总线的裁决方式和总线的同步方式。

第 7 章是外围设备,要求对计算机外围设备的特点和发展趋势有清楚的了解,了解外围设备的类型和主要外围设备的工作原理。

第 8 章是输入输出系统,要求掌握计算机输入输出系统的基本功能和原理,理解 CPU 与外围设备交换数据的方式;掌握数据传输的控制方式、中断的概念、DMA 以及输入输出通道的原理;能够区别各种输入输出方式的特点,能够分析和设计中断判优电路,能够分析和设备中断屏蔽字。

在复习时要根据大纲提供的考核点和考核要求来进行复习,这样就能抓住重点,进行有效复习,在做练习时,要根据考试的题型进行练习,在掌握基本概念的基础上,掌握一定的解题技巧。操作系统概论的考试题型有:单选题、多选题、填空题、简答题、计算题和应用题等题型。对于不同题型,要采用不同的答题方法。操作系统概论的考试题型有:

单选题:这种题型可考查考生的理解、推理分析,综合比较能力。在答题时,如果可以直接得出正确答案,对于没有太大把握的试题,也可以采用排除法,经过分析比较逐步排除错误答案,最终选定正确答案。

改错题：这种题型着重考核考生对概念、基本原理理解的准确性，在复习的过程中对考试大纲涉及的一些基本概念和原理要熟练掌握。

简答题：这种题型一般围绕基本概念、原理及其联系进行命题，着重考核考生对概念、知识、原理的掌握、辨别和理解能力。在答题时，答案要有层次性，条理清晰，列出要点，同时加以简要扩展就可以。

计算题：这种题型灵活性比较大，着重考核考生对概念、知识、原理的掌握和计算的能力。在答题时，要先明确题目的要求，然后根据进行推理、计算。

设计题：这种题型着重考核考生分析、解决实际问题的能力，考核考生综合应用能力和创见性。在答题时，要综合运用所学知识进行分析和设计。

考生在复习时在掌握知识点的同时也应抓住这些题型的特点，这样才能达到好的应试效果。

1.2 笔试考点归纳

1.2.1 概论

1. 计算机的硬件构成

(1) 计算机的五个基本功能部件及其相互关系

计算机硬件由运算器、存储器、控制器和输入输出设备五部分组成。计算机中的各功能单元一般通过一条或者几条称为总线（bus）的公共线路进行连接。这些硬件与软件一起构成完整的计算机系统。

(2) 计算机各功能部件的基本功能

1) 运算器。运算器是完成运算功能的部件，主要包括逻辑运算单元(AUL)和寄存器。逻辑运算单元执行各种数据运算操作，其中包括算术运算和逻辑运算。寄存器是在运算器中临时存放数据的部件。

2) 存储器。存储器的作用是存储程序和数据。分为主存储器(又称主存或内存)和辅助存储器(又称外存)在数字计算机中，信息以二进制数据的代码形式表示。任何数字、文字或者指令都表示成二进制数的编码。其中每个二进制数据代码称为“位”(bit)，它是数据的最小表示单位。

主存储器由大量的数据存储单元构成，每个存储单元可存储一位信息。数据的存储一般是以“字”(word)为单位进行。

存储器的数据读操作和写操作统称为存储器的访问(access)。访问存储器中一个字的数据所需的时间称为存储器周期时间。存储器的访问可分为随机访问、顺序访问和部分顺序访问。

3) 控制器。控制器在计算机指令的控制下进行工作。控制器对指令进行译码，并根据指令生成一系列时序控制信号，控制其他单元的工作。它是计算机的神经中枢。

数据是编码形式的各种信息，它在计算机中通常作为程序的操作对象。在计算机中，数据可以是整数、浮点数的编码，也可以是声音信息、图像信息的编码，还可以是程序代码等。

计算机的中央处理器 CPU 由运算器以及控制器构成，它是计算机的核心部件。

把 CPU、存储器和输入输出接口合在一起构成的处理系统称为主机，我们通常所说的

主机是不包括输入输出设备的。

指令流是存储器和 CPU 之间形成的不断传递的指令序列, 指令流从存储器流向控制器。

数据流是存储器和 CPU 之间形成的不断传递的数据序列, 数据流存在于运算器与存储器以及输入输出设备之间。

4) 输入输出设备。计算机从输入设备获得外部的信息。输入设备将外部信息以一定的数据格式送入系统, 如键盘、鼠标等等。

输出设备的功能是将计算机的处理结果提供给外部世界。如打印机、显示器等等。

计算机的输入输出设备通常称为外围设备。磁带、磁盘、光盘等外存储器通常也称为外围设备。外围设备通常通过输入输出接口与主机连接。输入输出接口是主机与外围设备之间传递数据与控制信息的电路。

