



高等学校规划教材

计算机组成原理

罗克露 单立平 刘 辉 俸志刚 编

俸远祯 主审



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

高等学校规划教材

计算机组成原理

罗克露 单立平 刘 辉 俸志刚 编

俸远楨 主审

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书以电子科技大学编写的六·五、八·五、九·五等全国规划教材为基础(其中八·五规划教材获得教育部科技进步奖教材三等奖),重新编写。从面向传统的小型机转到以当前主流微机技术为背景,系统地介绍计算机各功能子系统的逻辑组成和工作机制,更加强调系统级的整机概念。

全书共分七章。第1章概述计算机的基本概念和计算机系统的硬、软组织;第2章分别介绍数据信息和指令信息的表示方法;第3章讨论CPU子系统,将CPU作为集成化整体,通过一个CPU模型,详细分析指令的执行过程;第4章介绍存储子系统,着重讨论主存储器的设计;第5章介绍I/O子系统,包括接口的基本知识,系统总线,以及中断和DMA等I/O传送控制机制;第6章介绍常用输入/输出设备及其I/O接口;第7章给出一个计算机硬件系统模型,作为全书的总结。

本书可以作为高等院校计算机专业学生的“计算机组成原理”课程的教材,也可作为从事计算机专业的工程技术人员的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

计算机组成原理/罗克露等编. —北京:电子工业出版社, 2004.8

高等学校规划教材

ISBN 7-121-00166-7

I. 计… II. 罗… III. 计算机体系结构—高等学校—教材 IV. TP303

中国版本图书馆CIP数据核字(2004)第074757号

责任编辑:明足群 王昌铭

印 刷:北京李史山胶印厂

出版发行:电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路173信箱 邮编 100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:25.00 字数:636千字

印 次:2004年8月第1次印刷

印 数:5000册 定价:30.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。联系电话:(010)68279077。质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

出版说明

信息社会对每一个人,尤其是科技人员、教育工作者和广大学生提出了知识的更新和重构的问题,在知识增值的情况下,对如何获取知识和信息产生了强烈的要求。虽然我们可以从网络、从媒体、从书本上获得信息和知识,但在教育领域,教材和技术参考书仍然是重要的知识来源,它们是加工(处理)后的信息。

计算机科学技术的发展,引起了一场革命,形成了一种文化,创新了一种观念,促进了社会的发展。以微电子为依托,计算机与通信的联姻,带来了全球经济文化的大变革,信息全球化已经初见端倪。为了适应知识的更新和信息的获取,经过高等院校、企业事业单位和应用领域用户的认真讨论,我们规划了这套电子信息类教材和技术参考书。

在现代社会中,电子信息类教材和技术参考书包含了计算机专业类和非计算机专业类、网络与通信专业类、电子技术类,读者层次也覆盖了各个应用领域,这些教材和技术书籍将对我们的信息技术的普及起到极大的推动和促进作用。

在现代教育体系中,教材的作用是不言而喻的,但教材的内容则各有千秋。由于学科发展、专业结构、培养对象的不同,全国统编教材已经不完全适应各个学校。多种风格各异、独具特色的教材如雨后春笋般层出不穷,并深受读者欢迎。一本教材的内容不可能包罗万象,也不可能适应各层次的读者对象。但是,教材一定要有自己的特色。这些特色是通过知识点及其之间的关联性体现出来的,这样,教材的选用者就可以了解该教材的适用范围和对象,就能够很好地规划自己的教学。

信息技术类参考书是对专业教材内容的一个重要补充,也是对专业技术实践环节(实验、实训、实作)的一个实施。这种参考书体系将紧跟国际新技术发展,把最实用的技术及时介绍给读者。

本系列教材及技术参考书采用编委会形式,选题内容由需求引导,编委会讨论,专家(行家)撰写,编委会审阅,最后定稿,以保证书的编写质量。

当我们跨进 21 世纪大门的时候,希望广大的读者能与我们携手并进,共同关心我们的教材和技术参考书的出版,将需求、建议和意见及时反馈给我们,与我们一起开辟一片新的天地。

