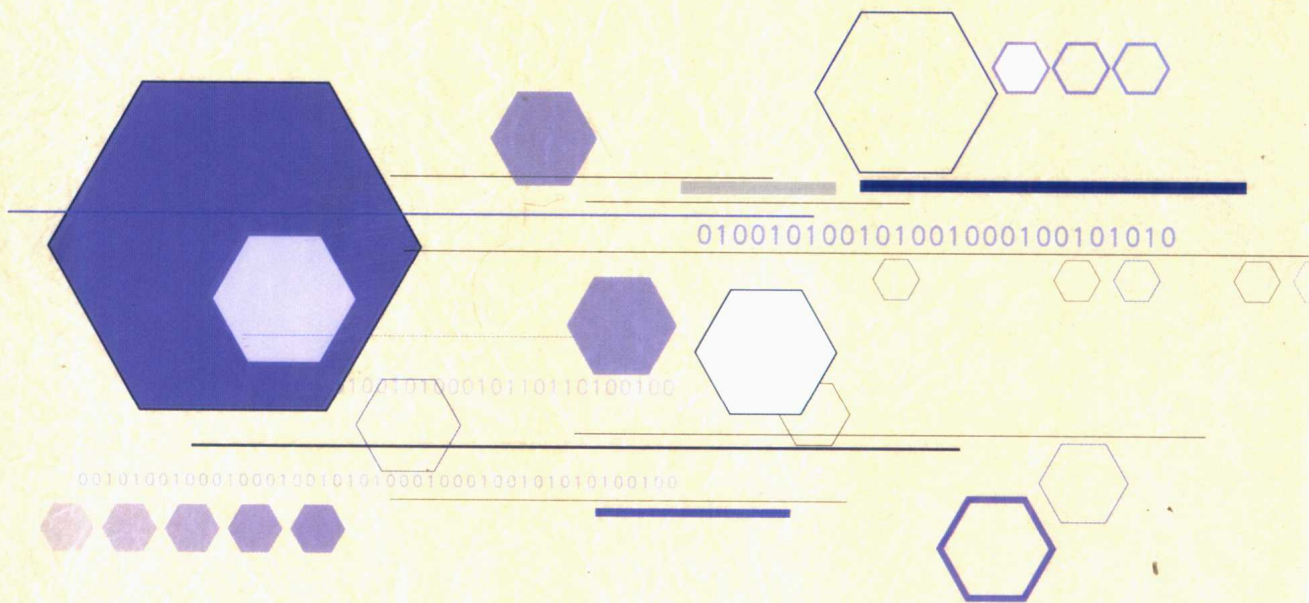


“十一五”高等院校规划教材



# 单片机基础

(第3版)



李广弟 朱月秀 冷祖祁 编著

 本书配套多媒体教学课件



北京航空航天大学出版社

“十一五”高等院校规划教材

# 单片机基础

(第3版)

李广弟 朱月秀 冷祖祁 编著

北京航空航天大学出版社

## 内 容 简 介

本书内容在前两版的基础上做了适度增删。删去了第2版中有关16位单片机的介绍,而对8位单片机的内容进行了扩充,特别是扩充了有关串行扩展的知识。本书着重讲述8位单片机的典型代表80C51系列,介绍其基本原理和应用。主要包括:80C51单片机的硬件结构及串并行扩展、指令系统和汇编语言程序设计以及单片机的发展和应用等。

本书内容系统全面,通俗易懂,适于初学者。可作为本科、专科、函授或培训班的教材,同样也可作为工程技术人员或业余智能产品开发爱好者的自学用书。

本书配套教学课件。

### 图书在版编目(CIP)数据

单片机基础/李广弟,朱月秀,冷祖祁编著. —3版.

