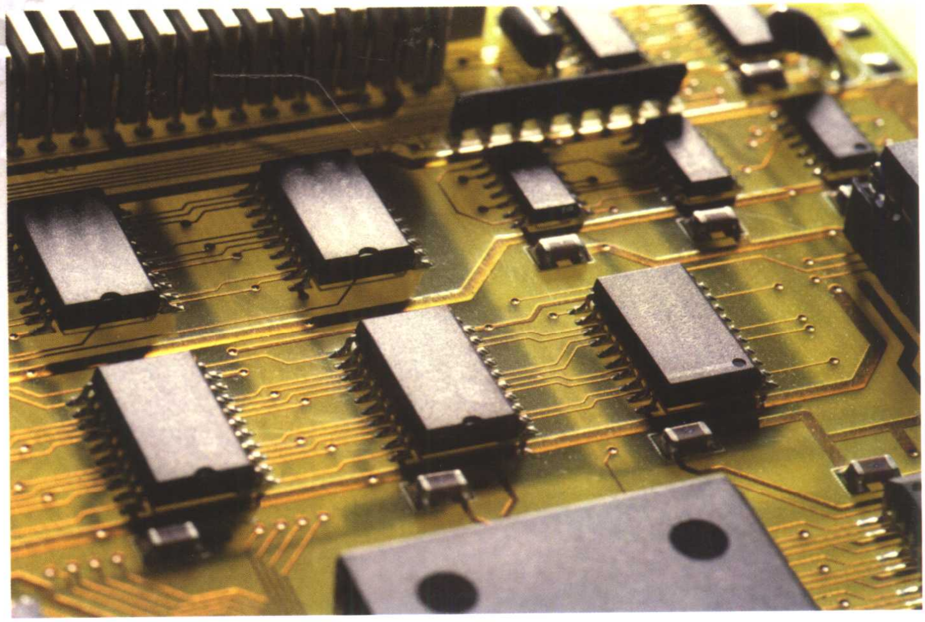
SHUZI DIANLUJICHU

# 数字电路基础

刘莉宏 主编



机械工业出版社

# 数 字 电 路 基 础

主　编　刘莉宏
副主编　高　青
参　编　张　敏　冯光丽
　　　　梁如军　吕　燕
主　审　丁景红

机 械 工 业 出 版 社

本书着重数字电路的基本知识、基本原理和基本方法的讲述，内容由浅入深，通俗易懂，注重理论联系实际，具有较强的教学实用性。书中每章都有小结，另有《数字电路基础练习册》与之配套。

本书可作为高等职业教育工科电子、通信、计算机类专业及相关专业的通用教材，也可作为从事电子技术工作的高等专业技术人员的参考书和中专学校相关专业的教材。

**图书在版编目（CIP）数据**

数字电路基础/刘莉宏主编 .—北京：机械工业出版社，2000.3
ISBN 7-111-07826-8

Ⅰ.数… Ⅱ.刘… Ⅲ.数字电路-专业学校-教材 Ⅳ.TH79-44

中国版本图书馆 CIP 数据核字（2000）第 01594 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）
责任编辑：李艳霞 版式设计：陈 伟 责任校对：吴美英
封面设计：东 方 责任印制：侯新民
北京京丰印刷厂印刷·新华书店北京发行所发行
2000 年 9 月第 1 版·2003 年 1 月第 3 次印刷
787mm×1092mm 1/16·11.25 印张·268 千字
7 001—10 000 册
定价：17.00 元

# 前　　言

《数字电路基础》是工科电子、通信、计算机等电类专业必修的技术基础课程。按照职业教育的培养目标，遵循理论教学必需够用的原则，编写了这本教材，供各高、中等职业教育工科电类专业及相关专业使用。

《数字电路基础》课的任务是教给学生有关数字电路的基本知识、基本原理、逻辑部件及中规模集成电路的分析方法、设计方法和应用举例，并使学生对数字电路有一个整体认识，为今后学习、使用和维修数控装置、电子计算机及数字式电子仪器和仪表等数字电路的应用打下一定的基础。