2. 计算机软件概述

(1) 软件的基本概念及其和硬件之间的关系

计算机软件是由程序构成的系统, 负责将指令组织起来完成复杂的工作。计算机软件一般可分为系统软件和应用软件两类。

(2) 系统软件、应用软件的概念

系统软件是整个计算机系统的一部分, 使得计算机系统的功能完整。如操作系统、编译程序、解释程序、汇编程序。应用软件是面向用户应用的功能软件, 专门为解决某个应用领域中的具体任务而编写。如计算机辅助设计 (CAD) 软件、数据处理软件、控制软件、模拟软件。

应用软件、系统软件和硬件构成了计算机系统的三个层次。

(3) 计算机系统的层次结构以及虚拟机的概念

应用软件为用户提供一个应用系统的界面, 使用户能够方便地使用计算机解决具体问题。系统软件则向用户提供一个基本的操作界面, 并向应用软件提供功能上的支持。硬件系统是整个计算机系统的基础和核心, 所有的功能最终由硬件完成。

虚拟机 (virtual machine) 是在实际硬件机器级外面的所有机器层次, 包括系统软件和应用软件, 它们都是由软件构成的计算机外部特性。

(4) 软件与硬件的等价性概念

计算机系统的大部分功能既可以用硬件实现, 也可以用软件实现。计算机功能的这两种实现在逻辑上是等效的, 其区别在于速度、成本、可靠性、存储容量和变更周期等因素。

3. 程序设计语言及其编译

(1) 机器语言、汇编语言、高级语言和应用语言的概念

机器语言是一种用二进制代码表示的能够被计算机硬件直接识别和执行的语言。机器语言与计算机的结构有关。在不同的计算机中, 机器语言一般是不同的。

汇编语言是采用文字方式 (助记符) 表示的程序设计语言。汇编语言的大部分指令是和机器语言中的指令一一对应的, 但不能被计算机的硬件直接识别。它仍然与计算机的结构特征相关。

高级语言是与计算机结构无关的程序设计语言。它具有更强的表达能力, 可方便地表示数据的运算和程序的控制结构, 能更好地描述各种算法, 而且容易学习掌握。

应用语言用于表示人们应用计算机完成各种任务的要求。与高级语言相比,它更加接近于人类的自然语言,因而使用更加方便。

(2) 编译和解释的概念

用程序设计语言编写的程序称为源程序。高级语言的源程序可以通过两种方法转换成机器语言程序,即目标程序。一种是编译,另一种是解释。

通过编译程序在运行之前将源程序转换成机器语言,这种方法叫编译。

通过解释程序进行解释,即执行逐个解释并立即执行源程序的语句,这种方法叫解释。

1.2.2 数据编码和数据运算

1. 数据与文字的编码

(1) 各种进位计数制,二、八、十和十六进制数的相互转换

八进制数和十六进制数与二进制数的转换十分方便。在将二进制数转换成八进制数时,因为3位二进制数正好对应1位八进制数,所以可将二进制数分成3位一组转换成八进制数。如果要将二进制数转换成十六进制数时,则可将二进制数分成4位一组进行转换。

十进制数到二进制数的转换可以采用除2取余的方法。即把十进制数除以2,所得余数作为二进制数的最低位,再除以2,所得余数作为次低位,如此重复,直到商数为零为止。然后将余数按次序连起来形成二进制数。

(2) 定点整数和定点小数的编码

1) 定点整数的编码。将小数点固定在数的最低位之后,这就是定点整数形式。

2) 定点小数的编码。将小数点固定在最高位的后面,即纯小数表示。

(3) 原码、补码、反码和移码的概念、编码方法和定义范围

1) 原码。原码表示法中用一个符号位表示数据的正负,用0代表正号,1代表负号,其余的代码表示数据的绝对值。

对于给定的二进制整数 $x = x_0x_1x_2 \cdots x_n$,原码的编码方法是:当 $x \geq 0$ 时, $[x]_{\text{原}}$ 的代码是 $0x_1 \cdots x_n$;当 $x < 0$ 时, $[x]_{\text{原}}$ 的代码是 $1x_1 \cdots x_n$ 。这里最高位 x_0 是符号位, $x_1 \cdots x_n$ 是数据的二进制值。

