

 高等教育规划教材

微型计算机

原理及应用

赵全利 主编

免费提供电子教案和习题解答



下载网址 <http://www.cmpedu.com>



机械工业出版社
CHINA MACHINE PRESS

高等教育规划教材

微型计算机原理及应用

赵金利 主编

张会敏 周 伟 副主编



机械工业出版社

本书综合了微处理器结构原理、汇编语言程序设计及接口应用技术 3 部分内容。首先,在概述计算机基础知识及微型计算机系统组成之后,以 8086 CPU 为基础,详细介绍 80x86 微处理器的硬件结构、工作原理、指令系统、汇编语言程序设计及 EMU8086 仿真软件应用示例。然后,对微型计算机存储器系统及接口设计、输入/输出接口、总线及中断应用技术做了详细的描述。最后,以常用集成可编程芯片为对象,以应用技术为核心,重点介绍了串行通信、并行通信、定时器/计数器、D-A 及 A-D 转换及接口电路软硬件设计,并通过 Proteus 电路原理设计对应用实例进行了仿真调试。

本书通俗易懂、构思清晰,应用与虚拟仿真并行,通过大量的实例及仿真环境调试,引导读者逐步认识、熟知、掌握微型计算机应用技术。

本书既可作为高等院校大学本专科电子、通信、计算机、自动控制、测控、机电及机械等专业的教学用书,也可作为相关技术人员的参考用书。

本书配有授课电子教案,需要的教师可登录 www.cmpedu.com 免费注册,审核通过后下载,或联系编辑索取(QQ: 2850823885, 电话: 010-88379739)。

图书在版编目(CIP)数据

微型计算机原理及应用/赵全利主编. —北京:机械工业出版社,2016.4
高等教育规划教材
ISBN 978-7-111-53315-3

I. ①微… II. ①赵… III. ①微型计算机—高等学校—教学参考资料
IV. ①TP36

中国版本图书馆 CIP 数据核字(2016)第 061474 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:和庆娣 责任编辑:和庆娣 张 恒

责任校对:张艳霞 责任印制:常天培

北京中兴印刷有限公司印刷

2016 年 6 月第 1 版·第 1 次印刷

184mm×260mm·21.75 印张·538 千字

0001—3000 册

标准书号:ISBN 978-7-111-53315-3

定价:49.90 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

服务咨询热线:(010)88379833

读者购书热线:(010)88379649

封面无防伪标均为盗版

网络服务

机工官网:www.cmpbook.com

机工官博:weibo.com/cmp1952

教育服务网:www.cmpedu.com

金书网:www.golden-book.com

出版说明

当前,我国正处在加快转变经济发展方式、推动产业转型升级的关键时期。为经济转型升级提供高层次人才,是高等院校最重要的历史使命和战略任务之一。高等教育要培养基础性、学术型人才,但更重要的是加大力度培养多规格、多样化的应用型、复合型人才。

为顺应高等教育迅猛发展的趋势,配合高等院校的教学改革,满足高质量高校教材的迫切需求,机械工业出版社邀请了全国多所高等院校的专家、一线教师及教务部门,通过充分的调研和讨论,针对相关课程的特点,总结教学中的实践经验,组织出版了这套“高等教育规划教材”。

本套教材具有以下特点:

1) 符合高等院校各专业人才的培养目标及课程体系的设置,注重培养学生的应用能力,加大案例篇幅或实训内容,强调知识、能力与素质的综合训练。

2) 针对多数学生的学习特点,采用通俗易懂的方法讲解知识,逻辑性强、层次分明、叙述准确而精炼、图文并茂,使学生可以快速掌握,学以致用。

3) 凝结一线骨干教师的课程改革和教学研究成果,融合先进的教学理念,在教学内容和方法上做出创新。

4) 为了体现建设“立体化”精品教材的宗旨,本套教材为主干课程配备了电子教案、学习与上机指导、习题解答、源代码或源程序、教学大纲、课程设计和毕业设计指导等资源。

5) 注重教材的实用性、通用性,适合各类高等院校、高等职业学校及相关院校的教学,也可作为各类培训班教材和自学用书。

欢迎教育界的专家和老师提出宝贵的意见和建议。衷心感谢广大教育工作者和读者的支持与帮助!

机械工业出版社

前 言

目前,微型计算机的应用已广泛深入到科学计算、信息处理、事务管理、系统控制、仪器仪表及人类学习、生活的各个领域,微型计算机应用技术已成为电子信息技术产业的核心。

微机原理、汇编语言程序设计及接口技术是微型计算机应用技术不可分割的三大核心内容,是高等院校计算机科学与技术、电子信息工程、通信工程、电气工程、机电工程、测控及自动化等专业的必修核心课程。