本书的主要内容有：数字电路基础知识、集成门电路、组合逻辑电路、集成触发器、时序逻辑电路、集成定时器及应用、D/A 与 A/D 转换电路、数字电路应用举例。全书教学时数为 60～70 学时，实验时数为 20 学时以上，学时较少时可适当删去一些章节。每章后有小结，并另有《数字电路基础练习册》与本书配套。

参加本书编写的有北京钢铁学校、北京电子工业学校、北京机械工业学校、北京信息职业技术学院、北京工业职业技术学院。全书由刘莉宏主编，高青副主编，丁景红主审。

本书在编写过程中，得到了北京市电类基础课研究会和各相关学校的热情支持和帮助，在此表示衷心的感谢。

由于编者水平有限，书中难免存在错误和不当之处，恳切希望读者批评指正。

编者

2000.4

# 目　　录

# 绪　　论

## 一、数字信号与数字电路

在电子工程中，按照所处理的信号形式，电子线路通常分为模拟电路和数字电路两大类。

模拟信号是指模拟物理量的信号形式。它是一种在时间上、数值上均连续的信号，可以在一定范围内任意取值。如：温度、压力、水位、电压、电流等信号，它们都是模拟信号。

数字信号是指在时间上、数值上均离散的信号，即它的变化在时间上不连续且它的数值大小及变化都采用数字形式。

模拟电路处理的是模拟信号，数字电路处理的是数字信号。

数字电路与模拟电路相比较，主要有以下特点：晶体管在数字电路中主要工作在饱和或截止状态；在数字电路中，电压、电流通常只有两种状态，即高电平或低电平、有电流或无电流，这样两种状态可以用逻辑“1”及逻辑“0”表示，因此计数进位制采用的是二进制；数字电路研究的重点是各个基本单元电路之间的状态关系，而不是数值关系；数字电路采用的分析和设计方法是逻辑分析和逻辑设计，其数学工具是逻辑代数。

随着电子技术的迅猛发展，特别是集成电路的发展，数字技术的应用已极为广泛，主要体现在以下几个方面：在数字通信系统中，图像、声音等信号都可以用若干个“0”和“1”编制成各种代码，分别代表不同的信息含义，以实现信息的传递；在自动控制系统中，可以利用数字电路的逻辑功能设计出各种各样的数字控制装置；在测量仪表中，可以利用数字电路对测量信号进行处理，并将测量结果用十进制数码表示出来；在数字电子计算机中，可以利用数字电路实现各种功能的数字信号处理。