北京:北京航空航天大学出版社,2007.6

ISBN 978-7-81077-837-4

I. 单… II. ①李…②朱…③冷… III. 单片微型计算机  
IV. TP368.1

中国版本图书馆CIP数据核字(2007)第054860号

© 2007,北京航空航天大学出版社,版权所有。

未经本书出版者书面许可,任何单位和个人不得以任何形式或手段复制或传播本书内容。侵权必究。

### 单片机基础(第3版)

李广弟 朱月秀 冷祖祁 编著

责任编辑 崔肖娜 王艳花

\*

北京航空航天大学出版社出版发行

北京市海淀区学院路37号(100083) 发行部电话:010-82317024 传真:010-82328026

<http://www.buaapress.com.cn> E-mail: [bhpress@263.net](mailto:bhpress@263.net)

北京市松源印刷有限公司印装 各地书店经销

\*

开本:787×960 1/16 印张:17 字数:381千字

2007年6月第1版 2007年7月第2次印刷 印数:5 001~13 000册

ISBN 978-7-81077-837-4 定价:24.00元

# 第3版前言

单片机是计算机技术、大规模集成电路技术和控制技术的综合产物。经过30多年的发展历程,单片机应用已十分广泛和深入。据Motorola公司统计,1990年,平均每辆汽车使用12个单片机,而到了2000年就增加到35个。所以可以毫不夸张地说,任何设备和产品的自动化、数字化和智能化都离不开单片机。现在,凡是电脑控制的设备和产品,必有单片机嵌入其中。这一切表明,单片机已成为人类生活中不可或缺的助手。

单片机应用系统设计不但要熟练掌握单片机程序设计语言和编程技术,而且还要具备扎实的单片机硬件方面的理论和实践知识。另外,考虑到当今使用最多的是8位单片机,所以在本次修订中着重突出以80C51为代表的8位单片机的基础地位。遵循软硬件并重和串并行并重的原则,详细讲述单片机的原理、应用和发展。

本次修订涉及的内容较多。修订后的第1章计算机基础知识,主要是针对那些没有系统学习过计算机知识的读者而准备的,这部分内容对后面的学习十分有用。第2、3、4、5章主要讲述80C51单片机的硬件和软件知识。第6、7、8、9、10章主要介绍单片机的系统扩展。第11、12章则主要介绍单片机的发展与应用。为帮助读者深入学习,各章后都配有适量练习题。

本次修订由李广弟、朱月秀和冷祖祁共同完成,并得到清华大学陆延丰老师的热情帮助。此外,在本书编写和修订过程中,李禹成、李铁庸、张金环、王玉民、李琳、史立红、黄堃、杜文仪、郭昊、任丽华、张蕴颖、吴友等同志也做了大量具体工作,在此对他们表示深深的谢意。在本书编写和修订过程中,我们还学习和参考了许多单片机方面的教材和资料,受益匪浅,在此向各位作者表示谢意。

鉴于作者水平有限,加之时间仓促,因此对书中错误和疏漏之处,敬请各位新老读者批评指正。

本教材配有教学课件。需要用于教学的老师,请与北京航空航天大学出版社联系。  
联系方式如下:

电话:010-82317027

传真:010-82327026

E-mail: bhkejian@126.com

作者

2007年1月

# 目 录

## 第 1 章 计算机基础知识

1.1 二进制数及其在计算机中的使用 .....	1
1.1.1 二进制数的进位计数特性 .....	1
1.1.2 机器数与机器数表示形式 .....	2
1.1.3 计算机中二进制数的单位 .....	4
1.1.4 计算机使用二进制数的原因 .....	4
1.2 二进制数的算术运算和逻辑运算 .....	5
1.2.1 二进制算术运算 .....	5
1.2.2 二进制逻辑运算 .....	7
1.3 供程序设计使用的其他进制数 .....	9
1.3.1 十进制数与十六进制数 .....	9
1.3.2 不同进制数之间的转换 .....	10
1.4 计算机中使用的编码 .....	12
1.5 微型计算机概述 .....	14
1.5.1 微型计算机硬件系统 .....	14
1.5.2 微型计算机软件系统 .....	15
1.5.3 微型计算机的工作过程 .....	16
练习题 .....	16