本书以微型计算机基本原理为基础,以应用为主要目的,内容包括:第1章介绍计算机基础知识及微型计算机系统组成;第2章以8086 CPU为基础,介绍80x86微处理器的硬件结构及工作原理;第3、4章介绍80x86 CPU的汇编指令系统、汇编语言、汇编语言程序设计及EMU8086仿真软件的应用;第5章介绍存储器原理、层次结构、与CPU接口及存储系统设计;第6章介绍输入/输出接口、DMAC控制器及总线技术;第7章介绍中断概念、80x86中断系统、中断控制器芯片及应用技术;第8、9章以常用集成电路可编程芯片为对象,详细介绍串行通信,并行通信,定时器/计数器,D-A和A-D转换的基本原理、性能及接口应用技术。

本书以丰富的应用实例,将主要知识点贯穿其中,对程序示例引入EMU8086仿真运行,对接口电路软、硬件示例通过Proteus仿真调试运行,以方便读者深入地理解微型计算机工作过程,引导读者逐步认识、熟悉、掌握微型计算机应用技术。

本书概念清楚,且注重知识的内在联系与规律,采用归纳、类比的方法,使读者通过本书的学习掌握微型计算机的结构原理、汇编语言程序设计及接口应用系统的组成与设计方法,并能够解决微型计算机在自身设置、工业控制、电子技术系统开发等方面的一些实际问题。为了便于读者理解、掌握本书的内容,提高应用技能,每章均配有大量的例题与习题。

本书内容翔实、结构完整、概念清晰、实例丰富,既便于循序渐进地进行教学,又可按需对内容进行筛选;既便于学生自学,又很大程度上减少了教材内容的冗余度。

本书由赵全利主编,副主编为张会敏、周伟。赵全利编写第1、2、3、8章,罗婕编写第4章,周伟编写第5、9章,张会敏编写第6章,谢泽奇编写第7章,参与编写的还有赵众、刘瑞新、田金雨、骆秋容、王如雪、曹媚珠、陈文焕、刘有荣、李刚、孙明建、李索、刘大学、刘克纯、沙世雁、缪丽丽、田金凤、陈文娟、李继臣、王如新、赵艳波、王茹霞、田同福、徐维维、徐云林。全书由赵全利教授统稿,刘瑞新教授审校。

由于编者水平有限,书中难免存在疏漏和不足之处,敬请广大读者批评指正。

编 者

目 录

出版说明

前言

第 1 章 计算机基础知识	1	2.2.1 时钟周期、总线周期和指令周期	43
1.1 计算机的产生、发展及应用特点	1	2.2.2 基本的总线时序	44
1.1.1 计算机产生的结构思想	1	2.3 从 8086 到 80x86 微处理器技术的发展	47
1.1.2 计算机的发展过程	2	2.3.1 80286 和 80386	47
1.1.3 计算机的特点及应用	3	2.3.2 80486 CPU	48
1.2 计算机中数据的表示及电路	4	2.4 Pentium (奔腾) CPU	52
1.2.1 计算机使用的数制及转换	4	2.4.1 Pentium (奔腾) CPU 概述	53
1.2.2 二进制运算及电路实现	9	2.4.2 Pentium 4 简介	55
1.2.3 二进制数编码	14	2.4.3 新一代微处理器——Itanium (安腾) CPU 简介	56
1.2.4 数的定点和浮点表示	18	2.5 习题	57
1.3 微型计算机的发展、特点及分类	20	第 3 章 微型计算机指令系统	59
1.3.1 微型计算机的发展及特点	20	3.1 指令系统简介	59
1.3.2 微型计算机分类	21	3.1.1 指令及指令系统	59
1.4 微型计算机常用术语及性能指标	22	3.1.2 指令格式	59
1.4.1 常用术语	22	3.2 8086 指令的寻址方式	60
1.4.2 性能指标	23	3.2.1 操作数及分类	60
1.5 微型计算机系统组成	24	3.2.2 8086 数据寻址方式	61
1.5.1 硬件组成	24	3.2.3 8086 转移地址寻址方式	68
1.5.2 软件系统	26	3.3 8086 指令系统	69
1.6 微型计算机工作过程	28	3.3.1 数据传送指令	70
1.7 习题	29	3.3.2 算术运算指令	75
第 2 章 微处理器及其体系结构	32	3.3.3 逻辑运算及移位指令	81
2.1 8086 微处理器	32	3.3.4 串操作类指令	86
2.1.1 8086 微处理器的内部结构和功能	32	3.3.5 控制转移类指令	93
2.1.2 8086 微处理器的寄存器组	33	3.3.6 处理器控制指令	100
2.1.3 8086 微处理器的引脚分布与工作模式	36	3.4 从 8086 到 80x86 指令系统的变化	101
2.1.4 8086 微处理器对存储器的管理	39	3.4.1 80x86 系统寻址方式	101
2.1.5 8086 微处理器的工作过程	42	3.4.2 80x86 增强和扩展指令	103
2.2 8086 微处理器的总线周期和操作时序	43	3.5 汇编指令 EMU8086 仿真调试	105