编委会

前 言

本书以 1985 年出版的《计算机组成原理》(六·五全国规划教材)、1996 年出版的《计算机组成原理》修订本(八·五全国规划教材,曾获得教育部科技进步奖教材类三等奖)、1997 年出版的《计算机组成原理与汇编语言程序设计》(九·五全国规划教材)为基础,重新编写。从面向传统的小型机转到以当前主流微机技术为背景,系统地介绍计算机各功能子系统的逻辑组成和工作机制,更加强调系统级的整机概念。

全书共分七章。按照先建立 CPU 整机概念,再发展为主机,最后形成硬件系统的教学思路来安排各章节的内容。第 1 章概述计算机的基本概念和计算机系统的硬、软组织,强调信息的数字化表示和存储程序的工作方式这两个重要概念;第 2 章分别介绍数据信息和指令信息的表示方法,说明信息表示数字化在计算机中的具体体现;第 3 章讨论 CPU 子系统,将 CPU 作为集成化整体,通过一个 CPU 模型,深入分析指令的执行过程,并详细阐明组合逻辑控制和微程序控制的设计方法;第 4 章介绍存储子系统,着重讨论主存储器的设计,说明主存与 CPU 的连接,并从存储系统的角度介绍外部存储器的存储原理;第 5 章介绍 I/O 子系统,包括接口的基本知识、系统总线,并且强调中断和 DMA 等 I/O 传送控制机制;第 6 章介绍常用输入/输出设备及其 I/O 接口;第 7 章给出一个计算机硬件系统模型,完整而简洁地描述其系统组成和典型 I/O 操作,作为全书的总结。

本书的第 1、3 章由罗克露编写,第 4、5 章由单立平编写,第 6 章由刘辉编写,第 2、7 章由俸志刚编写,全书由罗克露统稿。电子科技大学俸远祯教授担任本书主审,作为前述六·五、八·五、九·五等全国规划教材的主编,俸远祯教授对本书的编写予以积极支持、热情关心和认真指导,他仔细地审阅了书稿,提出许多宝贵意见,在此谨向俸远祯教授表达我们衷心的感谢。在本书编写过程中,还得到了电子科技大学教务处、计算机学院有关领导和老师们的支持与鼓励,以及电子工业出版社王昌铭老师等的热情关怀,谨向他们表示诚挚的谢意。

由于我们水平有限,对书中的错误和不足之处,恳请广大读者批评指正,我们将不胜感激。

编 者
2004 年 7 月

高等学校电子信息类教材编委会名单

主 任：刘乃琦

委 员：

丁志强 马 争 王世普

王昌铭 邓昌延 付 彦

刘乃琦 李志蜀 李玉柏

余 江 吴 跃 吴宗粹

张宽海 张 扬 张翠芳

钱寿宇 梁成华 黄书万

顾 问：

刘锦德 俸远楨

目 录

第 1 章 概论	1
1.1 计算机的基本概念	1
1.1.1 计算机与诺依曼体制	1
1.1.2 信息的数字化表示	2
1.1.3 存储程序工作方式	6
1.2 计算机系统的组织	7
1.2.1 硬件系统	7
1.2.2 软件系统	10
1.2.3 系统组成的层次结构	13
1.2.4 硬件、软件的功能划分与逻辑上的等价	16
1.3 计算机的特点与性能指标	17
1.3.1 数字计算机的特点	17
1.3.2 计算机的主要性能指标	18
练习题	20
第 2 章 计算机中的信息表示	21
2.1 数值型数据的表示方法	21
2.1.1 进位计数制	21
2.1.2 带符号数的表示	28
2.1.3 数的定点表示与浮点表示	33
2.2 字符表示	38
2.2.1 ASCII 码	39
2.2.2 汉字编码简介	39
2.3 指令信息的表示	41
2.3.1 指令格式	41
2.3.2 常见寻址方式	47
2.3.3 指令的功能和类型	63
练习题	72
第 3 章 CPU 子系统	75
3.1 概述	75
3.1.1 CPU 的基本组成	75
3.1.2 时序控制方式	81
3.1.3 控制器分类	84
3.1.4 CPU 与外部的信息交换	86