整数原码所能表示的数值范围是: $-2^n + 1 \leq x \leq 2^n - 1$

定点小数原码的数值范围是: $-1 + 2^{-n} \leq x \leq 1 - 2^{-n}$

2) 补码。补码表示法也是用最高一位代表符号,其余各位代码给出数值按2取模的结果。对于一个 $n+1$ 位的二进制整数 $x = x_0x_1 \cdots x_n$ 。其补码的编码方式为:正数的补码与原码相同;负数的补码是将二进制位按位取反后,在最低位上加1。

整数补码所能表示的数值范围是: $-2^n \leq x \leq 2^n - 1$

定点小数补码所能表示的数值范围是: $-1 \leq x \leq 1 - 2^{-n}$

3) 反码。反码表示法是用对负值的各位取反的表示方法,反码的最高位也是符号位。0表示正数,1表示负数。

对于 $n+1$ 位的二进制整数 $x = x_0x_1x_2 \cdots x_n$,反码的编码方法是:正数的反码与原码相同;负数的反码是将二进制位按位取反,也就是作按位求反运算。

$n+1$ 位的整数反码的数值范围: $-2^n + 1 \leq x \leq 2^n - 1$

定点小数反码的数值范围: $-1 + 2^{-n} \leq x \leq 1 - 2^{-n}$

4) 移码。对于 $n+1$ 位二进制整数 $x = x_0x_1 \cdots x_n$,移码的定义为: $[x]_{\text{移}} = 2^n + x$,也就是取其

加上 2^n 后的二进制数编码。

移码的数值范围为: $-2^n \leq x \leq 2^n - 1$

在带符号的移码表示中, 最高位仍可看作是符号位, 但 1 表示正号, 0 代表负号。将移码与补码相比较, 可以发现只是符号位不同, 其余位相同。

(4) 浮点数的编码原理、数值范围和规格化方法

在浮点数据编码中, 数据代码分为尾数和阶码两部分。尾数表示有效数字; 阶码则表示小数点的位置。加上符号位, 浮点数据通常表示为:

$$N = (-1)^S \times M \times R^E$$

其中 M (mantissa) 是浮点数的尾数, R (radix) 是基数, E (exponent) 是阶码, S (sign) 为数据的符号位。

在浮点数中, 当数据的绝对值太大, 以至于大于阶码所能表示的数据(阶码上溢)时, 称为浮点数的上溢 (overflow)。而当数据的绝对值太小, 以至于小于阶码所能表示的数据(阶码下溢)时, 则称为浮点数的下溢(underflow)。

当尾数用补码表示时, 规格化后正尾数的第一位为 1, 负尾数的第一位为 0, 即正尾数的代码形式为 $0.1xxx \cdots x$; 负尾数的代码形式为 $1.0xxx \cdots x$ 。

(5) 浮点数的编码的 IEEE 标准

在 IEEE 754 浮点数标准中, 符号位 s 仍然用 0 表示正数, 用 1 表示负数。正常数的阶码取值 e 的范围为 $1 \sim 254$, 尾数部分可以取任意的二进制数值 f 。这样, 单精度数所表示的数值为:

$$(-1)^s \times 1.f \times 2^{e-127}$$

IEEE 754 双精度数所表示的数值为:

$$(-1)^s \times 1.f \times 2^{e-1023}$$

(6) 西文字符和汉字字符的编码标准

对西方文字的编码通常用 7 位或者 8 位二进制的数表示一些字母、数字符号、标点符号和一些控制符号等, 通常采用一个字节表示一个字符信息。文字字符的编码方案有许多, 目前国际上普遍采用的一种字符编码系统是 ASCII 码。

汉字在计算机中的编码可分为输入码和机内码等。机内码是汉字在计算机内部进行存储和处理时采用的表示形式, 它同样是一种二进制代码。汉字机内码是用于汉字信息存储、交换、检索等操作的内部代码, 一般采用两个字节表示一个汉字。

(7) 十进制数的编码

十进制数据在计算机中主要有两种表示形式:

1) 字符串形式, 即一个字节存放一个十进制的数位或符号。用连续的多个字节表示一个完整的十进制数据。

2) 压缩的十进制数串形式, 即一个字节存放两个十进制的数位。它比较节省存储空间, 又便于进行算术运算。

(8) 奇偶校验码、海明码和循环码的概念和编码方法

能够发现某些错误或具有自动纠错能力的数据编码称为数据校验码或检错码。

常用的数据校验分组码有奇偶校验码、海明校验码和循环冗余校验码(CRC)等。

1) 奇偶校验码。奇偶校验码是一种最简单的检错码, 它在每 k 个信息位中增加 1 个校

验位代码,能发现数据代码中一位出错情况,常用于对存储器数据的检查或者传输数据的检查。偶校验码的构成规则是:所有信息位和单个校验位的模2加等于0,即每个码字(包括校验位)中1的数目为偶数。

奇校验码的构成规则是:所有信息位和单个校验位的模2加等于1,即每个码字中1的数目为奇数。表示为:

$$\text{奇校验: } x_1 + x_2 + \cdots + x_k + x_{k+1} = 1 \quad \text{mod } 2$$

$$\text{偶校验: } x_1 + x_2 + \cdots + x_k + x_{k+1} = 0 \quad \text{mod } 2$$

其中 $x_1, x_2, \dots, x_k$ 为信息位, x_{k+1} 为校验位。

2) 海明码。海明码是一种纠错码,海明码能检测出2位错误,并能纠正1位错误。其原理是在一个数据位组中加入几个校验位,增加数据代码间的码距,当某一位发生变化时就会引起校验结果发生变化,不同代码位上的错误会得出不同的校验结果。

3) 循环码。循环码也是一种纠错码,循环码C是具有下述特性的(n,k)分组码:若一个n位编码是码C的一个码字(合法代码),那么它向右循环移动一位后的n位编码也是码C的一个码字。

循环冗余校验码(CRC)的校验方法是基于模2运算建立有效信息和校验位之间关系的方法。

一个8位信息码($b_7b_6b_5b_4b_3b_2b_1b_0$)可以用一个7次二进制多项式表示:

$$B(x) = b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x^1 + b_0x^0$$

每个循环码都有它自己的生成多项式G(x),循环码中的每一个码字多项式都是这个生成多项式与信息码多项式的乘积,每个循环码的码字都可由这个生成多项式生成。选择不同的生成多项式可得到不同的循环码。

2. 定点数加减法运算

(1) 补码加法和减法的方法

补码的一个重要特点是它可以直接进行加减法运算。定点小数的补码加法公式是:

$$[x]_{\text{补}} + [y]_{\text{补}} = [x+y]_{\text{补}} \quad (\text{mod } 2)$$

两个数的补码相加,其结果就是这两个数相加后的补码。定点小数的补码的减法公式是:

$$[x-y]_{\text{补}} = [x]_{\text{补}} - [y]_{\text{补}} = [x]_{\text{补}} + [-y]_{\text{补}} \quad (\text{mod } 2)$$

(2) 溢出的概念和检测的方法

在计算机中,每种数据编码都有其数据表示范围。在运算过程中如出现数据超出这个表示范围的现象,就发生了溢出。

检测的方法之一:

对于加法,只在正数加正数和负数加负数两种情况下才会产生溢出,符号不同的两个数相加是会产生溢出的。

对于减法,只在正数减负数和负数减正数两种情况下才会产生溢出,符号相同的两个数相减是会产生溢出的。

检测的方法之二:

另一种判别运算溢出的方法是采用双符号位补码。正常时两个符号位的值相同,在运算结果中,当两个符号位不同时则表明发生了溢出。运算结果的符号位01表明两个整数相加,结果大于机器所能表示的最大正数,称为上溢;运算结果的符号位10表明两个负数相加,

结果小于机器所能表示的最小正数，称为下溢；也就是说，两个正数相加，数值位不应该向符号位产生进位，使得结果数的符号位和操作数的一样，为 00：

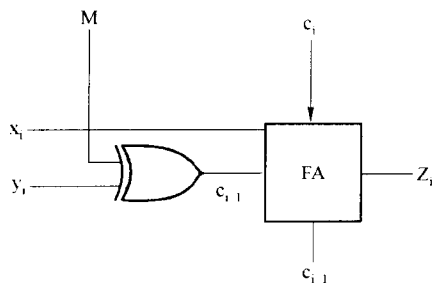
$$00+00+00(\text{进位})=00 \quad (\text{mod } 4)$$