## 二、脉冲信号与脉冲电路

脉冲信号是指持续时间极短的跃变的电信号。广义上讲，一切非直流又非正弦交流的电信号统称为脉冲信号。常见的脉冲波形如图 0-1 所示。这些脉冲波形都是时间的函数。

实际的脉冲波形与图 0-1 所示的理想波形是稍有差异的。图 0-2 所示为矩形脉冲信号

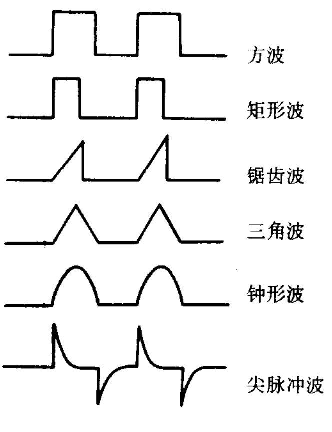



图 0-1　常见的几种脉冲波形

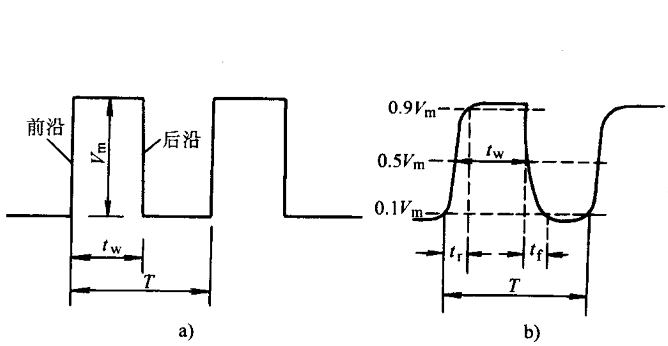



图 0-2　矩形脉冲波形

a) 理想矩形波　b) 实际矩形波

的理想波形和实际波形。现以它为例来介绍脉冲的主要参数。

1. 脉冲幅度 $V_m$

指一个脉冲信号从底部到顶部之间的变化量。它是表示脉冲信号强弱的重要参数。

2. 脉冲上升时间 $t_r$

又称前沿，指脉冲从 $0.1V_m$ 上升到 $0.9V_m$ 所需要的时间。$t_r$ 越短，脉冲前沿越陡，就越接近理想波形。

3. 脉冲下降时间 $t_f$

又称后沿，指脉冲从 $0.9V_m$ 下降到 $0.1V_m$ 所需要的时间。

4. 脉冲宽度 $t_w$

指脉冲持续时间。通常指脉冲 $0.5V_m$ 处从前沿到后沿的时间间隔。脉冲持续的时间极短，通常用毫秒（$1ms=10^{-3}s$）、微秒（$1\mu s=10^{-6}s$）或纳秒（$1ns=10^{-9}s$）来表示。

5. 脉冲周期 $T$

在一个周期性重复变化的脉冲序列中，两个相邻脉冲的前沿（或后沿）之间的时间间隔，或两个相邻脉冲重复出现的时间间隔。

6. 脉冲频率 $f$

指单位时间内脉冲信号重复出现的次数。用脉冲周期的倒数来表示。

$$f=\frac{1}{T}$$

7. 占空比 $D$

指脉冲宽度与脉冲周期之比。

$$D=\frac{t_w}{T}$$

占空比 $D=1/2$ 的矩形波为方波。

用来处理脉冲信号的电路称为脉冲电路。它可以用分立晶体管、场效应晶体管作为开关，与 $RC$ 或 $RL$ 电路构成，也可以由集成逻辑门电路或集成运算放大器和 $RC$ 电路构成。常见的脉冲电路有用来产生脉冲波形以及对脉冲波形进行整形、变换等电路。

脉冲技术已广泛用于雷达、电视、数字通信、测量仪表、自动控制和电子设备中。

**三、脉冲电路与数字电路的关系**

从信号上讲，当数字电路的基本单元不断地在“0”和“1”两种状态之间快速转换时，便产生一系列的矩形脉冲信号，故数字电路是一种脉冲电路，也可以说数字电路是以脉冲电路为基础的一个分支。

从电路分析、研究的内容和方法上讲，数字电路与脉冲电路是不属于同一范畴的电路。数字电路分析的重点是电路的逻辑功能，分析的方法是逻辑分析，它所用的数学工具是逻辑代数；脉冲电路分析的重点是电路输入、输出波形的形状、幅度和周期等，分析方法采用模拟电路的分析方法，即频域法和时域法。因此，数字电路与脉冲电路既有关联，又有区别。

**四、数字电路中的一些规定**

在日常生活和工作中，我们经常会遇到事件的两个相对的方面或状态，称为逻辑状态。例如，灯泡的亮与灭等。我们要用数学推理方法解决这些事件时，常把事件的两种状态分别用逻辑符号“1”和“0”表示。这里的“1”和“0”并不是表示数值的大小，而是作为一种

符号，以表示事件的两种逻辑状态。我们把它们称为逻辑“1”和逻辑“0”，以区别于数字符号的 1 和 0。例如，灯泡的“亮”与“灭”这两种逻辑状态，分别用“1”和“0”表示。

总之，一种逻辑状态要用“1”表示，而另一种逻辑状态就要用“0”表示。

在数字电路中，大于零的脉冲信号称为正脉冲，小于零的脉冲信号称为负脉冲。电位的高低常用电平表示。对于规定的零电平来说，高电位对应高电平，低电位对应低电平。电平的高和低均可用逻辑“1”或逻辑“0”来表示，这样就有两种表示方法，即用逻辑“1”表示高电平，用逻辑“0”表示低电平，这是正逻辑体制；用逻辑“1”表示低电平，用逻辑“0”表示高电平，这是负逻辑体制。

正逻辑和负逻辑是人为规定的。同一电路既可以采用正逻辑体制，也可以采用负逻辑体制，电路的本身并不因为正、负逻辑的规定而有所改变。但对于同一电路来说，从正逻辑和负逻辑的角度去分析它的逻辑关系时，结论却是截然不同的。

