



yingyongxing Rencai Peiyang Xilie Jiaocai

应用型人才培养系列教材

理 工 类

数字电子 技术基础

SHUZI DIANZI JISHU JICHU

杨鲁平 主编



电子科技大学出版社

数字电子技术基础

杨鲁平 主编

电子科技大学出版社

图书在版编目(CIP)数据

数字电子技术基础 / 杨鲁平主编. — 成都: 电子科技大学出版社, 2009.1

(应用型人才培养系列教材. 理工类)

ISBN 978-7-81114-990-6

I. 数… II. 杨… III. 数字电路—电子技术—高等学校—教材 IV. TN79

中国版本图书馆 CIP 数据核字(2008)第 140808 号

内 容 提 要

本书是根据教学大纲, 针对高等院校三级本科电子类工科专业, 由具有丰富课堂教学、实验教学和科研经验的教师编写的专业基础课教材。全书共分八章, 主要介绍了数字逻辑基础、集成门电路集成、组合逻辑电路、触发器、时序逻辑电路、脉冲产生电路、半导体存储器、数/模和模/数电路等。

本书力求在叙述上通俗易懂, 在内容上由浅入深, 并通过分析与设计方法的灵活多样, 使学生比较容易接受和掌握。本书可作为高校工科或理科三级本科生、高职高专学生以及自学考试、夜大、函大学生的基础课程教材, 也可供相关工程技术人员学习参考。

数字电子技术基础

杨鲁平 主编

出 版: 电子科技大学出版社(成都市一环路东一段 159 号电子信息产业大厦 邮编: 610051)

策划编辑: 曾 艺 罗 雅

责任编辑: 陈松明

主 页: www.uestcp.com.cn

电子邮件: uestcp@uestcp.com.cn

发 行: 新华书店经销

印 刷: 四川嘉华印业有限公司

成品尺寸: 175mm×260mm 印张 15 25 字数 350 千字

版 次: 2009 年 1 月第一版

印 次: 2009 年 1 月第一次印刷

书 号: ISBN 978-7-81114-990-6

定 价: 25.00 元

■ 版权所有 侵权必究 ■

- ◆ 本社发行部电话: 028-83202463; 本社邮购电话: 028-83208003。
- ◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。
- ◆ 课件下载在我社主页“下载专区”。

前 言

随着高等院校的发展，教育结构、教学方式的改革，三级本科院校的出现，急需针对三级本科学生编写一套与之相适应的教材，而本教材正顺应了广大三级本科院校发展的需求。

本教材对传统的一本内容做了精选和更新，避开了高深的理论，简明扼要地介绍了学生最需要的基础知识和技术；并用通俗易懂的语言介绍数字电子技术的理论基础知识，强化“外部”，淡化“内部”，着眼于方法和能力的培养；还通过大量的实例、例题和习题说明电路的实际应用，以加深学生对本书内容的掌握和理解，便于学生自学。

本书由贵州大学电子科学与信息技术学院集体编写，由屠葵葵编写第一章和第八章；王蓓编写第二章和第七章；黄晋编写第三章和第四章；刘洪编写第五章和第六章。全书由杨鲁平副教授担任主编，全部编写工作都是在杨鲁平副教授的亲自组织和具体指导下完成的。

由于编者水平有限，书中难免出现错误和不当之处，恳请读者指正。

编 者

2008年3月

目 录

第一章 数字逻辑基础

1.1 绪论.....	2
1.2 数制与码制	3
1.2.1 数制 (Number System)	3
1.2.2 数制间转换.....	4
1.2.3 码制.....	7
1.3 逻辑代数.....	10
1.3.1 逻辑代数的基本运算.....	10
1.3.2 逻辑代数的基本公式与常用公式.....	12
1.3.3 逻辑代数的运算规则.....	15
1.4 逻辑函数表示方法及化简.....	16
1.4.1 逻辑函数表示方法.....	16
1.4.2 逻辑函数化简.....	22
本章小结.....	29
习题.....	29