3.6 习题	107	5.1.2 存储器分类及特点	146
第4章 80x86 汇编语言及程序设计	112	5.1.3 存储器的主要性能参数	148
4.1 汇编语言语法基本知识	112	5.2 读写存储器 (RAM)	149
4.1.1 汇编语言和汇编程序	112	5.2.1 静态 RAM (SRAM)	149
4.1.2 汇编语言语句	113	5.2.2 动态 RAM (DRAM)	150
4.1.3 汇编语言的数据和表达式	114	5.2.3 RAM 的工作时序	152
4.1.4 汇编语言源程序的结构	117	5.3 只读存储器 (ROM)	153
4.2 常用汇编伪指令	118	5.3.1 只读存储器的结构	153
4.2.1 符号定义伪指令	118	5.3.2 只读存储器 EPROM	154
4.2.2 数据定义伪指令	119	5.3.3 只读存储器 EEPROM	155
4.2.3 程序分段定义伪指令	120	5.3.4 快闪存储器 Flash	156
4.2.4 定位操作伪指令	121	5.4 存储器系统设计	156
4.2.5 程序模块的定义和通信	121	5.4.1 确定存储器结构	156
4.2.6 宏操作伪指令	122	5.4.2 存储器地址分配及译码	158
4.3 80x86 宏汇编伪指令增强与 扩充	122	5.4.3 存储器与微处理器的接口连接	160
4.4 汇编语言程序设计的基本方法	124	5.4.4 简单存储器子系统的设计	161
4.4.1 程序设计步骤及技术	124	5.5 80x86 存储系统简介	164
4.4.2 顺序程序设计	125	5.5.1 并行存储器	164
4.4.3 选择程序设计	126	5.5.2 高速缓冲存储器	165
4.4.4 循环程序设计	127	5.6 习题	167
4.5 子程序设计	130	第6章 输入/输出及总线技术	169
4.5.1 过程定义伪指令	130	6.1 输入/输出接口	169
4.5.2 子程序调用与返回指令	131	6.1.1 输入/输出接口基本结构及 工作过程	169
4.5.3 子程序设计举例	132	6.1.2 输入/输出编址及电路实现	171
4.6 汇编语言程序上机过程	133	6.2 微处理器与外设之间数据 控制方式	175
4.6.1 上机步骤	133	6.2.1 无条件传送方式	175
4.6.2 调试工具 DEBUG	134	6.2.2 查询传送方式	176
4.7 常用汇编语言仿真软件及程序 调试示例	134	6.2.3 中断传送方式	177
4.7.1 EMU8086 汇编语言仿真软件 简介	134	6.2.4 DMA 控制方式	177
4.7.2 Masm for Windows 集成实验 环境简介	137	6.3 可编程序 DMA 控制器	179
4.7.3 使用 EMU8086 汇编语言仿真软件 调试示例	138	6.3.1 8237 功能及内部结构	179
4.8 习题	142	6.3.2 8237 工作方式	180
第5章 存储器及应用技术	145	6.3.3 8237 芯片引脚功能	180
5.1 存储器概述	145	6.3.4 内部计数器及寄存器组	182
5.1.1 主存储器结构及存储系统的层次 结构	145	6.3.5 DMA 应用编程	184
		6.4 总线技术简介	185
		6.4.1 标准总线及分类	185
		6.4.2 微机系统总线	187

6.4.3	总线驱动	191	8.3.1	8253 的性能、结构及引脚功能	274
6.4.4	串行通信总线标准	193	8.3.2	8253 控制字及工作方式	278
6.4.5	串行通信传输通道配置	196	8.3.3	8253 编程	288
6.5	习题	198	8.3.4	8253 应用实例	290
第 7 章	中断及应用技术	200	8.3.5	8253A 定时/计数器 Proteus 仿真 示例	291
7.1	中断系统	200	8.4	习题	295
7.1.1	中断概述	200	第 9 章	数-模/模-数转换及应用技术	298
7.1.2	80x86 中断系统	204	9.1	计算机闭环控制系统	298
7.2	8259A 可编程序中断控制器	208	9.1.1	模拟信号获取及变换	298
7.2.1	8259A 中断控制器逻辑功能	208	9.1.2	计算机闭环控制系统	300
7.2.2	8259A 内部结构及引脚功能	208	9.1.3	一个典型实时控制系统	301
7.2.3	8259A 的工作过程	210	9.2	D-A 转换器	302
7.2.4	8259A 编程	210	9.2.1	D-A 转换器的基本原理	302
7.2.5	8259A 应用实例	215	9.2.2	D-A 转换器的主要参数	303
7.3	中断应用	217	9.2.3	8 位集成 D-A 转换器—— DAC0832	303
7.3.1	ROM BIOS 中断调用	217	9.2.4	DAC0832 应用接口及编程	305
7.3.2	DOS 系统功能调用	218	9.3	A-D 转换器	308
7.3.3	中断程序设计	219	9.3.1	A-D 转换器的基本原理	308
7.3.4	基于 Proteus 软件的 8259A 接口 电路仿真调试	221	9.3.2	A-D 转换器的主要技术指标	310
7.4	习题	225	9.3.3	A-D 转换器的外部特性	310
第 8 章	常用可编程序接口芯片及 应用技术	227	9.3.4	集成 8 位 A-D 转换器—— ADC0809	310
8.1	串行通信及可编程序接口芯片 应用技术	227	9.3.5	ADC0809 应用接口及编程	313
8.1.1	串行通信	227	9.4	D-A 及 A-D 应用 Proteus 仿真示例	317
8.1.2	可编程序接口芯片 8251A	230	9.4.1	DAC0832 接口电路软硬件 仿真示例	317
8.1.3	8251A 控制字及初始化编程	235	9.4.2	ADC0809 接口电路软硬件 仿真示例	319
8.1.4	8251A 的应用实例	242	9.5	习题	323
8.1.5	8251A 串行通信 Proteus 仿真示例	245	附录		324
8.2	并行通信及可编程序接口芯片 应用技术	250	附录 A	ASCII (美国标准信息交换码) 码表	324
8.2.1	并行通信及接口基本概念	250	附录 B	80x86 指令系统表	325
8.2.2	简单并行 I/O 接口芯片	252	附录 C	DOS 系统功能调用	333
8.2.3	可编程序并行接口芯片 8255A	253	附录 D	BIOS 中断调用	337
8.2.4	8255A 控制字及工作方式	257	参考文献		340
8.2.5	8255A 应用举例	266			
8.2.6	8255A 并行通信 Proteus 仿真 示例	271			
8.3	可编程序定时器/计数器芯片及 应用技术	274			