## 第 2 章 80C51 单片机的硬件结构

2.1 单片机的概念 .....	18
2.2 80C51 单片机的逻辑结构及信号引脚 .....	18
2.2.1 80C51 单片机的内部逻辑结构 .....	19
2.2.2 80C51 单片机的封装与信号引脚 .....	21
2.3 80C51 单片机的内部存储器 .....	23
2.3.1 内部数据存储器低 128 单元区 .....	23
2.3.2 内部数据存储器高 128 单元区 .....	25
2.3.3 堆栈操作 .....	30
2.3.4 内部程序存储器 .....	32



2.4	80C51 单片机的并行 I/O 口	32
2.4.1	P0 口逻辑结构	33
2.4.2	P1 口逻辑结构	33
2.4.3	P2 口逻辑结构	34
2.4.4	P3 口逻辑结构	34
2.5	80C51 单片机的时钟与定时	35
2.5.1	时钟电路	35
2.5.2	定时单位	37
2.6	80C51 单片机的系统复位	37
2.6.1	复位方式与初始化状态	37
2.6.2	复位电路	38
2.7	单片机低功耗工作模式	40
2.7.1	单片机低功耗的意义	40
2.7.2	两种低功耗工作模式	41
2.7.3	低功耗模式的应用	42
	练习题	43
<b>第3章 80C51 单片机指令系统</b>		
3.1	单片机指令系统概述	45
3.2	80C51 单片机指令寻址方式	46
3.3	80C51 单片机指令分类介绍	49
3.3.1	数据传送类指令	50
3.3.2	算术运算类指令	55
3.3.3	逻辑运算及移位类指令	60
3.3.4	控制转移类指令	64
3.3.5	位操作类指令	70
	练习题	74
<b>第4章 80C51 单片机汇编语言程序设计</b>		
4.1	单片机程序设计语言概述	76
4.1.1	机器语言和汇编语言	76
4.1.2	单片机使用的高级语言	76
4.1.3	80C51 单片机汇编语言的语句格式	77
4.2	汇编语言程序的基本结构形式	78
4.2.1	顺序程序结构	79
4.2.2	分支程序结构	79
4.2.3	循环程序结构	83
4.3	80C51 单片机汇编语言程序设计举例	83

4.3.1	算术运算程序	84
4.3.2	定时程序	88
4.3.3	查表程序	90
4.4	单片机汇编语言源程序的编辑和汇编	91
4.4.1	手工编程与汇编	92
4.4.2	机器编辑与交叉汇编	92
4.5	80C51 单片机汇编语言伪指令	93
	练习题	96
<b>第 5 章 80C51 单片机的中断与定时</b>		
5.1	中断概述	99
5.2	80C51 单片机的中断系统	100
5.2.1	中断源与中断向量	100
5.2.2	中断控制	101
5.2.3	中断优先级控制	103
5.2.4	中断响应过程	105
5.2.5	中断服务程序	106
5.3	80C51 单片机的定时器/计数器	108
5.3.1	定时器/计数器的计数和定时功能	108
5.3.2	用于定时器/计数器控制的寄存器	109
5.3.3	定时器工作方式 0	110
5.3.4	定时器工作方式 1	113
5.3.5	定时器工作方式 2	113
5.3.6	定时器工作方式 3	115
	练习题	117
<b>第 6 章 单片机并行存储器扩展</b>		
6.1	单片机并行外扩展系统	119
6.1.1	单片机并行扩展总线	119
6.1.2	并行扩展系统的 I/O 编址和芯片选取	121
6.2	存储器分类	123
6.2.1	只读存储器	123
6.2.2	读/写存储器	125
6.3	存储器并行扩展	125
6.3.1	程序存储器并行扩展	125
6.3.2	数据存储器并行扩展	127
6.3.3	使用 RAM 芯片扩展可读/写的程序存储器	128
6.4	80C51 单片机存储器系统的特点和使用方法	129