本书若不加以说明，一般都采用正逻辑体制。在实际电路中，有时既用正逻辑又用负逻辑，即使用的是混合逻辑。

数字电路是一门实践性很强的专业基础课，除要掌握其基本原理、基本方法外，更重要的是要灵活运用。因此，在学习中要完成一定数量的习题，还要密切配合实验、课程设计等实践课程。只有这样，才能掌握本课程的基本知识，掌握分析和设计的基本方法，培养实际动手能力及灵活应用的能力。

# 第一章　数字电路基础知识

## 第一节　数制与码制

数制即计数的方法。在不同的数制中，数的进位方式和计数方法就各不相同。在日常生活中，人们习惯用十进制数，而在数字电路中则更多的是采用二进制数、八进制数、十六进制数等。

**一、几种常用数制**

1. 十进制

人们都熟悉十进制数，它是由 0、1、2……9 十个数字符号表示的，这些数字符号称为“数码”。如：2345 意为 2000 + 300 + 40 + 5 可写成

$$2345 = 2\times10^3 + 3\times10^2 + 4\times10^1 + 5\times10^0$$

式中的 10 称为数制的基数，也是一个数制中数码的个数，即表示逢十进一。$10^3$、$10^2$、$10^1$、$10^0$ 称之为“位权”。十进制数的各个数位的位权值是 10 的幂。再如：123.45 可写成

$$123.45 = 1\times10^2 + 2\times10^1 + 3\times10^0 + 4\times10^{-1} + 5\times10^{-2}$$

对于任意一个十进制数 $(N)_{10}$，如果它有 $n$ 位整数和 $m$ 位小数，都可以按位权展开为

$$\begin{aligned}(N)_{10} &= a_{n-1}a_{n-2}\cdots a_1a_0 \cdot a_{-1}a_{-2}\cdots a_{-m}\\ &= a_{n-1}\times10^{n-1} + a_{n-2}\times10^{n-2} + \cdots + a_1\times10^1 + a_0\times10^0 +\\ &\quad a_{-1}\times10^{-1} + a_{-2}\times10^{-2} + \cdots + a_{-m}\times10^{-m}\\ &= \sum_{i=-m}^{n-1} a_i\times10^i \end{aligned} \tag{1-1}$$

式中　$a_i$——第 $i$ 位的十进制数码；

$10^i$——第 $i$ 位的位权；

$(N)_{10}$——下标 10 表示十进制数；

n——二进制数整数位数；

m——二进制数小数位数。

在数字电路中是通过电路的不同状态表示数码的。要使电路具有十个不同状态来表示 0 ~9 十个数码，则相当困难，比较容易实现的是两种不同状态，如用电路的“通”和“断”电位的“高”和“低”等等来表示“0”和“1”两个数码。

2. 二进制

二进制数只有 0 和 1 两个数码，即基数为 2，其计数规律是逢二进一，各个数位的位权值是 2 的幂。

对于任意一个二进制数 $(N)_2$ 也可按其位权展开为

$$(N)_2 = a_{n-1}\times2^{n-1} + a_{n-2}\times2^{n-2} + \cdots + a_1\times2^1 + a_0\times2^0 + a_{-1}\times2^{-1} + \cdots + a_{-m}\times2^{-m}$$

$$= \sum_{i=-m}^{n-1} a_i \times 2^i \tag{1-2}$$

式中　$a_i$——第 $i$ 位的二进制数码；

$2^i$——第 $i$ 位的位权；

$(N)_2$——下标 2 表示二进制数。

例如，二进制数 1011.01 可按位权展开为

$$(1011.01)_2 = 1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 1\times 2^0 + 0\times 2^{-1} + 1\times 2^{-2}$$