第1章 计算机基础知识

本章以计算机产生的结构思想为引导，首先对计算机的产生及冯·诺依曼计算机的经典设计方案进行了概述，然后介绍了计算机中表示信息的数制及其转换方法、二进制运算及实现电路、二进制数制编码，最后详细介绍了微型计算机的基本概念及系统组成。

1.1 计算机的产生、发展及应用特点

1946年2月14日，在美国宾夕法尼亚（Pennsylvania）大学的一间大厅里，由美国陆军的一位将军按下了一个按钮，一件对现代世界影响巨大的事件发生了，世界上第一台电子多用途数字计算机（ENIAC）启动了，如图1-1所示。ENIAC能够重新编程，解决各种计算问题。ENIAC是按照十进制，而不是按照二进制来计算的。

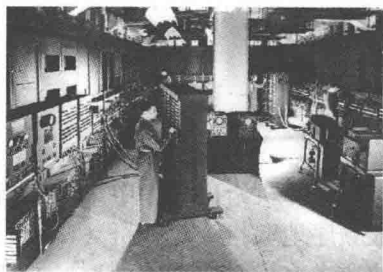


图1-1 ENIAC电子数字计算机

1.1.1 计算机产生的结构思想

在ENIAC充分运行之前，离散变量自动电子计算机（EDVAC）的建造计划就被提出，其设计工作就已经开始。ENIAC和EDVAC的建造者均为宾夕法尼亚大学的电气工程师约翰·莫奇利和普雷斯波·艾克特。和ENIAC一样，EDVAC也是为美国陆军阿伯丁试验场的弹道研究实验室研制的。与ENIAC不同的是，冯·诺依曼（Von Neumann）以技术顾问形式加入，他总结和详细说明了EDVAC的逻辑设计，1945年6月发表了著名的关于EDVAC的报告草案，提出了以“二进制存储信息”“存储程序（自动执行程序）”为基础的计算机结构思想，报告提出的体系结构一直延续至今，即冯·诺依曼结构。EDVAC是冯·诺依曼结构的通用电子计算机，从1951年EDVAC成功运行开始，计算机经历了多次的更新换代，但使用的仍然是冯·诺依曼结构，冯·诺依曼与EDVAC计算机如图1-2所示。

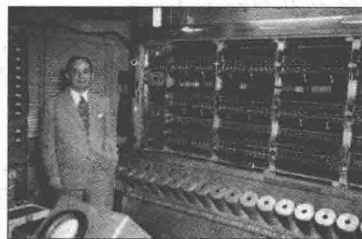


图1-2 冯·诺依曼与EDVAC

冯·诺依曼计算机设计方案包含以下3个要点。

- 1) 采用二进制数的形式表示指令和数据。
- 2) 将指令和数据存放在存储器中。
- 3) 计算机硬件由控制器、运算器、存储器、输入设备和输出设备5部分组成。

在计算机中，二进制数是计算机硬件能直接识别并进行处理的唯一形式。

计算机所做的任何工作都必须以二进制数据所表示的指令形式送入计算机内存中存储，一条条有序指令的集合称为程序。

根据冯·诺依曼的设计，计算机应能自动执行程序，而执行程序又归结为逐条执行指

令。计算机对任何问题的处理都是对数据的处理，计算机所做的任何操作都是执行指令的结果。充分理解和理解计算机产生的结构思想，才能理解数据、程序与计算机硬件之间的关系，这对于学习和掌握计算机基本原理是十分重要的。

按照冯·诺依曼结构思想，计算机由运算器、控制器、存储器、输入设备和输出设备组成，如图 1-3 所示。

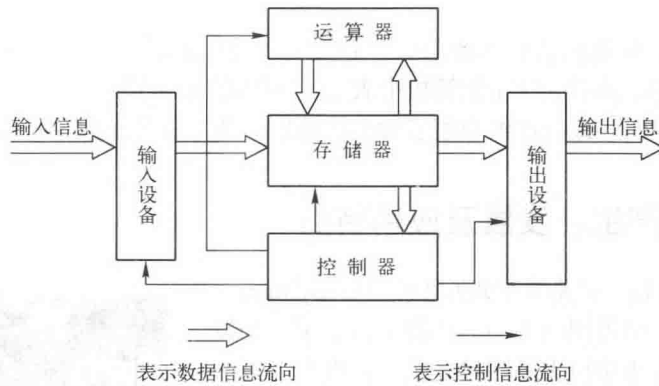


图 1-3 计算机硬件经典结构

计算机给出程序中第一条指令在存储器中的存储地址，控制器则依据第一条指令的地址顺序地取指令、分析（译码）指令、执行指令。在执行指令的过程中，运算器根据指令的要求完成对数据的处理，并把处理结果送入存储器存储。然后，由输出设备展示数据处理结果。这样，在控制器的控制下，计算机周而复始地完成全部的指令流操作，从而实现程序控制。

1.1.2 计算机的发展过程

从 ENIAC 的诞生到现在，计算机在体积、性能、速度、应用领域、生产成本等各方面都发生了巨大的变化。通常以计算机所使用的主要电子元器件为衡量标准，可以把计算机的发展历程划分为如下 4 代。

第 1 代：电子管计算机时代（约 1946~1957 年）。

采用电子管作为运算和逻辑元器件，以二进制代码和汇编语言编写指令和程序，主要用于科学和工程计算。

第 2 代：晶体管计算机时代（约 1958~1964 年）。

采用晶体管作为运算和逻辑元器件，产生了高级程序设计语言，主要用于科学计算、数据处理，并开始用于过程控制。

第 3 代：小规模集成电路计算机时代（约 1965~1971 年）。

采用半导体集成电路作为计算机的主要功能部件。计算机在科学计算、数据处理及过程控制等领域得到了更加广泛的应用。

第 4 代：大规模或超大规模集成电路计算机时代（1972 年开始至今）。

采用大规模集成电路（LSI）和超大规模集成电路（VLSI）作为计算机的主要功能部件。随着 CPU 技术的迅速发展，相继诞生了 32 位、64 位及双内核 64 位微处理器，时钟主频达到 2GHz，如图 1-4 所示。半导体存储器因集成度不断提高，容量也越来越大，外存广泛使用软

硬磁盘和光盘，各种输入/输出设备相继出现。软件产业高速发展，各种系统软件、应用软件相继出现且日趋完善，尤其是 20 世纪 90 年代开始的计算机网络及多媒体技术的迅猛发展，使计算机广泛应用于社会各个领域。

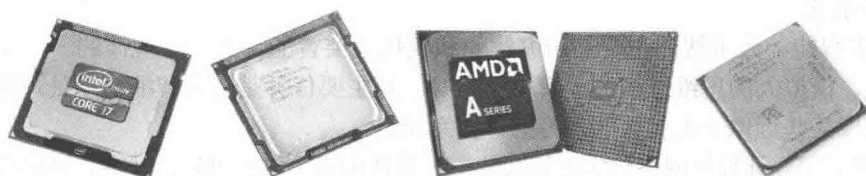


图 1-4 超大规模集成电路制作的 CPU 芯片 (Intel Core i、AMD A)

在应用需求的强力推动下，计算机未来发展趋势主要集中在以下几个方面。

- 1) 提高计算机处理速度是计算机发展的主要目标。
- 2) 在提高性能指标前提下，计算机继续朝着微型化方向发展。
- 3) 计算机价格的下降趋势还在继续发展。
- 4) 计算机的信息处理功能更加多媒体化。
- 5) 网络通信更加完善、广泛。
- 6) 计算机智能化可以模拟人的感觉、行为、思维过程的机理，使计算机尽量具备和人一样的思维和行为能力。

1.1.3 计算机的特点及应用

计算机是一种能迅速而高效地自动完成信息处理的电子设备，它能按照程序对信息进行加工、处理和存储。

1. 计算机的特点

现代计算机具有以下主要特点。

- 1) 具有很高的信息处理速度。中央处理器 (CPU) 的速度几乎每两年翻一番，计算机的运算速度已达上亿次/s。
- 2) CPU 芯片的集成化程度越来越高。CPU 芯片上的半导体管器件的数量每 18 个月翻一番。
- 3) 具有极大的信息存储容量。
- 4) 具有精确的计算能力和逻辑判断能力。
- 5) 具有多样的输入/输出手段和多媒体信息处理能力。
- 6) 计算机网络使资源能够共享，信息迅速而方便地向四面八方传递。

2. 计算机的应用

计算机的主要应用有以下几个方面。

- 1) 科学计算和科学研究。计算机主要应用于解决科学研究和工程技术中所提出的数学问题 (数值计算)。
- 2) 数据处理 (信息处理)。主要是利用计算机速度快和精度高的特点来对数字信息进行加工。
- 3) 工业控制。用单板微型计算机实现底层的分散过程控制级 (DDC 级) 控制，用微型计算机实现中间层的监督控制级 (SCC 级) 监督管理控制，用高档微型计算机实现 SCC 或低层

MIS 管理。

4) 计算机辅助系统。计算机辅助系统主要有计算机辅助教学 (CAI)、计算机辅助设计 (CAD)、计算机辅助制造 (CAM)、计算机辅助测试 (CAT)、计算机集成制造 (CIMS) 等系统。

5) 人工智能。人工智能主要是研究解释和模拟人类智能行为及其规律的一门学科, 包括智能机器人、模拟人的思维过程、计算机学习等。其主要任务是建立智能信息处理理论, 进而设计可以展现某些近似于人类智能行为的计算系统。

6) 网络应用。计算机网络化实现资源共享、数据传输、电子商务、网络计算和控制等。

7) 家用电器。目前, 大多数家用电器都已嵌入单片微控制器, 使其具有记忆、存储等智能化功能。

1.2 计算机中数据的表示及电路

人们要计算机执行的任何操作都必须转换为计算机所能识别的二进制数的形式, 计算机对任何信息的处理实际上是对二进制数的处理。计算机中各种不同的信息形式, 需要对其进行编码或进行不同码制之间的数据转换。

1.2.1 计算机使用的数制及转换

1. 数制

数制就是计数方式。

日常生活中常用的是十进制计数方式, 而计算机内部使用的是二进制数据。因此, 计算机在处理数据时, 必须进行数制之间的相互转换。

(1) 二进制数

二进制数只有两个数字符号: 0 和 1, 计数时按“逢二进一”的原则进行计数, 也称其基数为二。

一般情况下, 二进制数可表示为: $(110)_2$ 、 $(110.11)_2$ 、 $10110B$ 等。

在任何进制数制中, 每个数所处位置不同, 实际代表的数值也不相同, 把不同位置所表示的数值称为权值, 二进制也如同十进制一样可以写成一种展开的形式。

所谓按位权展开法, 就是将任一 r 进制数的各位的权值乘以该位的数值, 然后求和。

例如, 十进制数 $123=1\times 10^2+2\times 10^1+3\times 10^0$ 。

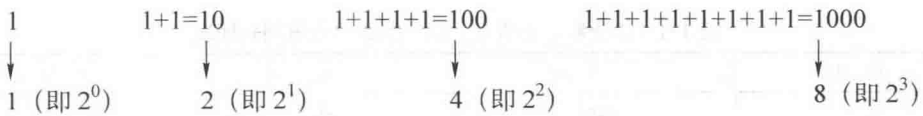
任一个 r 进制数都可以表示成:

$$\begin{aligned} N &= d_m r^m + d_{m-1} r^{m-1} + \cdots + d_0 r^0 + d_{-1} r^{-1} + \cdots + d_{-n} r^{-n} \\ &= \sum_{i=-n}^m d_i r^i \quad (n, m \geq 0) \end{aligned}$$

上式中, r 称为基数 (二进制数为 2), d_i 表明第 i 位上可取的数字 (如二进制数取 0 或 1); i 为 $0\sim m$ 时, 从低到高依次表示整数位, i 为 $-1\sim -n$ 时, 则依次表示小数位; r^i (即 r 的 i 次方) 称为第 i 位的权值。

把一个 r 进制数 N 按权展开, 则 N 可表示为 r 进制数的每位数字 d_i 乘以其权 r^i 所得积之和。

根据位权表示法, 每一位二进制数在其不同位置表示不同的值。例如:



对于 8 位二进制整数（由低位~高位分别用 $D_0 \sim D_7$ 表示）及小数部分（十分位~万分位分别用 $D_{-1} \sim D_{-4}$ 表示），则各位所对应的权值如下。

整数部分								小数部分				
1	1	1	1	1	1	1	1	1	1	1	1
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	D_{-1}	D_{-2}	D_{-3}	D_{-4}

对于任何二进制数，可按位权求和展开为与之相应的十进制数，例如：

$$(10)_2 = 1 \times 2^1 + 0 \times 2^0 = (2)_{10}$$

$$(10.1)_2 = 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} = (2.5)_{10}$$

$$(11)_2 = 1 \times 2^1 + 1 \times 2^0 = (3)_{10}$$

$$(110)_2 = 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (6)_{10}$$

$$(111)_2 = 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (7)_{10}$$

$$(1111)_2 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (15)_{10}$$

$$(10110)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = (22)_{10}$$

对于 8 位二进制整数，其最大值为：

$$(11111111)_2 = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = (255)_{10} = 2^8 - 1$$

对于 16 位二进制整数，其最大值为：

$$(1111111111111111)_2 = 65535 = 2^{16} - 1$$

对于 n 位二进制整数，其最大数值为： $2^n - 1$ 。

例如，二进制数 10110111，按位权展开求和计算可得：

$$\begin{aligned} (10110111)_2 &= 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= 128 + 0 + 32 + 16 + 0 + 4 + 2 + 1 \\ &= (183)_{10} \end{aligned}$$

例如，二进制数 10110.101，按位权展开求和计算可得：

$$\begin{aligned} (10110.101)_2 &= 1 \times 2^4 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 16 + 4 + 2 + 0.5 + 0.125 \\ &= (22.625)_{10} \end{aligned}$$

必须指出：在计算机中，一个二进制数（如 8 位、16 位或 32 位）既可以表示数值，也可以表示一种符号的代码，还可以表示某种操作（即指令），计算机在程序运行时按程序的规则自动识别，这就是所谓的一切信息都是以二进制数据进行存储的。

(2) 十六进制数

计算机在信息输入/输出或书写相应程序或数据时，可采用简短的十六进制数表示相应的位数较长的二进制数。

十六进制数有 16 个数字符号，其中 0~9 与十进制相同，剩余 6 个为 A、B、C、D、E、F（或 a、b、c、d、e、f），分别表示十进制数的 10~15，见表 1-1。

表 1-1 二进制、十进制、八进制、十六进制转换表

十进制	二进制	八进制	十六进制
0	0000B	000Q	0H
1	0001B	001Q	1H
2	0010B	002Q	2H
3	0011B	003Q	3H
4	0100B	004Q	4H
5	0101B	005Q	5H
6	0110B	006Q	6H
7	0111B	007Q	7H
8	1000B	010Q	8H
9	1001B	011Q	9H
10	1010B	012Q	AH
11	1011B	013Q	BH
12	1100B	014Q	CH
13	1101B	015Q	DH
14	1110B	016Q	EH
15	1111B	017Q	FH

十六进制数的计数原则是“逢十六进一”，也称其基数为 16，整数部分各位的权值由低位到高位分别为： 16^0 、 16^1 、 16^2 、 16^3 ...

例如：

$$(31)_{16} = 3 \times 16^1 + 1 \times 16^0 = (49)_{10}$$

$$(2AF)_{16} = 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 = (687)_{10}$$

为了便于区别不同进制的数据，一般情况下可在数据后跟一后缀：

二进制数用“B”表示（如 00111010B）；

十六进制数用“H”表示（如 3A5H）；

十进制数用“D”表示（如 39D 或 39）；

八进制数用“Q”表示（如 123Q）。

对于以 A~F 或 a~f 开始的十六进制数，则在该数据前加 0。

$$\begin{aligned} \text{例如：} \quad 0AF36H &= 0AH \times 16^3 + 0FH \times 16^2 + 3 \times 16^1 + 6 \times 16^0 \\ &= 10 \times 16^3 + 15 \times 16^2 + 3 \times 16^1 + 6 \times 16^0 \end{aligned}$$

(3) 八进制数

八进制数使用 0、1、2、3、4、5、6、7 共 8 个数字符号，八进制数的计数原则是“逢八进一”，也称其基数为 8，整数部分各位的权值由低位到高位分别为： 8^0 、 8^1 、 8^2 、 8^3 ...

$$\text{例如：} \quad (127)_8 = 127Q = 1 \times 8^2 + 2 \times 8^1 + 7 \times 8^0$$

2. 不同数制之间的转换

计算机中的数只能用二进制表示，十六进制数读写方便，而日常生活中使用的是十进制数。因此，计算机必须根据需要对各种进制数据进行转换。

(1) 二进制数与十进制数相互转换

对任意二进制数均可按权值展开将其转化为十进制数。例如：

$$10111\text{B}=1\times 2^4+0\times 2^3+1\times 2^2+1\times 2^1+1\times 2^0$$

$$=23\text{D}$$

$$10111.011\text{B}=1\times 2^4+0\times 2^3+1\times 2^2+1\times 2^1+1\times 2^0+0\times 2^{-1}+1\times 2^{-2}+1\times 2^{-3}$$

$$=23.375\text{D}$$

十进制数转换为二进制数，可将整数部分和小数部分分别进行转换，然后合并。其中整数部分可采用“除2取余法”进行转换，小数部分可采用“乘2取整法”进行转换。

1) 除2取余法。

例如：采用“除2取余法”将37D转换为二进制数。

	余数	低位 (第一次余数必为低位)
2 37	1	↓ 高位 (直到商数等于0为止)
2 18	0	
2 9	1	
2 4	0	
2 2	0	
2 1	1	
0		

把所得余数由高到低排列起来可得：37=100101B。

2) 乘2取整法。

例如：采用“乘2取整法”将0.625转换为二进制数小数。

0.625		
× 2		
1.250	取出整数 1	→ 1 高位 (第一次整数 1 必为二进制数小数权值的最高位)
× 2		
0.500	取出整数 0	→ 0
× 2		
1.000	取出整数 1	→ 1 低位

直至小数部分为0 (若不为0，则根据误差要求取小数位)。

把所得整数由高到低排列起来可得：0.625=0.101B。

同理，把37.625转换为二进制数，只需将以上转换合并起来可得：37.625=100101.101B。

【例 1-1】 把十进制数 57.63 转换成二进制数。

可将十进制数 57.63 的整数部分和小数部分分别转换。十进制整数转换成二进制数，整数部分可以采取除2取余法。小数转换成二进制数可以采取乘2取整法。

求整数部分：

	余数	低位
2 57 1	↓ 高位
2 28 0	
2 14 0	
2 7 1	
2 3 1	
2 1 1	
0		

求小数部分:

$0.63 \times 2 = 1.26$ 整数部分=1
 $0.26 \times 2 = 0.52$ 整数部分=0
 $0.52 \times 2 = 1.04$ 整数部分=1
 $0.04 \times 2 = 0.08$ 整数部分=0

高位
 ↓
 低位

则 $57.63D = 111001.1010B$

十进制数转换为二进制数，也可以采用权值比较法。

所谓权值比较法，即将十进制数与二进制位权从高位到低位逐位比较，若该位十进制数权值大于或等于二进制某位权值，则该位取“1”，否则该位取“0”，采用按位分割法进行转换。

例如：将 37.625 转换为二进制数。

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
0	0	1	0	0	1	0	1

将整数部分 37 与二进制各位权值从高位到低位比较， $37 > 32$ ，则该位取 1，剩余 $37 - 32 = 5$ ，逐位比较，得 00100101B。

将小数部分 0.625 按同样方法转换，得 0.101B。

结果为： $37.625D = 100101.101B$ 。

(2) 二进制数与十六进制数的相互转换

在计算机进行输入、输出显示时，常采用十六进制数。十六进制数可看作是二进制数的简化表示。

因为 $2^4 = 16$ ，所以 1 位十六进制数可表示 4 位二进制数，其二进制、十进制、八进制、十六进制对应数的转换关系见表 1-1。

在将二进制数转换为十六进制数时，其整数部分可由小数点开始向左每 4 位为一组进行分组，直至高位，若高位不足 4 位，则左侧补 0 使其成为 4 位二进制数，然后按表 1-1 中的对应关系进行转换。其小数部分由小数点向右每 4 位为一组进行分组，不足 4 位则末位补 0，使其成为 4 位二进制数，然后按表 1-1 的对应关系进行转换。

例如：

$11000101.011B = 1100,0101.0110B = 0C5.6H$

$10001010B = 1000,1010B = 8AH$

$100101.101B = 0010\ 0101.1010B = 25.AH$

需要将十六进制数转换为二进制数时，则为上述方法的逆过程。

例如： $0C5.A7H = 1100\ 0101.1010\ 0111\ B$

例如：

$7ABFH = 0111\ 1010\ 1011\ 1111\ B$

 $\underbrace{\hspace{2em}} \underbrace{\hspace{2em}} \underbrace{\hspace{2em}} \underbrace{\hspace{2em}}$
 7 A B F

即 $7ABFH = 11110101011111B$

(3) 二进制数与八进制数的相互转换

因为 $2^3 = 8$ ，所以 1 位八进制数可表示 3 位二进制数，转换关系见表 1-1。

在将二进制数转换为八进制数时，其整数部分可由小数点开始向左每 3 位为一组进行分

组，直至高位。若高位不足3位，则补0，使其成为3位二进制数，然后按表1-1的对应关系进行转换。其小数部分由小数点向右每3位为一组进行分组，不足3位则末位补0，使其成为3位二进制数，然后按表1-1的对应关系进行转换。

例如：123Q=001 010 011B

1011111B=001 011 111B=137Q

(4) 十进制数与十六进制数的相互转换

十进制数与十六进制数的相互转换可直接进行，也可先转换为二进制数，然后再把二进制数转换为十六进制数或十进制数。

例如：将十进制数37D转为十六进制数。

37D=100101B=00100101B=25H

例如：将十六进制数41H转换为十进制数。

41H=01000001B=65D

也可按位权展开求和方式将十六进制数直接转换为十进制数，这里不再详述。

1.2.2 二进制运算及电路实现

对二进制数0和1，既可以表示数值，也可以表示逻辑关系，因而二进制数有两种不同类型的运算，即算术运算和逻辑运算。

1. 二进制算术运算及电路实现

二进制加法运算是计算机的基本运算，其运算电路是计算机中最核心的功能电路。

(1) 二进制算术运算

1位二进制数的加法基本运算规则如下。

0	0	1	1
+0	+1	+0	+1
0	1	1	10

(向高位进1)

对于多位二进制数的运算，从低位到高位依次进行相加，逢二进一，同时要考虑低位向高位产生的进位位。

例如，8位二进制数加法

$$\begin{array}{r}
 11100101 \\
 + 01110011 \\
 \hline
 01011000
 \end{array}$$

(2) 1位二进制加法电路

1) 半加器电路。

设第0位二进制数输入量 A_0 和 B_0 相加，输出量为本位和 S_1 和进位位 C_0 ，其真值表如图1-5a所示。由真值表得到输入与输出之间的逻辑表达式如下。

$$S_0 = \bar{A}_0 B_0 \vee A_0 \bar{B}_0 = A_0 \oplus B_0$$

$$C_1 = A_0 \wedge B_0$$

由此得到1位二进制加法电路称为半加器电路，如图1-5b所示。