3.2 算术、逻辑运算部件	93
3.2.1 加法单元	93
3.2.2 并行加法器与进位逻辑结构	94
3.2.3 多功能算术、逻辑运算部件	97
3.2.4 运算器组织	101
3.3 运算方法	103
3.3.1 定点加减运算	104
3.3.2 溢出判断与移位	106
3.3.3 定点乘法运算	109
3.3.4 定点除法运算	117
3.3.5 浮点四则运算	123
3.4 CPU 模型	126
3.4.1 CPU 设计步骤	126
3.4.2 模型机的指令系统	127
3.4.3 模型机的组成与数据通路	131
3.5 组合逻辑控制方式	136
3.5.1 组合逻辑控制器时序系统	136
3.5.2 指令流程与操作时间表	138
3.5.3 微命令的综合与产生	148
3.6 微程序控制方式	149
3.6.1 微程序控制的基本原理	150
3.6.2 微指令的编码方式与微地址的形成方式	151
3.6.3 模型机微指令格式	155
3.6.4 模型机微程序设计	157
3.7 典型 CPU 简介	164
3.7.1 Intel 8086/8088	164
3.7.2 Intel 80386/80486	168
3.7.3 Pentium 微处理器	171
3.7.4 Alpha 微处理器	173
3.7.5 CRAY-1	175
3.7.6 Transputer	176
练习题	178
第 4 章 存储子系统	180
4.1 概述	180
4.1.1 存储系统的层次结构	180
4.1.2 物理存储器与虚拟存储器	184
4.1.3 存储器的分类	184
4.1.4 存储器系统的关键特性	187
4.2 半导体存储原理及存储芯片	188

4.2.1	双极型存储单元与芯片	188
4.2.2	静态 MOS 存储单元与芯片	191
4.2.3	动态 MOS 存储单元与芯片	195
4.2.4	半导体只读存储器与芯片	201
4.3	主存储器的组织	205
4.3.1	主存储器设计的一般原则	205
4.3.2	主存储器的逻辑设计	207
4.3.3	主存储器与 CPU、系统总线的连接	209
4.3.4	主存储器芯片技术	211
4.3.5	动态存储器的刷新	214
4.3.6	主存储器的校验	215
4.4	磁表面存储器的存储原理	217
4.4.1	记录介质与磁头	217
4.4.2	读写原理	219
4.4.3	磁记录方式	220
4.4.4	磁表面存储器的校验	224
4.5	光存储原理	228
4.5.1	光存储部件	229
4.5.2	光存储基本原理	230
4.6	提高存储系统性能的一些措施	231
4.6.1	高速缓冲存储器	232
4.6.2	虚拟存储器	235
4.6.3	双端口存储器	239
4.6.4	并行存储器	240
4.6.5	相联存储器	244
	练习题	245
第 5 章	输入/输出系统	247
5.1	概述	247
5.1.1	主机与外部设备的连接模式	248
5.1.2	总线类型与总线标准	250
5.1.3	接口功能与接口分类	253
5.2	系统总线	255
5.2.1	总线设计的要素	255
5.2.2	系统总线的信号组成	260
5.3	直接程序传送方式与接口	262
5.4	中断方式与接口	264
5.4.1	中断方式基本概念	264
5.4.2	中断请求	268
5.4.3	中断判优	270

5.4.4	中断响应	274
5.4.5	中断处理	277
5.4.6	中断接口组成模型	280
5.4.7	中断接口举例	283
5.5	DMA 方式与接口	287
5.5.1	DMA 方式基本概念	287
5.5.2	DMA 控制器与接口的连接	290
5.5.3	DMA 控制器的组成	294
5.5.4	DMA 传送操作过程	298
5.5.5	DMA 接口举例	300
	练习题	306
第 6 章	输入/输出设备及 I/O 接口	308
6.1	概述	308
6.1.1	输入/输出设备的一般功能	308
6.1.2	输入/输出设备的类型	309
6.1.3	输入/输出设备与主机系统间的信息交换	312
6.2	键盘及接口	313
6.2.1	键的类型	314
6.2.2	硬件扫描键盘	316
6.2.3	软件扫描键盘及接口电路	317
6.3	显示设备及接口	320
6.3.1	概述	320
6.3.2	显示技术中的相关术语	321
6.3.3	显示方式与显示规格	323
6.3.4	光栅扫描成像原理	324
6.3.5	屏幕显示与显示缓存间的对应关系	328
6.3.6	显示适配器	334
6.4	打印设备及接口	336
6.4.1	概述	336
6.4.2	打印机的相关性能指标	337
6.4.3	点阵针式打印机	338
6.4.4	激光打印机	343
6.4.5	喷墨打印机	346
6.4.6	打印机适配器	348
6.5	磁盘存储器及接口	349
6.5.1	软盘存储器	350
6.5.2	硬盘存储器	354
6.5.3	磁盘适配器	362
6.6	光盘存储器	365