上述按位权展开的表示方法可以推广到任意进制的计数制。对一个计数制基数为 $R$ 的 $R$ 进制，共有 0、1、…、$(R-1)$，$R$ 个不同的数码，一个 $R$ 进制的数按位权可展开为

$$\begin{aligned}(N)_R &= a_{n-1}a_{n-2}\cdots a_1 a_0 . a_{-1}a_{-2}\cdots a_{-m}\\ &= a_{n-1}\times R^{n-1} + a_{n-2}\times R^{n-2} + \cdots + a_1\times R^1 + a_0\times R^0 + a_{-1}\\ &\quad \times R^{-1} + a_{-2}\times R^{-2} + \cdots + a_{-m}\times R^{-m}\\ &= \sum_{i=-m}^{n-1} a_i \times R^i \end{aligned}\tag{1-3}$$

式中　$a_i$——$R$ 进制数中的第 $i$ 位数码；

$R^i$——第 $i$ 位的位权。

3．八进制

八进制数有 0、1、2、…7 八个数码，它的基数为 8，各个数位的位权值是 8 的幂，其计数规律遵循逢八进一的原则。

按式（1－3）可将八进制数展成位权表达式。例如八进制数 $(257)_8$ 按位权展开为

$$(257)_8 = 2\times 8^2 + 5\times 8^1 + 7\times 8^0$$

4．十六进制

十六进制的基数是 16。习惯上采用 0～9、A、B、C、D、E、F 组成十六个数码。各个数位的位权值是 16 的幂。计数规律遵循逢十六进一原则。同八进制一样可按式（1－3）展成位权表达式。

为了表示各种数制的对应关系，表 1－1 列出了常用数制的计数方法。

**表 1－1　常用计数进制对照表**

| 十进制数 | 二进制数 | 八进制数 | 十六进制数 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |
| 2 | 10 | 2 | 2 |
| 3 | 11 | 3 | 3 |
| 4 | 100 | 4 | 4 |
| 5 | 101 | 5 | 5 |
| 6 | 110 | 6 | 6 |
| 7 | 111 | 7 | 7 |
| 8 | 1000 | 10 | 8 |

（续）

| 十进制数 | 二进制数 | 八进制数 | 十六进制数 |
|---|---|---|---|
| 9 | 1001 | 11 | 9 |
| 10 | 1010 | 12 | A |
| 11 | 1011 | 13 | B |
| 12 | 1100 | 14 | C |
| 13 | 1101 | 15 | D |
| 14 | 1110 | 16 | E |
| 15 | 1111 | 17 | F |
| 16 | 10000 | 20 | 10 |
| 17 | 10001 | 21 | 11 |
| 18 | 10010 | 22 | 12 |

## 二、不同进制数的相互转换

同一个数在不同进制间的相互转换是根据两个有理数相等，即两个数的整数部分一定相等，小数部分也一定相等的原则进行的。

1. 二进制数与十进制数的转换

(1) 二进制数转换成十进制数

将二进制数转换成十进制数，只需把二进制数按位权展开式展开，然后求出各项和，其结果即为相应的十进制数。

〔**例 1-1**〕 将二进制数 $(1101)_2$ 转换成十进制数

**解**：$(1101)_2 = 1\times2^3+1\times2^2+0\times2^1+1\times2^0=8+4+0+1$

$= (13)_{10}$

按位权展开式运算的转换方法，适用于整数、纯小数和带小数。

〔**例 1-2**〕 将二进制数 $(11010.011)_2$ 转换成十进制数。

**解**：$(11010.011)_2 = 1\times2^4+1\times2^3+0\times2^2+1\times2^1+0\times2^0+0\times2^{-1}+1\times2^{-2}+1\times2^{-3}$

$=16+8+0+2+0+0+0.25+0.125$

$=(26.375)_{10}$

(2) 十进制数转换成二进制数

将十进制数转换成二进制数一般采用基数乘除法。

十进制整数转换成二进制整数，采用逐次除以 2 的方法，其步骤如下：

第一步，把给出的十进制整数除以 2，余数 0 或 1 表示二进制数的最低位。

第二步，把前一步所得之商再除以 2，余数 0 或 1 表示二进制数的次低位。

第三步及以后各步，继续除以 2，记下余数，直到最后相除之商为 0 为止。最后得到的余数表示二进制数的最高位。

〔**例 1-3**〕 把十进制数 $(53)_{10}$转换成二进制数。

**解：**

$$\begin{array}{r|l l}
2 & \underline{53} & \\
2 & \underline{26} & —— 1 \\
2 & \underline{13} & —— 0 \\
2 & \underline{6} & —— 1 \\
2 & \underline{3} & —— 0 \\
2 & \underline{1} & —— 1 \\
 & 0 & —— 1
\end{array}
\quad \uparrow \text{读数方向}$$