第二章 逻辑门电路

2.1 概述.....	32
2.2 二极管开关特性.....	32
2.3 三极管的开关特性.....	34
2.4 基本门电路.....	36
2.4.1 二极管与门电路.....	36
2.4.2 三极管非门.....	37
2.5 TTL 门电路.....	39
2.5.1 TTL 反相器.....	39
2.5.2 其他类型的 TTL 门电路.....	42
2.6 MOS 门电路.....	46
2.6.1 CMOS 逻辑门电路.....	46
2.6.2 NMOS 逻辑门电路.....	48
2.7 TTL 电路与 CMOS 电路的接口.....	49
2.7.1 TTL 电路驱动 CMOS 电路.....	50
2.7.2 CMOS 电路驱动 TTL 电路.....	51
本章小结.....	51
习题.....	51



第三章 组合逻辑电路

3.1 组合逻辑电路的分析和设计方法	58
3.1.1 组合逻辑电路的分析方法	58
3.1.2 组合逻辑电路的分析方法	60
3.2 编码器和译码器	63
3.2.1 编码器	63
3.2.2 译码器	68
3.3 数据选择器	75
3.3.1 4选1数据选择器	75
3.3.2 8选1数据选择器	77
3.3.3 数据选择器的应用	79
3.4 显示器	80
3.4.1 七段数码显示器	80
3.4.2 七段显示译码器	81
3.5 数值比较器	83
3.5.1 1位数值比较器	83
3.5.2 多位数值比较器	84
3.6 算术运算电路	87
3.6.1 半加器	87
3.6.2 全加器	88
3.6.3 多位数加法器	89
3.7 组合逻辑电路中的竞争冒险	91
3.7.1 产生竞争冒险的原因	91
3.7.2 冒险现象的判断	92
3.7.3 消除冒险现象的方法	93
本章小结	95
习题	95

第四章 触发器

4.1 基本RS触发器	102
4.1.1 概述	102
4.1.2 由与非门组成的基本RS触发器	102
4.1.3 由或非门组成的基本RS触发器	104
4.2 同步RS触发器	105
4.3 RS主从触发器	107
4.4 JK主从触发器和JK边沿触发器	108
4.4.1 JK主从触发器	108
4.4.2 JK边沿触发器	111
4.5 D触发器	113

4.5.1 同步 D 触发器	113
4.5.2 边沿 D 触发器	115
本章小结	117
习题	118

第五章 时序逻辑电路

5.1 概述	128
5.1.1 时序逻辑电路的特点	128
5.1.2 时序逻辑电路的分类	129
5.2 时序逻辑电路的分析方法	129
5.2.1 时序逻辑电路的状态转换表、状态转换图和时序图	129
5.2.2 同步时序逻辑电路的分析方法	130
5.2.3 异步时序逻辑电路的分析方法	133
5.3 寄存器与移位寄存器	134
5.3.1 寄存器	134
5.3.2 移位寄存器	135
5.4 计数器	139
5.4.1 二进制计数器	140
5.4.2 十进制计数器	147
5.4.3 任意进制计数器的构成方法	155
5.5 时序逻辑电路的设计方法	159
本章小结	162
习题	163

第六章 脉冲波形的产生与整形

6.1 概述	170
6.2 多谐振荡器	170
6.2.1 CMOS 反相器构成的自激多谐振荡器	171
6.2.2 石英晶体多谐振荡器	174
6.3 单稳态触发器	175
6.3.1 门电路组成的单稳态触发器	175
6.3.2 集成单稳态触发器	178
6.3.3 单稳态触发器的应用	180
6.4 施密特触发器	181
6.4.1 用门电路组成的施密特触发器	182
6.4.2 集成施密特触发器	184
6.4.3 施密特触发器的应用	185
6.5 555 定时器及其应用	186
6.5.1 555 定时器的电路结构与功能	187
6.5.2 用 555 定时器接成的施密特触发器	188