6.4.1 单片机存储器系统的特点 .....	129
6.4.2 80C51 单片机存储器的使用 .....	130
练习题 .....	131
<b>第7章 单片机并行 I/O 扩展</b>	
7.1 单片机 I/O 扩展基础知识 .....	133
7.1.1 I/O 接口电路的功能 .....	133
7.1.2 关于接口电路的更多说明 .....	134
7.1.3 I/O 编址技术 .....	135
7.1.4 单片机 I/O 控制方式 .....	136
7.2 可编程并行接口芯片 8255 .....	137
7.2.1 8255 硬件逻辑结构 .....	137
7.2.2 8255 工作方式 .....	139
7.2.3 8255 的编程内容 .....	139
7.2.4 8255 接口应用 .....	141
7.3 键盘接口技术 .....	142
7.3.1 键扫描和键码生成 .....	142
7.3.2 用 8255 实现键盘接口 .....	145
7.4 LED 显示器接口技术 .....	148
7.4.1 LED 显示器概述 .....	148
7.4.2 LED 显示器显示原理 .....	149
7.4.3 LED 显示器接口 .....	150
7.5 打印机接口技术 .....	152
7.5.1 微型打印机概述 .....	152
7.5.2 打印机接口 .....	152
练习题 .....	155
<b>第8章 80C51 单片机串行通信</b>	
8.1 串行通信基础知识 .....	157
8.1.1 异步通信和同步通信 .....	157
8.1.2 串行通信线路形式 .....	159
8.2 80C51 串行口 .....	160
8.2.1 80C51 串行口硬件结构 .....	160
8.2.2 串行口控制机制 .....	161
8.3 80C51 串行口工作方式 .....	162
8.3.1 串行工作方式 0 .....	162
8.3.2 串行工作方式 1 .....	163
8.3.3 串行工作方式 2 和 3 .....	164





8.4 串行通信数据传输速率 .....	164
8.4.1 传输速率的表示方法 .....	164
8.4.2 80C51 的波特率设置 .....	165
8.5 串行通信应用 .....	166
8.5.1 近程串行通信 .....	166
8.5.2 调制解调器的使用 .....	167
8.5.3 双机通信 .....	168
8.5.4 多机通信 .....	172
练习题 .....	174
<b>第 9 章 单片机串行扩展</b>	
9.1 单片机串行扩展概述 .....	176
9.1.1 单片机需要串行扩展的原因 .....	176
9.1.2 单片机串行扩展实现方法 .....	176
9.2 I <sup>2</sup> C 总线 .....	178
9.2.1 I <sup>2</sup> C 总线结构和信号 .....	178
9.2.2 I <sup>2</sup> C 总线数据传输方式 .....	181
9.2.3 器件与器件寻址 .....	183
9.3 单片机 8×C552 的 I <sup>2</sup> C 总线 .....	185
9.3.1 8×C552 的 I <sup>2</sup> C 总线接口电路 .....	185
9.3.2 8×C552 的 I <sup>2</sup> C 总线控制机制 .....	188
9.3.3 由 8×C552 构成的单主 I <sup>2</sup> C 总线系统 .....	192
9.4 单片机 8×C552 的串行扩展 .....	195
9.4.1 通过 I <sup>2</sup> C 总线扩展串行数据存储器 .....	196
9.4.2 I <sup>2</sup> C 总线的发展 .....	199
9.4.3 通过 I <sup>2</sup> C 总线扩展 LED 显示器 .....	200
9.5 单片机 80C51 的串行扩展 .....	202
9.5.1 通过 UART 进行串行程序存储器扩展 .....	202
9.5.2 串行接口的软件模拟 .....	203
9.5.3 I <sup>2</sup> C 总线接口芯片 PCF8584 .....	205
练习题 .....	207
<b>第 10 章 单片机 A/D 及 D/A 转换接口</b>	
10.1 单片机测控系统与模拟输入通道 .....	209
10.1.1 单片机测控系统概述 .....	209
10.1.2 模拟输入通道 .....	209
10.2 A/D 转换器接口 .....	212
10.2.1 8 位 A/D 转换芯片与 80C51 接口 .....	212