两个负数相加，数值位应该向符号位产生进位，使得两个负数的双符号位的运算为 11：

$$11+11+01(\text{进位})=11 \quad (\text{mod } 4)$$

(3) 定点加法器和减法器的结构原理

如下图所示：



(4) 快速进位的原理

当且仅当 4 位组中的每一位都传递进位时，这个 4 位组传递进位。产生的规律是：当且仅当 4 位组的最高一位产生进位时，这个 4 位组产生进位。

3. 定点数乘除法运算

(1) 原码和补码一位乘法的运算方法及其算法流程

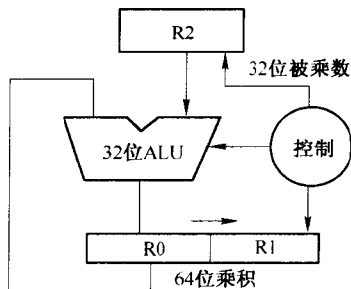
原码的乘法将符号位与数值位分开进行运算，运算结果的数值部分是乘数和被乘数数值位的乘积，运算结果的符号位是乘数和被乘数符号位的异或。

对补码直接进行乘法运算时，只需对被乘数进行符号扩展，构成双倍字长的数据即可。

典型的带符号乘法的方法是布斯(Booth)算法。

(2) 乘法器的结构原理

一个 32 位乘法器的结构框图如下图所示，其中 32 位被乘数放在 R2 中，运算开始时 32 位乘数放在 R1 中，运算结束时 64 位乘积的高位放在 R0 中，低位放在 R1 中，R0 和 R1 单联移位。



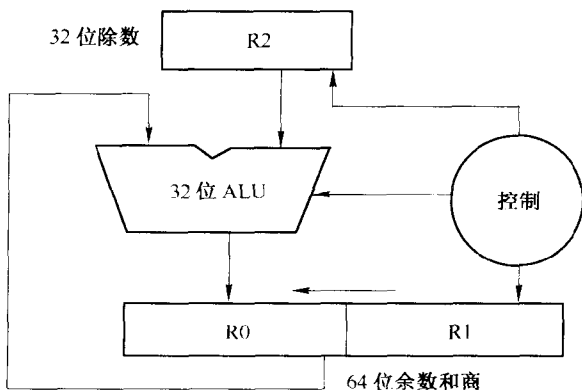
(3) 原码一位除法的运算方法及其算法流程

手工进行二进制除法的规则是判断被除数的大小，若被除数小，则上商 0，并把被除数的下一位移下来或者补 0，再用余数和右移一位的除数比；若被除数大则上商 1 并做减法。

通常每次进行一位的除法操作，因此又称原码一位除法。

(4) 除法器的结构原理

32 位的除法器的构成如下图所示：



除法的计算方法首先是进行移位操作，因为第一次比较的结果总是被除数小并且商 0，否则意味着结果溢出。因此，在流程图中不作第一步比较，步骤一中直接进行移位和商 0 的操作，然后在循环中包含三个步骤：减法、上商和移位。因为余数寄存器的移位同时也对商进行了移位，在循环体中对商移位 32 次意味着对余数多做一次移位。这样最后必须用一个纠正步骤，将余数右移一位以恢复正确结果。

4. 逻辑运算

(1) 逻辑与、逻辑或和逻辑非运算的概念

逻辑非：逻辑非运算对数据的每一位进行求反，将 1 变成 0，0 变成 1。这种运算称为按位求反，或称按位求非。

逻辑加：逻辑加运算就是按位求逻辑或的运算，与二进制加法的一个重要区别就是运算是按位进行的，位与位之间没有进位。

逻辑乘：逻辑乘运算就是按位求逻辑与的运算，运算是按位进行的，位与位之间没有进位。

(2) 各种移位操作的概念与移位结果的计算

移位操作是一种逻辑运算。移位操作分为算术移位、逻辑移位和循环移位三种，每种移位操作又分为左移和右移。

算术左移将寄存器的每一位数据向左移动一个位置。最右位移入 0，最高位的数据移出到保存进位状态位的寄存器的 C 位中。算术左移操作相当于将寄存器中的数据乘以 2 的操作，结果正确的条件是操作中没有发生溢出，即移位过程中符号位的内容没有发生变化。在左移的过程中，符号位的内容被最高数据位代替，如果这两个位的内容不同，移位操作就发生了溢出。

算术右移将寄存器的每一位数据向右移动一个位置。最右位移入 C，最高位移入的数据与原来的值相同。这样可使得移位后数据的符号位不变。如果寄存器中存放的是数据的补码的话，算术右移操作相当于将数据除以 2 的操作。

5. 定点运算器的组成和结构

(1) 算术逻辑功能部件的结构原理

定点数的算术运算部件通常兼有逻辑运算的功能。