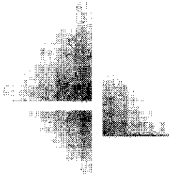
6.5.3 用 555 定时器接成的单稳态触发器	189
6.5.4 用 555 定时器接成的多谐振荡器	191
本章小结	193
习题	194

第七章 半导体存储器

7.1 半导体存储器	202
7.2 半导体存储器组成	203
7.2.1 可擦除可编程只读存储器 (EPROM)	203
7.2.2 静态随机存取存储器 (SRAM) 的存储单元	206
7.2.3 动态随机存储器 (DRAM) 的存储单元	207
7.2.4 RAM 的扩展	209
7.3 可编程逻辑器件	210
7.3.1 可编程逻辑器件	210
7.3.2 PLD 的基本结构	211
本章小结	212
习题	213

第八章 D/A 和 A/D 转换

8.1 D/A 转换	216
8.1.1 D/A 转换的基本原理	216
8.1.2 D/A 转换的主要技术指标	218
8.1.3 典型 D/A 转换器	219
8.2 A/D 转换	221
8.2.1 A/D 转换的基本原理	222
8.2.2 A/D 转换的主要技术指标	225
8.2.3 常用 A/D 转换器	226
本章小结	232
习题	232
参考文献	234



第一章



◎数字逻辑基础

内容提要

数字电路技术是现代电子技术的重要组成部分,主要研究数字逻辑的电路实现方法,所以又称为数字逻辑电路。数字逻辑是用数学方法研究和处理事件之间的逻辑关系,数字逻辑系统就是利用数字逻辑技术建立的系统。随着科学技术的发展,特别是计算机和集成电路技术的发展,数字逻辑和数字电路已成为现代工业的技术基础。因此,数字逻辑系统的分析和设计已成为现代电气、电子和计算机工程师必须掌握的基本技术。本章将介绍有关数字逻辑的基础知识。

1.1 绪论

在自然界中存在着各种各样的物理量,这些物理量可以分为两大类:模拟量和数字量。模拟量在时间和数值上是连续变化的,如声音、温度、速度等都是模拟量;数字量在时间和数值上是不连续变化即离散变化的,如十字路口的交通信号灯、工厂产品的数量等都属于数字量。

在电子电路中,数字量和模拟量都是以电信号形式出现的。表示模拟量的电信号称为模拟信号,它是时间连续、幅值也连续的信号,如正弦波、三角波等(如图 1.1.1 所示)。处理模拟信号的电路称为模拟电路。

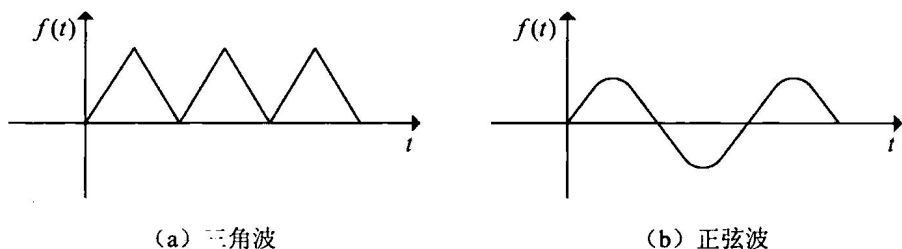
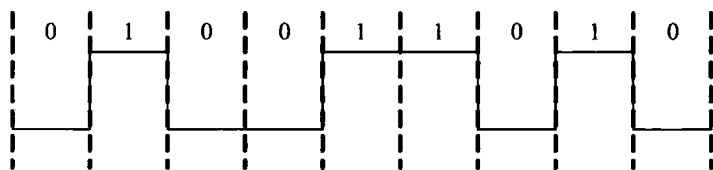


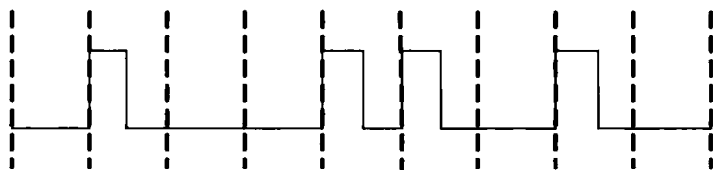
图 1.1.1 模拟信号波形

表示数字量的电信号称为数字信号,它是时间离散、数值也离散的信号。处理数字信号的电路称为数字电路。由于数字电路主要研究电路输入和输出之间的逻辑关系,即研究事件的“有”或“无”、“是”或“非”等互相对立的两种逻辑状态,因此数字信号常用 0 和 1 来表示,通常有电平型和脉冲型两种波形。电平型数字信号以一个时间节拍内信号是高电平还是低电平来表示 1 或 0,而脉冲型数字信号则以一个时间节拍内有无脉冲来表示 1 或 0,如图 1.1.2 所示。

同模拟信号相比,数字信号只有 0 和 1 两个值,因此具有传输可靠、易于存储、抗干扰能力强、稳定性好等优点,使得数字电路在众多领域中获得了越来越广泛的应用。从 20 世纪 70 年代以来,由于电子器件制作工艺的长足进步,数字电路从小规模集成电路 (SSI) 发展成为中规模集成电路 (MSI)、大规模集成电路 (LSI) 和超大规模集成电路 (VLSI),目前已经进入系统集成 (SOC) 阶段,并向低功耗、高速度、可编程、可测试和多值化方向发展。



(a) 电平型数字信号



(b) 脉冲型数字信号

图 1.1.2 数字信号传输波形

1.2 数制与码制

数字逻辑系统的分析和设计与数和字符的表示紧密相关。因此，数制和码制是学习数字电路的重要基础。

1.2.1 数制 (Number System)

数制是人类表示数值大小的各种方法的统称。一般是将数划分为不同的数位来表示，当某一数位累计到一定数量之后，该位又从零开始，同时向高位进位，所以也称为进位计数制。数制可以用少量的数码来表示较大的数，因而被广泛应用。

在数制中，任意一个数 N 可以用多项式来表示，即按位权展开式来表示：

$$N = K_{n-1}R^{n-1} + K_{n-2}R^{n-2} + \cdots + K_0R^0 + K_{-1}R^{-1} + \cdots + K_{-m}R^{-m}$$

$$= \sum_{i=-m}^{n-1} K_i R^i \quad (1.2.1)$$

式中 R 为数制的基数； n 为整数部分的位数； m 为小数部分的位数； K_i 为第 i 位的系数； R^i 为第 i 位的权值，其中第 i 位的权值是指第 i 位上数码为 1 时所表示的数值，也简称为“权”，基数 R 是指一个数制中所包含基本数码的个数。如 R 为 10，则采用 0、1、2、3、4、5、6、7、8、9 十个数码来表示任意数；又如 R 为 8，则采用 0、1、2、3、4、5、6、7 八个数码来表示任意数。数制基数取值为 R ，该数制就称为 R 进制。根据 R 的不同取值，常用的数制有十进制、二进制、十六进制和八进制。

一、十进制 (Decimal)

基数 R 为 10 的数制称为十进制。十进制有 0~9 共十个数码，低位向邻近高位进位的规则是逢十进一。十进制数一般用 $(N)_D$ 或 $(N)_{10}$ 表示，如 $(53)_D$ 、 $(91)_{10}$ 等。

$$(N)_D = \sum_{i=-m}^{n-1} K_i \times 10^i \quad (1.2.2)$$

$$(91)_{10} = 9 \times 10^1 + 1 \times 10^0$$

十进制是人们生活中最常用的数制，但十进制数有 10 个数码，在计算机等数字设备中实现起来比较困难。

二、二进制 (Binary)

基数 R 为 2 的数制称为二进制。二进制只有 0 和 1 两个数码，低位向邻近高位进位的规则是逢二进一。二进制数一般用 $(N)_B$ 或 $(N)_2$ 表示，如 $(11001)_B$ 、 $(1011)_2$ 等。

$$(N)_B = \sum_{i=-m}^{n-1} K_i \times 2^i \quad (1.2.3)$$

$$(1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

目前，数字设备中所用的数字电路通常只有低电平和高电平两个状态，而二进制的基数也只有两个数码，所以数字系统中用得最多的是二进制数。但采用二进制表示一个数时数位过多，所以也常用与二进制有简单关系的十六进制和八进制来计数。

三、十六进制 (Hexadecimal)

基数 R 为 16 的数制称为十六进制。十六进制有 0、1、2、3、4、5、6、7、8、9、A(10)、B(11)、C(12)、D(13)、E(14)、F(15) 共十六个数码，低位向邻近高位进位的规则是逢十六进一。十六进制数一般用 $(N)_H$ 或 $(N)_{16}$ 表示，如 $(9E)_H$ 、 $(C2)_{16}$ 等。

$$(N)_H = \sum_{i=-m}^{n-1} K_i \times 16^i \quad (1.2.4)$$

$$(C2)_{16} = 12 \times 16^1 + 2 \times 16^0$$

四、八进制 (Octal)

基数 R 为 8 的数制称为八进制。八进制有 0~7 共 8 个数码，低位向邻近高位进位的规则是逢八进一。八进制数一般用 $(N)_O$ 或 $(N)_8$ 表示，如 $(167)_O$ 、 $(752)_8$ 等。

$$(N)_O = \sum_{i=-m}^{n-1} K_i \times 8^i \quad (1.2.5)$$

$$(752)_8 = 7 \times 8^2 + 5 \times 8^1 + 2 \times 8^0$$

1.2.2 数制间转换

一个数可以用不同数制来表示。十进制是人们熟悉和惯用的数制，但不利于数字逻辑系统使用。一般数字逻辑系统中常用的是二进制数和十六进制数，因为它们实现简单，便于存储和传输。因此，常常需要在不同数制之间进行转换。

一、二进制与十进制间转换

将二进制数转换为十进制数只需按位权展开式展开，且以十进制运算规则进行计算，就得到相应的十进制数。

[例 1.2.1] 将 $(11001.01)_B$ 转换为十进制数。

$$\begin{aligned} \text{解: } (11001.01)_B &= (1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}) \\ &= 16 + 8 + 0 + 0 + 1 + 0 + 0.25 \\ &= (25.25)_D \end{aligned}$$

将十进制数转换为二进制数要分别对整数部分和小数部分进行转换。进行整数部分转换时，先将十进制数除以 2，再对每次得到的商除以 2，直到商等于 0。然后将



各次余数按倒序写出,即得到二进制数的整数部分。进行小数部分转换时,先将十进制数乘以 2,积的整数作为相应二进制数的小数部分,再对积的小数部分乘以 2。以此类推,直到积的小数部分为 0。然后将各次积的整数按顺序写出,即得到二进制数的小数部分。

[例 1.2.2] 将 $(13.125)_D$ 转换为二进制数。

解:先转换整数部分:

	余数
$\frac{13}{2} = 6$	1
$\frac{6}{2} = 3$	0
$\frac{3}{2} = 1$	1
$\frac{1}{2} = 0$	1

所以对应二进制数的整数部分为 $(1101)_B$;

再转换小数部分:

	积的整数
$0.125 \times 2 = 0.25$	0
$0.25 \times 2 = 0.5$	0
$0.5 \times 2 = 1.00$	1

所以对应二进制数的小数部分为 $(0.001)_B$;

综合以上, $(13.125)_D$ 转换为二进制数为 $(1101.001)_B$ 。

二、十六进制与二进制间转换

将十六进制数转换为二进制数,就是将十六进制数的每一位展开成 4 位二进制数。

[例 1.2.3] 将十六进制数 $(2EF.6)_H$ 转换为二进制数。

解:十六进制数

2 E F . 6

二进制数

$\overbrace{0010} \overbrace{1111} \overbrace{1011} \overbrace{1111} . \overbrace{0110}$