结果为　$(53)_{10}=(110101)_2$

十进制纯小数转换成二进制纯小数采用逐次乘以 2 的方法，其步骤如下：

第一步，把给出的十进制纯小数乘以 2，其结果或者是≥1，或者是<1。如果是≥1 则记 1，如果是<1，则记 0。所记之数作为二进制小数的最高位。

第二步，把前一步所得结果的小数部分再乘以 2，同第一步方法，根据运算结果记下 0 或者 1，把它作为二进制小数的次高位。

第三步及以后各步，重复第二步，当乘以 2 以后小数部分变为 0 时，运算完成，或者已达到要求的小数精度时，运算结束。

〔**例 1－4**〕　把十进制数 $(0.375)_{10}$转换成二进制数。

**解：**

$$\begin{array}{r l l}
 & 0.375 & \\
\times & 2 & \\
\hline
 & 0.750 & —— 0 \quad a_{-1} \\
 & 0.750 & \\
\times & 2 & \\
\hline
 & 1.500 & —— 1 \quad a_{-2} \\
 & 0.500 & \\
\times & 2 & \\
\hline
 & 1.000 & —— 1 \quad a_{-3}
\end{array}
\quad \downarrow \text{读数方向}$$

结果为　$(0.375)_{10}=(0.011)_2$

对于带小数的十进制数转换成二进制数，可按上述方法分别进行整数部分和小数部分的转换，最后将它们加在一起即可。

〔**例 1－5**〕　将十进制数 $(97.68)_{10}$转换为二进制数（小数点后保留四位）。

**解：**

$$\begin{array}{r|l l}
2 & \underline{97} & \\
2 & \underline{48} & —— 1 \\
2 & \underline{24} & —— 0 \\
2 & \underline{12} & —— 0 \\
2 & \underline{6} & —— 0 \\
2 & \underline{3} & —— 0 \\
2 & \underline{1} & —— 1 \\
 & 0 & —— 1
\end{array}
\quad \uparrow \text{读数方向}$$

$$\begin{array}{r l l}
0.68 & & \\
\times\quad 2 & & \\
\hline
1.36 & —— 1 & a_{-1} \\
0.36 & & \\
\times\quad 2 & & \\
\hline
0.72 & —— 0 & a_{-2} \\
0.72 & & \\
\times\quad 2 & & \\
\hline
1.44 & —— 1 & a_{-3} \\
0.44 & & \\
\times\quad 2 & & \\
\hline
0.88 & —— 0 & a_{-4}
\end{array} \quad \downarrow \text{读数方向}$$

将 0.88 四舍五入，则 $a_{-4} \approx 1$

结果为 $(97.68)_{10} \approx (1100001.1011)_2$

2. 八进制数与十进制数的转换

八进制数与十进制数的相互转换，可仿照二进制数与十进制数的转换方法。

**〔例 1－6〕** 将八进制数 $(257)_8$ 转换成十进制数

**解：**

$$\begin{aligned}
(257)_8 &= 2 \times 8^2 + 5 \times 8^1 + 7 \times 8^0 \\
&= 128 + 40 + 7 \\
&= (175)_{10}
\end{aligned}$$

**〔例 1－7〕** 将十进制数 $(84.6875)_{10}$ 转换成八进制数。

**解：** 整数部分

$$\begin{array}{r l l}
8 \,\underline{|\,84} & & \\
8 \,\underline{|\,10} & — & 4 \\
8 \,\underline{|\,1} & — & 2 \\
0 & — & 1
\end{array} \quad \uparrow \text{读数方向}$$

小数部分

$$\begin{array}{r l l}
0.6875 & & \\
\times\quad 8 & & \\
\hline
5.5000 & —— 5 & a_{-1} \\
0.5000 & & \\
\times\quad 8 & & \\
\hline
4.0000 & —— 4 & a_{-2}
\end{array} \quad \downarrow \text{读数方向}$$