10.2.2	12位 A/D 转换芯片与 80C51 接口	215
10.2.3	A/D 转换芯片应用说明	215
10.3	D/A 转换器接口	218
10.3.1	D/A 转换芯片	218
10.3.2	DAC0832 单缓冲连接方式	220
10.3.3	DAC0832 双缓冲连接方式	224
10.4	A/D 与 D/A 转换器芯片的串行接口	226
10.4.1	通过 I <sup>2</sup> C 总线的串行接口	226
10.4.2	通过软件模拟的串行接口	228
	练习题	229

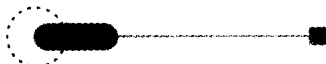
### 第 11 章 8 位单片机的发展

11.1	80C51 单片机的发展	230
11.1.1	在 MCS-51 基础上发展起来的 80C51	230
11.1.2	80C51 的衍生芯片	231
11.2	从 8×C522 看 8 位单片机功能的增强	232
11.2.1	8×C522 的硬件结构	233
11.2.2	事件捕捉与事件定时输出	237
11.2.3	监视定时器 WDT	238
11.2.4	脉宽调制器 PWM	240
11.3	闪速存储器及其在单片机中的应用	242
11.3.1	闪速存储器概述	242
11.3.2	闪速存储芯片	242
11.3.3	闪存单片机芯片	245
11.3.4	闪速存储器编程	247
	练习题	249

### 第 12 章 单片机应用

12.1	单片机简单控制应用	250
12.1.1	时钟计时	250
12.1.2	数字式热敏电阻温度计	254
12.2	单片机应用的发展	258
12.2.1	微控制技术与嵌入式系统	258
12.2.2	单片机的 Internet 技术	258
12.3	单片机开发系统	259

### 参考文献



# 第 1 章

## 计算机基础知识

### 1.1 二进制数及其在计算机中的使用

二进制数是计算机工作的基础,在计算机中只能使用二进制数。所有指令、数据、字符和地址的表示,以及它们的存储、处理和传送,都是以二进制的形式进行的,因此计算机的电路逻辑和处理方法也都是按二进制的原则实现的。正因为计算机是建立在二进制基础之上的,所以有人说,没有二进制也就没有电子计算机。

#### 1.1.1 二进制数的进位计数特性

##### 1. 进位计数制

进位计数制的最大特点是:同样的数字符号,由于在数字序列中所处的位置不同,因而它所代表的数值就不同。先以十进制数为例进行说明。例如,十进制数

555.55

同样的数字符号 5,但从左向右第 1 个 5 的值是 500,第 2 个 5 的值是 50,以下依次为 5、0.5 和 0.05。因此,可把这个数的值写成如下计算式:

$$555.55 = 5 \times 10^2 + 5 \times 10^1 + 5 \times 10^0 + 5 \times 10^{-1} + 5 \times 10^{-2}$$

在该式中,各位数的值即为该位数字乘以一个系数,通常把这个系数称作该位数的权。数的位置不同,权的大小也不同,但它们都是 10 的幂的形式。其中整数位的权,从低位向高位依次为  $10^0$ 、 $10^1$ 、 $10^2$ ...;而小数位的权,从高位向低位依次为  $10^{-1}$ 、 $10^{-2}$ 、 $10^{-3}$ ...。进位计数制具有如下特点:

- 数字符号的个数等于计数制的基数;
- 逢基数进位;
- 数字的权与其位置有关,且为基数的幂的形式。

对于十进制数,共有 10 个数字符号,即 0、1、2、3、4、5、6、7、8、9,基数为 10,逢 10 进位,各



位数的权是10的幂。

## 2. 二进制数

二进制数只有两个数字符号0和1,基数为2,逢2进位,各位数的权是2的幂。例如,二进制数1101,其值为十进制数13。该值是由下式计算出来的:

$$1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 0 + 1 = 13$$

下面列出的是几个对应的二进制数与十进制数。从中可以看到,在表示同一个数值时,十进制的位数少,二进制的位数多。因为二进制的基数小。

二进制数	1011	10000	101101	1010101	1111111
十进制数	11	16	45	85	127

### 1.1.2 机器数与机器数表示形式

计算机中使用的二进制数称为机器数,对于机器数,需要了解它的多种表示形式。

#### 1. 机器数

由于计算机中的二进制数称为机器数,反之,可以说机器数都是二进制数,因为在计算机中只能使用二进制数。机器数的值称为真值。机器数有多种表示形式,下面作详细介绍。

#### 2. 符号数和无符号数

符号数和无符号数是针对符号出现的两种机器数表示方法。同一个二进制数,对符号数和无符号数具有不同的含义。首先我们讨论符号及符号数值化的问题,然后再说明符号数和无符号数。

符号数具有正负的概念,其值可正、可负。书写中,为表示数的正负,需在数的前面加一个正负号,例如:

+75 的二进制表示为 +1001011

-75 的二进制表示为 -1001011

由于计算机无法在二进制数前直接“写”上一个正号或负号,因此,采用符号数值化的方法表示机器数。所谓符号数值化,就是用二进制数“0”代表正号,用二进制数“1”代表负号,并放在数码序列的前面。例如:

+75 的机器数表示为 01001011

-75 的机器数表示为 11001011

即符号数的最高位为符号位,其余位为数值位。8位二进制数是计算机的基本数据单位,称为字节。这样,对于一字节的符号数,由1位符号位(最高位)和7位数值位(剩余位)组成。

无符号数是逻辑数,没有正负的概念,就是一串二进制代码。无符号数用于表示没有正负概念的数和代码等,例如,存储器地址就是一串无符号的二进制数,表示字符的 ASCII 码就是一组无符号的二进制数。

### 3. 定点数与浮点数

定点数和浮点数是针对小数点出现的两种机器数表示方法。计算机中数的表示不但有正负之分,而且还要考虑整数和小数的问题。因为在数值计算中,小数的存在是不可避免的,定点数和浮点数就是用于解决这个问题的。定点和浮点中的“点”是指小数点。

定点数中小数点的位置固定。按小数点在计算机中的位置,定点数又分为定点小数和定点整数两种表示方法。而浮点数的小数点位置是不固定的。

### 4. 原码、反码和补码

机器数有原码、反码和补码共 3 种表示方法。一种数有多种表示方法是简化运算电路和提高运算速度的需要。

#### (1) 原 码

原码是二进制数符号数值化以后的表示形式,是机器数的原始表示,是对应于反码和补码的称呼。

#### (2) 反 码

正数的反码与原码相同。负数的反码是由原码转换得到的,转换方法为:符号位不变,数值位按位取反。例如:

十进制数+87 的原码表示为  $[+87]_{\text{原}} = 01010111$

十进制数+87 的反码表示为  $[+87]_{\text{反}} = 01010111$

十进制数-87 的原码表示为  $[-87]_{\text{原}} = 11010111$

十进制数-87 的反码表示为  $[-87]_{\text{反}} = 10101000$

#### (3) 补 码

正数的补码与原码相同。负数的补码是把反码的最低位加 1。例如:

十进制数+4 的原码表示为  $[+4]_{\text{原}} = 00000100$

十进制数+4 的反码表示为  $[+4]_{\text{反}} = 00000100$

十进制数+4 的补码表示为  $[+4]_{\text{补}} = 00000100$

十进制数-4 的原码表示为  $[-4]_{\text{原}} = 10000100$

十进制数-4 的反码表示为  $[-4]_{\text{反}} = 11111011$

十进制数-4 的补码表示为  $[-4]_{\text{补}} = 11111100$

原码、反码和补码都是二进制符号数的表示方法,其共同特点是:最高位为符号位,正数的原码、反码和补码相同。此外,还应注意以下两点:



- 负数补码的转换过程是：原码→反码→补码。
- 负数的补码再取补就得到原码，以十进制数-95为例进行说明：

$$[-95]_{\text{原}} = 11011111$$

$$[-95]_{\text{反}} = 10100000$$

$$[-95]_{\text{补}} = 10100001$$

$$[(-95)_{\text{补}}]_{\text{反}} = 11011110$$

$$[(-95)_{\text{补}}]_{\text{补}} = 11011111(\text{原码})$$

讲解原码和反码是为了引出补码的概念，因为补码在计算机中被广泛采用。补码运算可以将符号位当成数据位对待，因此可以把有符号数与无符号数统一起来，并将二进制减法运算变为加法运算，从而给符号数的运算提供了方便；同时也有利于简化运算电路，提高运算速度。

### 1.1.3 计算机中二进制数的单位

在计算机中使用的二进制数共有3个单位，从小到大依次为：位、字节和字。

#### 1. 位(Bit)

这里所说的位是指二进制数的位。位是数的最小单位，Bit是位的英文名称，读作“比特”。在计算机中位仅有0和1两个数值，表示两种状态。

#### 2. 字节(Byte)

8位二进制数称为一个字节，其英文名称是Byte，读作“拜特”，在使用时常用大写字母B表示。字节是最基本的数据单位，计算机中的数据、代码、指令、地址多以字节为单位。

#### 3. 字(Word)

字是一台计算机上所能并行处理的二进制数，字的位数(或长度)称之为字长。字长必须是字节的整数倍。例如，MCS-51单片机字长为8位，MCS-96单片机字长为16位，在微型机中还有32位、64位字长的计算机。

### 1.1.4 计算机使用二进制数的原因

为什么在计算机中要使用人们不熟悉的二进制数呢？其原因主要有以下几点：

#### 1. 易于实现

在计算机中，数是用不同的物理状态来表示的。因为二进制数只有两个数字0和1，用两种物理状态就可以表示出来。而两种相反的物理状态在技术上极易实现。例如，开关的接通与断开，晶体管的导通与截止，电平的高低，脉冲的有无，磁性物质的不同磁化方向等。

对于这样两种截然相反的物理状态，不但易于实现，而且状态稳定可靠。而两种以上的物理状态，不但难以实现，而且稳定性也差。



## 2. 运算简单

因为二进制数只有两个数字,所以对二进制数的运算比我们熟悉的十进制数的运算要简单得多,而运算简单将有利于简化计算机的电路结构。

## 3. 具有逻辑属性

由于二进制数的 0 和 1 正好与逻辑值的“假(F)”和“真(T)”相对应,因此可以使用二进制数实现逻辑运算,从而使逻辑代数运算成为可能。

## 4. 可靠性高

由于二进制数用两种截然相反的物理状态表示,十分稳定。因此二进制数的处理、存储和传送都最为可靠。

## 5. 节省硬件设备

根据理论计算,构成计算机硬件系统设备最省的是  $e$  进制( $e=2.72\cdots$ )。但  $e$  不是整数,无法作为进制的基数。与  $e$  最接近的数是 3,但使用三进制不具备上述优点。而继三进制之后,设备最省的就是二进制了。

# 1.2 二进制数的算术运算和逻辑运算

尽管计算机功能十分强大,但计算机的运算基础却很简单。因为它只能进行二进制数的算术运算和逻辑运算,一切复杂的运算和操作都是建立在这些最基本的算术和逻辑运算之上的。

## 1.2.1 二进制算术运算

与十进制相同,二进制算术运算也包括加、减、乘、除运算。

### 1. 二进制加法运算

二进制加法运算的规则为:

$$0+0=0$$

$$1+0=1$$

$$0+1=1$$

$$1+1=0 \quad \text{向上位进位 1}$$

$$1+1+1=1 \quad \text{向上位进位 1}$$

例如,对二进制数 1101 和 1011 进行加法运算,即



$$\begin{array}{r}
 1101 \text{ (被加数)} \\
 +)1011 \text{ (加数)} \\
 \hline
 111 \text{ (进位)} \\
 1\leftarrow 1000 \text{ (和)}
 \end{array}$$

## 2. 二进制减法运算

二进制减法运算的规则为:

$$\begin{array}{l}
 0-0=0 \\
 1-1=0 \\
 1-0=1 \\
 0-1=1 \quad \text{向上位借位 1}
 \end{array}$$

例如,对二进制数 1101 和 1011 进行减法运算,即

$$\begin{array}{r}
 1101 \text{ (被减数)} \\
 -)1011 \text{ (减数)} \\
 \hline
 1 \text{ (借位)} \\
 0010 \text{ (差)}
 \end{array}$$

## 3. 二进制乘法运算

二进制乘法运算的规则为:

$$\begin{array}{l}
 0 \times 0 = 0 \\
 1 \times 0 = 0 \\
 0 \times 1 = 0 \\
 1 \times 1 = 1
 \end{array}$$

可见,1位二进制数的乘法运算比加减法运算还简单,除  $1 \times 1 = 1$  外,其他各种情况都等于 0。

## 4. 二进制除法运算

二进制除法运算的规则为:

$$\begin{array}{l}
 0 \div 0 \quad \text{无意义} \\
 0 \div 1 = 0 \\
 1 \div 1 = 1 \\
 1 \div 0 \quad \text{无意义}
 \end{array}$$



## 1.2.2 二进制逻辑运算

逻辑运算是对逻辑数据进行的运算。由于二进制数具有逻辑属性,因此,可以使用二进制数实现逻辑运算。常用的逻辑运算共有4种,即逻辑“或”运算、逻辑“与”运算、逻辑“非”运算和逻辑“异或”运算。逻辑运算的结果数据也是逻辑型的。

### 1. 逻辑“或”运算

逻辑“或”运算也称为逻辑加法运算,运算符号为“+”或“ $\vee$ ”。假定有逻辑变量A和B,逻辑“或”运算的结果为C,则逻辑“或”运算可表示为:

$$C=A+B \quad \text{或} \quad C=A \vee B$$

对于A和B的不同状态组合,逻辑“或”运算如表1.1所列。

从表1.1中可以看到,在两个逻辑变量中只要有一个为“真”,则逻辑“或”运算的结果即为“真”。能说明逻辑“或”关系的例子很多,例如照明电路上的并联开关,如图1.1所示。

表 1.1 逻辑“或”运算

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

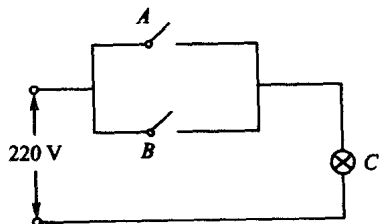


图 1.1 具有逻辑“或”关系的并联开关

A和B两个开关中只要有一个闭合或两个都闭合,都能使电灯C点亮。这种开关并联的关系就是典型的逻辑“或”关系。

在计算机的数据处理应用中,有时需要使用二进制数的逻辑“或”运算来实现,因此逻辑“或”运算也是二进制数的一种基本运算。例如,二进制数1010和1011的“或”运算可表示为:

$$\begin{array}{r} 1010 \\ \vee) 1011 \\ \hline 1011 \end{array}$$

可见,二进制数“或”运算的最重要特点是按位进行,不同位之间不发生任何联系。

### 2. 逻辑“与”运算

逻辑“与”运算也称为逻辑乘法运算,运算符号为“ $\times$ ”、“ $\wedge$ ”或“ $\cdot$ ”。假定有逻辑变量A和B,逻辑“与”运算的结果为C,则逻辑“与”运算可表示为:

$$C=A \times B \quad \text{或} \quad C=A \wedge B \quad \text{或} \quad C=A \cdot B$$