所以十六进制数 $(2EF.6)_H$ 转换为二进制数为 $(1011101111.011)_B$ 。

将二进制数转换为十六进制数,整数部分从右向左每 4 位一组,每组用 1 位十六进制数代替,不足 4 位用 0 补足;小数部分则是从左向右每 4 位一组,每组用 1 位十六进制数代替,不足 4 位用 0 补足。

[例 1.2.4] 将二进制数 $(1011010011.101)_B$ 转换为十六进制数。

解:二进制数

$\underbrace{0010} \underbrace{1101} \underbrace{0011} . \underbrace{1010}$

十六进制数

2 D 3 . A

所以二进制数 $(1011010011.101)_B$ 转换为十六进制数为 $(2D3.A)_H$ 。



三、十六进制与十进制间转换

将十六进制数转换为十进制数只需按位权展开式展开，以十进制运算规则进行计算，就得到相应的十进制数。

[例 1.2.5] 将十六进制数 $(B51.C)_H$ 转换为十进制数。

$$\begin{aligned} \text{解: } (B51.C)_H &= 11 \times 16^2 + 5 \times 16^1 + 1 \times 16^0 + 12 \times 16^{-1} \\ &= (2897.75)_D \end{aligned}$$

将十进制数转换为十六进制数要分别对整数部分和小数部分进行转换。进行整数部分转换时，先将十进制数除以 16，再对每次得到的商除以 16，直到商等于 0。然后将各次余数按倒序写出，即得到十六进制数的整数部分。进行小数部分转换时，先将十进制数乘以 16，积的整数作为相应十六进制数的小数部分，再对积的小数部分乘以 16。以此类推，直到小数部分为 0。然后将积的整数按顺序写出，即得到十六进制数的小数部分。

[例 1.2.6] 将 $(287.62890625)_D$ 转换为十六进制数。

解： 先转换整数部分：

$$\begin{array}{rcl} & & \text{余数} \\ \frac{287}{16} & = 17 & \text{F} \\ \frac{17}{16} & = 1 & 1 \\ \frac{1}{16} & = 0 & 1 \end{array}$$

所以对应十六进制数的整数部分为 $(11F)_H$ ；

再转换小数部分：

$$\begin{array}{rcl} & & \text{积的整数} \\ 0.62890625 \times 16 & = 10.0625 & \text{A} \\ 0.0625 \times 16 & = 1 & 1 \end{array}$$

所以对应十六进制数的小数部分为 $(0.A1)_H$ ；

综上所述，将 $(287.62890625)_D$ 转换为十六进制数为 $(11F.A1)_H$ 。常用数制十进制、二进制和十六进制之间的对应关系如表 1.2.1 所示。

表 1.2.1 常用数制对应关系

十进制	二进制	十六进制
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6

(续表)

十进制	二进制	十六进制
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

1.2.3 码制

在数字系统中,除了利用数制来描述用于计算的数值之外,还常将日常生活中具有特定意义的信息(如文字、符号等)用一种规定的二进制数码表示,称为码制。常用的码制有BCD码、格雷码、ASCII码等。

一、BCD码

在数字计算机中,十进制数除了转换成二进制数外,还可以用十进制数码表示,称为BCD码。它具有二进制数的形式,却有十进制数的特点,是用4位二进制数来表示十进制数中0~9共10个数码的。由于4位二进制数码可以表示16种不同的组合状态,根据不同的选取方式,可以得到不同的BCD码,如表1.2.2所示。

表 1.2.2 几种常用BCD码

十进制数	8421码	5421码	2421码	余3码
0	0000	0000	0000	0011
1	0001	0001	0001	0100
2	0010	0010	0010	0101
3	0011	0011	0011	0110
4	0100	0100	0100	0111
5	0101	1000	1011	1000
6	0110	1001	1100	1001
7	0111	1010	1101	1010
8	1000	1011	1110	1011
9	1001	1100	1111	1100