结果为 $(84.6875)_{10} = (124.54)_8$

3. 十六进制数与十进制数的转换

十六进制数与十进制数的相互转换仍可仿照二进制数与十进制数的转换方法。

**〔例 1－8〕** 将十六进制数 $(2E.A)_{16}$ 转换成十进制数。

**解：**

$$\begin{aligned}
(2E.A)_{16} &= 2 \times 16^1 + E \times 16^0 + A \times 16^{-1} \\
&= 2 \times 16 + 14 \times 1 + 10 \times 0.0625 \\
&= (46.625)_{10}
\end{aligned}$$

**〔例 1－9〕** 将十进制数 $(54.48)_{10}$ 转换成十六进制数（小数点后保留三位）。

**解**：整数部分

$$\begin{array}{r|l} 16 & \underline{54} \\ 16 & \underline{3} \quad —6 \\ & 0 \quad\ —3 \end{array} \quad \uparrow \text{读数方向}$$

小数部分

$$\begin{array}{r} 0.48 \\ \times\ \ 16 \\ \hline 7.68 \end{array} \ —7 \quad a_{-1}$$

$$\begin{array}{r} 0.68 \\ \times\ \ 16 \\ \hline 10.88 \end{array} \ —10 \quad a_{-2}$$

$$\begin{array}{r} 0.88 \\ \times\ \ 16 \\ \hline 14.08 \end{array} \ —14 \quad a_{-3} \qquad \downarrow \text{读数方向}$$

结果为　$(54.48)_{10} \approx (36.7AE)_{16}$

4. 八进制数与二进制数的转换

(1) 八进制数转换成二进制数

由于八进制数的基数 8 是二进制基数 2 的三次幂。所以转换很方便，只要将八进制数的每一位转换为三位二进制的等值数即可完成八进制数向二进制数的转换。

〔**例 1-10**〕　将八进制数 $(23)_8$ 转换成二进制数。

**解**：

$$\begin{array}{cc} 2 & 3 \\ \downarrow & \downarrow \\ 010 & 011 \end{array}$$

结果为　$(23)_8 = (10011)_2$

(2) 二进制数转换成八进制数

若将二进制数转换成八进制数，只要将二进制数从小数点开始每三位编一组，两端不够三位时，可添零补足三位，然后把每一组转换成一位八进制等值数即可。

〔**例 1-11**〕　将二进制数 $(1011.01101)_2$ 转换成八进制数。

**解**：
$$\begin{aligned} (1011.01101)_2 &= (001,\ 011.011,\ 010)_2 \\ &= (13.32)_8 \end{aligned}$$

5. 十六进制数与二进制数的转换

十六进制的基数是 2 的四次幂，因此十六进制数与二进制数的互换和八进制数与二进制数互换的方法类似。

(1) 十六进制数换转成二进制数

十六进制数转换为二进制数，只需将每位十六进制数用四位二进制数表示即可。

〔**例 1-12**〕　将十六进制数 $(51A.2)_{16}$转换成二进制数。

**解**：

$$\begin{array}{ccccc} 5 & 1 & A & . & 2 \\ \downarrow & \downarrow & \downarrow & & \downarrow \\ 0101 & 0001 & 1010 & & 0010 \end{array}$$

结果为　$(51A.2)_{16} = (10100011010.001)_2$

(2) 二进制数转换成十六进制数

二进制数转换成十六进制数，将二进制数由小数点开始每四位一组，最高组和最低组不足四位的补零，然后把每一组转换成一位十六进制等值数即可。

**〔例 1-13〕** 将二进制数 $(1100101.101)_2$ 转换成十六进制数。

**解：** $(1100101.101)_2 = (0110, 0101.1010)_2$

$= (65.A)_{16}$

**三、码制**

码制即编码的方式。编码就是用按一定规则组合成的二进制码去表示数或字符等。这里只介绍几种常用的码制。

1. 二—十进制编码（BCD 码）

由于二进制码优点很多，因此数字系统中均采用二进制。考虑到人们习惯于十进制，所以输入数码一般采用十进制，然后转换成二进制，输出时又转换成十进制显示。为使二进制和十进制之间转换更方便，常使用二进制编码的十进制代码，这种代码称为二—十进制码，简称 BCD 码。