6.6.1	概述	365
6.6.2	光盘存储器的读/写原理	366
6.6.3	光盘存储器的特性	367
6.6.4	光盘驱动器	367
6.7	其他常用的输入/输出设备	369
6.7.1	光学字符识别设备	369
6.7.2	图形图像输入设备	370
6.7.3	语音识别设备	371
6.7.4	条形码识别仪	372
	练习题	373
第 7 章	计算机硬件系统模型	374
7.1	模型机系统及其信号互连	374
7.1.1	系统组成	374
7.1.2	系统总线	375
7.1.3	各部件的信号线	376
7.2	模型机典型 I/O 操作举例	380
7.2.1	直接程序控制方式的 I/O 操作	380
7.2.2	程序中断方式下的 I/O 操作	381
7.2.3	DMA 方式下的 I/O 操作	382
7.3	系统配置举例	383
	练习题	386
	参考文献	387

第1章 概 论

“计算机组成原理”这门课程的主要内容是以单机系统为对象，阐述计算机系统的硬件组成，其核心是建立计算机系统的整机概念。这里的整机概念包括两个方面，即计算机系统的逻辑组成与工作机制。本书将从 CPU 级和硬件系统级这两个层次逐步建立整机概念。为此，在概论这一章首先阐明 3 个重要的基本概念：信息的数字化表示，存储程序工作方式和计算机系统的层次结构；将以上概念作为了解计算机的逻辑组成与工作机制的基本出发点。

1.1 计算机的基本概念

1.1.1 计算机与诺依曼体制

计算机是 20 世纪人类最伟大的发明之一。它通过自动、连续地运行程序，能够代替人类完成各种复杂的计算和实现对各类信息的处理。

一个计算机系统由硬件和软件两大部分组成。硬件是指看得见、摸得着的设备实体，包括运算器、控制器、存储器、输入设备、输出设备等，如图 1-1 所示。软件则不能直接触摸，比如程序、文档等。构造硬件的基本思想是处理功能逻辑化，即用逻辑电路构造各种功能部件，如用门电路、触发器来构造运算器、控制器、存储器等。在硬件基础上，可以根据需要配置各种软件，如操作系统、编程语言、各种支撑软件等。硬件与软件按层次结构组成复杂的计算机系统。

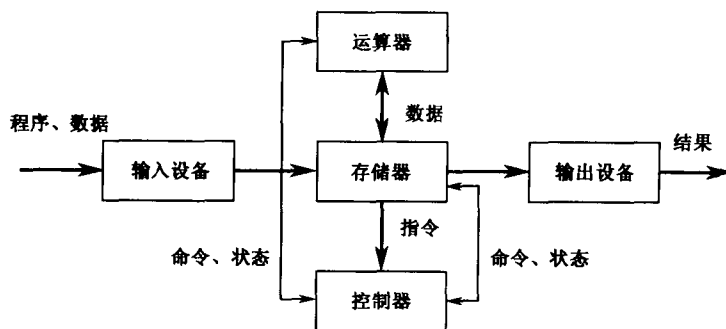


图 1-1 计算机组成示意图

一个计算机系统是如何工作的呢？不管做一次复杂的数学计算，还是对大量的数据进行查询，或者对一个过程实现自动控制，用户都必须按照处理的步骤，用编程语言事先编写程序，然后通过输入设备（如键盘）将程序和需要处理的数据送入计算机，存放在存储器中。用户编写的程序称为源程序，是不能被计算机直接执行的。计算机只能执行机器指令，即要求计算机完成某种操作的命令，简称指令，如执行加法操作的加法指令、执行乘法操作的乘法指令、执行传送操作的传送指令，等等。因此，计算机在运行程序之前，必须将源程序转换为指令序列，并将这些指令按一定顺序存放在存储器的若干单元中。每个单元都有一个固

定的编号，称为地址。只要给出某个地址，就能访问相应的存储单元，对该单元的内容进行读/写操作。

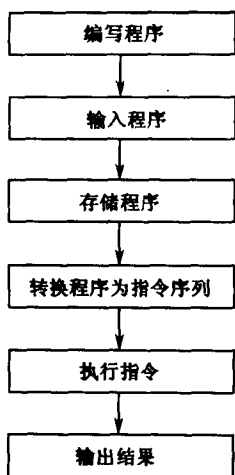


图 1-2 计算机工作流程

当计算机启动运行后，控制器将某个地址送往存储器，从该地址单元取回一条指令。控制器根据这条指令的含义，发出相应的操作命令，控制该指令的执行。比如执行一条加法指令，先要从存储单元或寄存器中取出操作数，送入运算器，再将两个操作数相加，并将运算处理的结果送回存储单元或寄存器存放。如果用户需要了解处理结果，则计算机通过输出设备（如显示器、打印机等），将结果显示在屏幕上，或打印在纸上。图 1-2 给出了计算机的简单工作流程。

从以上的描述可以看出，计算机作为一个处理信息的工具，首先需要解决两个最基本的问题：第一，信息如何表示，才能被计算机识别；第二，采用什么工作方式，才能使计算机自动地对信息进行处理。对这两个问题的解决做出杰出贡献，并且产生深远影响的是一位美籍匈牙利数学家冯·诺依曼。他在 1945 年提出的 EDVAC 计算机设计方案中，总结了计算机的设计思想，被称为诺依曼思想，采用这一思想体制的计算机就称为诺依曼机。几十年来，尽管计算机的体系结构发生了许多演变，但是诺依曼体制的核心思想仍然是普遍采用的结构原则，现在绝大多数实用的计算机仍属于诺依曼计算机。

诺依曼体制的主要思想包括：

- 采用二进制代码形式表示信息（数据、指令）；
- 采用存储程序工作方式（这是诺依曼思想最核心的概念）；
- 计算机硬件系统由五大部件（存储器、运算器、控制器、输入设备和输出设备）组成。

这些思想奠定了现代计算机的基本结构，并且开创了程序设计的新时代。当然，从本质上讲，传统的诺依曼机采用串行处理的工作机制，即逐条执行指令序列。要想提高计算机的性能，其根本方向之一是采取并行处理机制，例如用多个处理部件形成流水线，依靠时间上的重叠来提高处理效率；或者用多个诺依曼机组成多机系统，以支持并行算法，等等。

学习计算机工作原理应该从了解诺依曼思想入门，因此本节先讨论信息的数字化表示和存储程序工作方式这两个要点，下一节再讨论硬件系统的组成。

1.1.2 信息的数字化表示

计算机是通过执行程序（指令序列）来实现对数据的加工处理的。因此，计算机中的信息可以分为两大类：控制信息和数据信息。控制信息用来控制计算机的工作。计算机执行指令时，用指令产生的控制命令（称为微命令）控制有关操作，所以指令序列和微命令序列属于控制信息类。数据信息是计算机加工处理的对象。计算机根据指令要求取出的操作数，以及对操作数进行处理的结果，等等，都属于数据信息类。数据信息又分为数值型数据和非数值型数据两类。数值型数据有数值大小与正负之分，如 6, -15 等。非数值型数据则无数值大小，也不分正数负数，如字符、文字、图像、声音等人们能够识别的信息，以及条件、状态、命令等用于判定的逻辑信息。那么，在计算机中如何表示这些信息呢？

计算机的主要部件是用逻辑电路，即电子电路构成的，所以，在电子数字计算机中传送与处理的信息都采用数字化表示方法。信息的数字化表示包含了两层含义：

- 用数字代码表示各种信息；
- 用数字信号表示数字代码。

信息表示数字化这个重要概念是理解计算机工作原理的一个基本出发点。下面通过多个例子说明这两层含义。

1. 在计算机中用数字代码表示各种信息

数字代码是指一组数字的集合，这里的数字代码通常指二进制数字代码。可以根据需要描述的信息（某类控制信息或某类数据信息），用一组约定含义的数字代码来表示它。

【例 1-1】 用数字代码表示数值型数据。

6 和 -7 是两个数值型数据。可以约定，用一位二进制代码表示每个数的符号，比如用 0 表示正数，用 1 表示负数；再用 4 位二进制代码表示每个数的大小。这样，代码 00110 表示 6（左边第一个 0 代表正号），代码 10111 表示 -7（左边第一个 1 代表负号）。

当然，也可以用 8 位二进制代码表示一个数的大小。代码位数增多，则数的表示范围将扩大。例如，用 4 位二进制代码不能表示 200 这个数，但用 8 位二进制代码则可以表示它。

【例 1-2】 用数字代码表示字符。

字符本身没有大小和正负之分，但仍然可用数字代码来表示它。计算机中常常约定用 7 位代码表示一个西文字符，如用 1000001 表示字符 A，用 1000010 表示字符 B；或用 7 位代码表示一个控制字符，如用 0001100 表示换页（FF），用 0001101 表示回车（CR）。字符的这种编码称为 ASCII 码，是国际上广泛采用的一种字符表示方法。另外，还可约定用两组 8 位二进制代码表示一个中文字符，如用 01010110 01010000 表示“中”，用 00111001 01111010 表示“国”，等等。总之，用数字代码可以表示各种字符，而以字符为基础又可以表示范围广泛的各种文字。

【例 1-3】 用数字代码表示图像。

字符的种类总是有限的，因而可以用若干位编码来表示；图像则不然，其变化是无穷无尽的，那么如何用数字代码来表示这些随机分布的图像信息呢？实际上，一幅图像可以被细分为若干个点，这些点称为像点。也就是说，可用像点的组合来逼近真实的图像。图像划分得越细，像点越多，组成的图像也就越真实。按照信息表示数字化的思想，可用数字代码表示像点。例如用一位代码表示一个像点，若像点是亮的，则用代码 1 表示；若像点是暗的，则用代码 0 表示。再将表示一幅图像所有像点的代码按照像点在图像中的位置进行组织，就可以实现用数字代码来表示图像了。

【例 1-4】 用数字代码表示声音。

为了对声音信息数字化，首先要将声波转换为电波，再按一定频率对电波进行采样，即在长度相同的时间间隔内分别对电流波的幅值进行测量，每次测到的电流幅值都用一个数字量来表示。只要采样频率足够高，所得到的数字信息就能逼真地保持声波信息，还原后真实地再现原来的声音。

【例 1-5】 用数字代码表示指令。

指令属于控制信息。通常，一条指令需要提供要求计算机做什么操作，以及如何获取操作数等信息。因此，可用一段数字代码表示操作类型，这段代码称为操作码；用另一段代码

表示获取操作数的途径，这段代码称为地址码。将操作码和地址码组合在一起，就形成机器指令代码。例如，操作码取 4 位，可以约定用 0000 表示传送操作，0001 表示加法操作，0010 表示减法操作……；地址码取 12 位，用 6 位表示一个操作数的来源，用其余 6 位表示另一个操作数的来源。如约定用 000000 表示操作数来自 0 号寄存器，000001 表示另一操作数来自 1 号寄存器。这样，16 位代码 0001000000000001 表示一条加法指令，它的含义是将 0 号寄存器的内容与 1 号寄存器的内容相加，结果存放在 1 号寄存器中。

【例 1-6】 用数字代码表示设备状态。

计算机在工作时往往需要了解外部设备的状态，根据外设状态决定做什么操作。不同的外部设备可能有不同的工作状态，例如打印机将字符打印在纸上，而显示器则将字符显示在屏幕上。我们可以将这些设备的状态抽象、归纳为 3 种：空闲（设备没有工作）、忙（设备正在工作）、完成（设备做完一次操作）。相应地，可用约定的数字代码表示这 3 种状态，比如，用 00 表示空闲，01 表示忙，10 表示完成。

2. 在物理机制上用数字信号表示数字代码

为什么要用数字代码表示各种信息呢？这就涉及计算机的物理机制了。计算机是一种复杂的电子线路，它传送和处理的实际对象是电信号。电信号又分为模拟信号和数字信号两种类型。

模拟信号是一种随时间连续变化的电信号，如电流信号、电压信号等。我们可用电流或电压的幅值来模拟数值或物理量的大小，比如模拟温度的高低、压力的大小等。处理模拟信号的计算机称为模拟计算机，只应用在极特殊的领域中。用模拟信号表示数据的大小有许多缺点，如表示的精度低、表示的范围小、抗干扰能力弱、不便于存储等。用数字信号表示信息则可以克服以上缺点。

数字信号是一种在时间上或空间上断续变化的电信号，如电平信号和脉冲信号。单个数字信号一般只取两种状态，例如电平的高或低、脉冲的有或无，这样，就可用这两种状态分别表示数字代码 1 和 0，称为二值逻辑。比如，用高电平状态表示 1，低电平状态表示 0；或者用有脉冲的状态表示 1，无脉冲的状态表示 0。用一位数字信号表示一位数字代码，用多位

数字信号的组合就可表示多位数字代码。处理数字信号的计算机称为数字计算机，电平信号和脉冲信号是数字计算机中最基本的信号形式。由于可用数字信号表示数字代码，用数字代码又可表示各种信息，因而数字计算机能够用于各行各业，处理广泛的信息。下面通过两个例子说明如何用多位数字信号的组合来表示多位数字代码。

【例 1-7】 用一组电平信号表示四位数字代码。

电平信号利用信号电平的高、低状态表示不同的代码，所以电平信号通常需要一段有效维持时间。可用 4 根信号线分别输出 4 个电平信号，每个电平信号表示一位代码。我们约定 +5V 为高电平，表示 1；0V 为低电平，表示 0。如图 1-3 所示，4 位电平信号表示 4 位数字代码 1011，它们可能表示一个 4 位的二进制数，也可能表示一个命令或一种状态的编码。

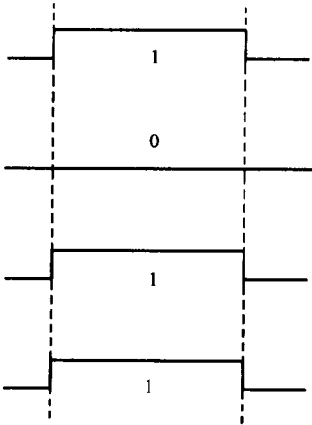


图 1-3 用一组电平信号表示多位数字代码

每位信号各占用一根信号线,因而这组电平信号在空间上的分布是离散的。在计算机中常用电平信号表示并行传送的信息,如用若干根信号线同时传送的数据、地址或其他信息的编码。

【例 1-8】用一串脉冲信号表示 4 位数字代码。

与电平信号不同,脉冲信号的电平维持时间很短,例如,信号电平从 0V 向 +5V (或 -5V) 跳变,维持极短时间后再回到原来的 0V 状态。因此,信号出现时其电平为 +5V (或 -5V),信号未出现时其电平为 0V,如图 1-4 所示。由于脉冲信号在时

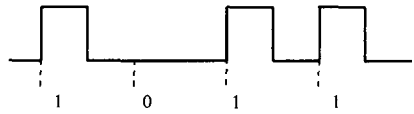


图 1-4 用一串脉冲信号表示多位数字代码

间上的分布是离散的,因而可用一根信号线发出一串脉冲信号,在约定的时间内有脉冲表示 1,无脉冲表示 0。图 1-4 中的脉冲串表示 4 位数字代码 1011。

可以使用脉冲信号的上升边沿或下降边沿表示某一时刻,对某些操作定时。例如,在脉冲上升边沿将数据送入某个寄存器中。另外,在计算机中常用脉冲信号表示串行传送的数据。

从事计算机技术工作的重要基础,是善于使用约定的数字代码表示各种需要描述的信息。这里,再次强调信息数字化这个重要概念:

- 计算机中的各种信息用数字代码表示,这些信息包括数值型的数字,非数值型的字符、图像、声音,逻辑型的命令、状态,等等;
- 数字代码中的每位用数字信号表示,数字信号可以是电平信号或脉冲信号。

3. 用数字化方法表示信息的主要优点

(1) 在物理上容易实现信息的表示与存储

每位信号只取两种可能的状态表示 1 或 0,因而在物理上可用多种方法来实现,如开关的接通或断开、晶体管的导通或截止、电容上有电荷或无电荷、磁性材料的正向磁化或反向磁化、磁化状态的变化或不变,等等。凡是具有两种稳定状态的物理介质均可用来存储信息,比如,用双稳态触发器存储信息,或用电容上存储的电荷来存储信息,还可以用磁性材料记录信息,或者用激光照射过的介质记录信息。

(2) 抗干扰能力强,可靠性高

由于单个数字信号的两种状态(高电平与低电平,或者有脉冲与无脉冲)差别较大,即使信号受到一定程度的干扰,仍然能比较可靠地鉴别电平的高低或信号的有无。例如高电平 +5V 表示 1,低电平 0V 表示 0,假设信号处于 0 状态,如果出现了 2V 的干扰信号,也不会将原来信号的 0 状态改变到 1 状态。

(3) 数值的表示范围大,表示精度高

一位数字信号的表示范围很窄,但用多位数字信号的组合表示一个数时,可以获得很大的表示范围和很高的精度。例如,用 4 位电平信号表示一个 4 位的二进制整数时(不考虑符号),能够表示的最大数值是 15。若表示一个 4 位的二进制小数,同样不考虑符号,则数的精度为 2^{-4} 。位数越多,数的表示范围越大,或者数的表示精度越高。从理论上讲,位数的增加是有限制的,但位数增多,所花费的硬件开销也相应增大。

(4) 表示的信息类型极其广泛

各种非电量类型的信息可以先转换为电信号,模拟电信号又可以转换为数字电信号,因此表示的信息类型和范围几乎没有限制。

(5) 能用数字逻辑技术进行信息处理

根据处理功能逻辑化的思想,计算机的所有操作最终是用数字逻辑电路来实现的。因此,用逻辑代数对信息进行处理,就形成了计算机硬件设计的基础,可以用非常有限的几种逻辑单元(如与门、或门、非门等)构造出变化无穷的计算机系统和其他数字系统。

1.1.3 存储程序工作方式

存储程序是诺依曼思想的核心内容,它表明了计算机的工作方式,包含以下3个要点:

- 事先编制程序;
- 事先存储程序;
- 自动、连续地执行程序。

这3点体现了用计算机求解问题的过程,下面分别加以说明。

1. 根据求解问题事先编制程序

计算机处理任何复杂的问题都是通过执行程序来实现的。因此,在求解某个问题时,用户需要根据解决这个问题所采用的算法事先编制程序,规定计算机需要做哪些事情,按什么步骤去做。程序中还应提供需要处理的数据,或者规定计算机在什么时候、什么情况下从输入设备取得数据,或向输出设备输出数据。

2. 事先将程序存入计算机中

如前所述,用户用某种编程语言编写的程序称为源程序,它是由字符组成的,计算机不能识别。因此,需要通过输入设备将源程序转换为二进制代码,送入计算机的存储器中。这时的程序还不是指令代码,不能被计算机执行,还需进一步转换为符合某种格式的机器指令序列。所以,事先编写的程序最终将变为指令序列和原始数据,且被保存在存储器中,提供给计算机执行。

3. 计算机自动、连续地执行程序

程序已经存储在计算机内部。计算机被启动后,不需要人工干预,就能自动、连续地从存储器中逐条读取指令,按指令要求完成相应操作,直到整个程序执行完毕。当然,在某些采用人机对话方式工作的场合,也允许用户以外部请求方式干预程序的运行。

指令和数据都以二进制代码的形式存放在存储器中,那么计算机如何区分它们?又如何自动地从存储器中读取指令呢?首先,将指令和数据分开存放。由于多数情况下程序是顺序执行的,因此大多数指令需要依次紧靠着存放,而将数据放在该程序区中不同的区间。其次,可以设置一个程序计数器(Program Counter),称为PC,用它存放当前指令所在的存储单元的地址。如果程序顺序执行,则在读取当前指令后将PC的内容加1(当前指令只占用一个存储单元),指示下一条指令的地址。如果程序需要转移,则将转移目标地址送入PC,以便按照转移地址读取后续指令。所以,依靠PC的指示,计算机就能自动地从存储器中读取指令,再根据指令提供的操作数地址读取数据。

对于传统的诺依曼机而言,存储程序工作方式是一种控制流驱动方式,即按照指令的执行序列依次读取指令,再根据指令所含的控制信息调用数据进行处理。这里的控制流也称指令流,是指在程序执行过程中,各条指令逐步发出的控制信息,它们始终驱动计算机工作。而依次被处理的数据信息称为数据流,它们是被驱动的对象。