1. 8421码

8421码为有权码,各位的权值分别为8,4,2,1。虽然8421码的权值与4位二进制数码的权值相同,但8421只取用了4位自然二进制数码的前10种组合。

2. 5421 码

5421 码也为有权码，各位的权值分别为 5，4，2，1，其显著特点是最高位连续 5 个 0 后再连续 5 个 1。当计数器采用这种编码时，最高位可产生对称方波输出。

3. 2421 码

2421 码也是有权码，各位的权值分别为 2，4，2，1。其特点是码制中的 0 和 9，1 和 8，2 和 7，3 和 6，4 和 5，各对组码相加均为 1111。

4. 余 3 码

余 3 码是 8421 码的每个码组加 0011 形成的，是一种无权码。其特点是码制中的 0 和 9，1 和 8，2 和 7，3 和 6，4 和 5，各对组码相加也均为 1111。

二、格雷码 (Gray)

格雷码又称循环码、反射码，是一种无权码。它利用了 4 位二进制数码的 16 种组合，其特点是任意两个相邻代码间只有 1 位数码不同，避免了逻辑错误，利于转换和传输。4 位格雷码如表 1.2.3 所示。

表 1.2.3 4 位格雷码

十进制数	格雷码	十进制数	格雷码
0	0000	8	1100
1	0001	9	1101
2	0011	10	1111
3	0010	11	1110
4	0110	12	1010
5	0111	13	1011
6	0101	14	1001
7	0100	15	1000

三、ASCII 码

ASCII 码是美国信息交换标准码制 (American Standard Code for Information Inter-change) 的简称，是目前国际上最通用的一种字符码。它采用 7 位二进制数码表示十进制数、英文大小写字母、运算符、控制符以及特殊符号，如表 1.2.4 所示。

表 1.2.4 ASCII 码

高位 \ 低位								
	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u

(续表)

高位\低位	000	001	010	011	100	101	110	111
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O		o	DEL

ASCII 码包含 52 个大、小写英文字母，10 个十进制数字字符，32 个标点符号、运算符号、特殊号，还有 34 个不可显示打印的控制字符码，一共是 128 个码，正好可以用 7 位二进制数进行编码。有的数字计算机系统也使用由 8 位二进制数编码的扩展 ASCII 码，其前 128 个是标准的 ASCII 码字符编码，后 128 个是扩充的字符编码。表 1.2.4 中一些控制符的含义如表 1.2.5 所示。

表 1.2.5 7 位 ASCII 码字符对照表

字符	含义	字符	含义
NUL	Null 空白	DC1	Device Control 1 设备控制 1
SOH	Start of Heading 标题开始	DC2	Device Control 2 设备控制 2
STX	Start of Text 正文开始	DC3	Device Control 3 设备控制 3
ETX	End of Text 正文结束	DC4	Device Control 4 设备控制 4
EOT	End of Transmission 传输结束	NAK	Negative Acknowledge 否认
ENQ	Enquiry 询问	SYN	Synchronous Idle 同步空传
ACK	Acknowledge 确认	ETB	End of Transmission Block 块结束
BEL	Bell 响铃(警告)	CAN	Cancel 取消
BS	BackSpace 退一格	EM	End of Medium 纸尽
HT	Horizontal Tabulation 水平列表	SUB	Substitute 替换
LF	Line Feed 换行	ESC	Escape 脱离
VT	Vertical Tabulation 垂直列表	FS	File Separator 文件分离符
FF	From Feed 走纸	GS	Group Separator 字组分离符
CR	Carriage Return 回车	RS	Record Separator 记录分离符
SO	Shift Out 移出	US	Unit Separator 单元分离符
SI	Shift Out 移入	SP	Space 空格
DLE	Data Link Escape 数据链路换码	DEL	Delete 删除