十进制数有 10 种状态，需 4 位二进制数表示，但 4 位二进制数共有 16 种状态，即 0000、0001、0010、……1111，因此有六种多余状态。由于去掉六种多余状态的方法不同，因而出现不同的 BCD 码。如去掉最后六种状态 1010、1011、1100……1111，则得到 8421 码；去掉最前和最后三种状态，即 0000、0001、0010 和 1101、1110、1111、则得到余 3 码。以上都是为了便于某种运算所用的 BCD 码。此外，还有一种 BCD 码叫格雷码。它有很多种编码方式，但各种格雷码都有一个共同特点，就是在任意相邻的两组代码中只有一位码不同。表 1-2 列出了常用的几种 BCD 码。

**表 1-2　常用的 BCD 码**

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

表 1-2 中 8421 码、5421 码、2421 码是有权码。有权码是指 4 位二进制编码中每位数码都有确定的位权值，可以根据位权展开求得所代表的十进制数。需要指出的是表 1-2 中所列的 5421 码、2421 码的编码不是唯一的。

8421 码从高位到低位的位权分别为 $2^3$、$2^2$、$2^1$、$2^0$ 即 8、4、2、1 故称 8421 码。8421 码与十进制之间的转换可直接按位权换算。

**〔例 1-14〕** 用 8421 码表示十进制数 82.5。

**解：**

$$\begin{array}{ccccc} 8 & & 2 & . & 5 \\ \downarrow & & \downarrow & & \downarrow \\ 1000 & & 0010 & . & 0101 \end{array}$$

结果为　$(82.5)_{10} = (10000010.0101)_{8421BCD}$

注意：8421 码和二进制数所表示的多位十进制的方法不同。如 82.5 这个十进制数，用 8421 码表示需 12 位数，而用纯二进制数只需 8 位。

将 8421 码转换成十进制数也十分方便。采用分组的方法，自小数点开始每四个数码为一组，若最后不足 4 位可补零，再把每一组转换成 1 位十进制等值数即可。

〔**例 1-15**〕　将 8421 码 $(0101001001)_{8421BCD}$转换成十进制数。

**解：**

$$(0101001001)_{8421BCD} = (0001,0100,1001)_{8421BCD}$$

$$\begin{array}{ccc} \downarrow & \downarrow & \downarrow \\ 1 & 4 & 9 \end{array}$$

结果为　$(0101001001)_{8421BCD} = (149)_{10}$

同理可将 5421 码、2421 码与十进制数进行转换。

〔**例 1-16**〕　将十进制数 763 分别用 5421 码、2421 码表示。

**解：**

$$\begin{array}{cccc} 7 & 6 & 3 & \\ \downarrow & \downarrow & \downarrow & \\ 1010 & 1001 & 0011 & 5421\text{码} \\ 7 & 6 & 3 & \\ \downarrow & \downarrow & \downarrow & \\ 1101 & 1100 & 0011 & 2421\text{码} \end{array}$$

结果为　$(763)_{10} = (101010010011)_{5421BCD}$

$(763)_{10} = (110111000011)_{2421BCD}$

表 1-2 中余 3 码和格雷码是无权码。无权码是指编码中每位代码没有确定的位权值，因此不能按位权展开式来求它所代表的十进制数。但是这些代码都有其特点，在不同的场合可以根据需要选用。例如余 3 码，是在每个 8421 码上加 $(3)_{10} = (0011)_2$ 而得到的，用余 3 码进行加减运算十分方便。

2. 海明码

二进制信息在传送时，可能会发生错误，利用海明码不但可以发现错误，还能校正错误。下面以 8421 海明校验码为例来说明。

8421 海明校验码是由 8421 码作信息位，再加 3 位校验位组成，它是一个 7 位代码。表 1-3 给出了 8421 海明校验码的编码方式。

**表 1-3　8421 海明校验码**

| 位　序 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|
| 十进制数 | $B_4$ | $B_3$ | $B_2$ | $P_3$ | $B_1$ | $P_2$ | $P_1$ |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |