数字电子技术基础题解

经典

经典教材辅导用书 电子信息系列

知识要点

重点与难点

例题精选

习题解答

高教版《数字电子技术基础题解》(第5版)(阎石 主编)

4

李小珉 叶晓慧 主编

華中科技大學出版社

http://www.hustp.com

数字电子技术基础题解

李小珉　叶晓慧　主编

华中科技大学出版社

中国·武汉

图书在版编目(CIP)数据

数字电子技术基础题解/李小珉　叶晓慧　主编. —武汉:华中科技大学出版社,2007年12月
ISBN 978-7-5609-4186-8

Ⅰ.数…　Ⅱ.①李…　②叶…　Ⅲ.数字电路-电子技术-高等学校-教学参考资料　Ⅳ.TN79

中国版本图书馆CIP数据核字(2007)第134844号

数字电子技术基础题解　　　　李小珉　叶晓慧　主编

策划编辑:李　德
责任编辑:李　德　　　　封面设计:潘　群
责任校对:陈　骏　　　　责任监印:周治超

出版发行:华中科技大学出版社(中国·武汉)
武昌喻家山　　邮编:430074　　电话:(027)87557437

录　　排:武汉众心图文激光照排中心
印　　刷:湖北恒泰印务有限公司

开本:710mm×1000mm　1/16　　印张:24.75　　字数:540 000
版次:2007年12月第1版　　印次:2007年12月第1次印刷　　定价:34.80元
ISBN 978-7-5609-4186-8/TN·113

内容简介

本书是与阎石主编的《数字电子技术基础》(第五版)配套的教学辅导教材。编者按照 2004 年秋教育部重新修订的《数字电子技术基础课程教学基本要求》，在总结了自己教学经验的基础上，重新细化了教学基本要求。每章按知识要点、重点与难点、例题精选和习题解答的顺序编写。对每章的知识点、重点、难点均作了明确、细致的归纳；例题精选都是在习题课中证明有较好教学效果的范题，它们取材于学生常见的错误、实际应用实例及考试的典型试题。独具特色的习题解答，对易错、常错、重点和难点习题由教学经验丰富的老教师执笔给予了详略得当的点拨，既注重基础知识的应用研究，又注重启迪思维培养能力。

本书可作为从事本课程教学的教师和进行本课程学习的学生的辅导用书，也是报考工科硕士研究生考前学习的良师益友。

前　言

2004年秋，教育部电子信息科学与电气信息类基础课程教学指导分委员会为了适应数字化电子应用技术在广阔领域内的深度扩展，面对当前数字技术的新产品的开发日新月异，电子设计自动化(EDA)技术越来越普及，再次对“数字电子技术基础课程教学基本要求”进行了修订。由清华大学阎石教授主编的《数字电子技术基础》(第五版)在第四版的基础上按照新的教学基本要求重新进行修订而成。本版教材在内容、体系的充实与更新上，在习题的安排与选择上，不仅充分体现了新的教学基本要求的思路，更进一步地注重基础、注重应用、注重发展，而且还更加贴近了数字电子技术发展的实际，更具特色。

本书就是为配合第五版的新教材的使用而编写的。其目的在于：一方面能通过以教材为媒介与从事《数字电子技术基础》课程教学的同仁们一道共同切磋新的教学基本要求在教学实践中如何把握与研究，以促进教学质量的提高。另一方面能帮助正在进行本课程学习的学生们去深入具体地全面了解本课程的知识结构、基本知识点、重点与难点，对增进他们主动学好本课程起到积极的推动作用。

全书按教材的章节顺序分为知识要点、重点与难点、例题精选、习题解答四个部分。

本书在编写过程中，根据我们的教学实践经验，突出了以下几点。

(1) 在结合新的教学基本要求，认真研究第五版新教材的内容体系结构的基础上，对照新的教学基本要求，对教材各章节的知识要点、重点和难点重新进行了明确细致的归纳。在归纳中特别注重各知识点之间的层次和关联，并对它们的应用和实践要求都专门作了明确的提示，以保证教师在教学中和学习者在学习中都能做到心中有数和准确的把握。

(2) 例题精选中所选的例题有的是综合了学生常见的典型错误而专门设计的范题，有的是取材于实际应用的实例，还有的是取材于本课程考试的典型试题及本校的考研试题。它们既能紧扣教材的重点和难点，贴近教学实际，有力配合课堂的教学需求，同时又特别注重拓宽知识面，培养知识的综合应用能力的教学基本要求。教学实践证明，它们确实能收到了较好的教学效果。

(3) 独具特色的习题解答。对于教材中的习题解答不仅仅拘泥于答案的给出，而是结合数字电路的特点，对于易错、常错、重点与难点的习题，特由教学经验丰富、对学生作业有专门研究的老教师执笔，结合学生在作业中常犯的错误、难懂的问题有针对性地给予了详略得当的点拨，并且还特别注意了解题方法的指导，以达到启迪思维，培养能力的目的。

全书由李小珉、叶晓慧主编。参加本书编写工作的有嵇碧波、潭浩、王杰玉、张海波、王拓、符志威、吴昊、何光进等同志对各章部分习题作了较为详细的解答。全书由胡

文超高级工程师统稿，并对部分易错、常错、重点、难点的习题亲自执笔一一作答。

本书的出版得到华中科技大学出版社的大力支持和帮助，在此一并表示深切谢意。

由于编者水平有限，书中的错误和不妥之处恳请读者批评指正。

编　者

2007 年 4 月于海工大

目　　录

第一章　数制和码制

一、知识要点

1. 模拟量与数字量、模拟信号与数字信号、模拟电路与数字电路的区别。

2. 数字电路的基本功能：处理各种数字信号。

3. 数制及各种数制之间的互相转换。

4. 二进制算数运算的规则及运算的两个特点。

5. 真值、原码、补码和反码的概念、表示方法及相互间的区别。

6. 模的概念（一个系统的量程或此系统所能表示的最大的数）。在时针系统能表示的最大数为 12，模为 12；长度（又称字长）为 n 位的二进制存数单元，其模为 2^n（它是一个 $n+1$ 位的二进制数 $1\underbrace{00\cdots0}_{n位}$。由于在 n 位的存数单元中只能存入其 n 个 0，最左的“1”就自动舍弃）。当模为 2^n 时，2^n 与 0 是相等的，常记为 $2^n=0(\mathrm{mod}\ \ 2^n)$。$(2^n-1)$ 为 n 位二进制数且全为 1，即 $\underbrace{111\cdots1}_{n位}$。

7. 补码的求法；如何用补码的方法来实现二进制数的减法。

8. 求二进制数补码对应的原码的方法。

9. 数码的两种功能：一是可表示数量的大小，当用来表示数量时，对应为数制；二是可表示不同的事物或事物的状态。当数码用作表示不同事物和状态时，称为代码（即事物或不同状态的代号），或称为码制。

10. 二进制数码与十进制代码之间的区别。

11. 常见的十进制恒权代码（如 8421 码、2421 码、5211 码等）及非恒权代码（如余 3 码）在编码上的各自特点。

12. 格雷码的编码特点及该特点所带来的电路优点。

13. ASCII 代码由几位二进制代码构成。其代码涵盖了几类内容（数字代码、英文大小写字母代码、各种符号代码及控制码）。

二、重点与难点

1. 权的概念。二进制数、八进制数、十进制数、十六进制数之间的相互转换。

2. 二进制算数运算的规则及乘除法运算的特点（可用“移位”和“相加”实现乘除法运算的电路结构思路）。

3. 真值、原码、补码和反码的概念。“补码及补码运算”是本章内容的难点。

(1) 在数字电路中所表示的带符号的二进制数与该二进制数本身书写的表达方式是不相同的。为了区别二者，称二进制数本身(即书写表达)为“真值”。例如，$+10010$，-11001。

(2) 当真值在数字电路中表达时，若其符号位用数码“0”表示，则其为正，用数码“1”表示，则其为负，而数值部分则按真值的二进制形式表示，那么这种电路中的表示形式称为“原码”表示法。如真值为：$+10010$，则其原码表示为$[+10010]_{原码}=\mathbf{0}10010$。又如$-11001$，其原码表示为$[-11001]_{原码}=\mathbf{1}11001$，即在数值部分之前添加一符号位。对于数值部分有效数字为n位的二进制数N，其真值有两种情况：$+N$或$-N$。这时其原码用下式表示为

$$[\pm N]_{原码}=\begin{cases} N & 当\ 0\leqslant +N<2^n(即当真值为+N时) \\ 2^n-(-N) & 当\ -2^n<-N\leqslant 0(即当真值为-N时) \end{cases}$$

注意2^n表示成二进制数时，它是一个$(n+1)$位的二进制数$100\cdots0$，其“1”右边共有n位且均为0。例如$[-11001]_{原码}=2^5-(-11001)=100000+11001=111001$。

(3) 原码、补码和反码都是用数字电路来表示真值的表示方法。采用补码表示法的目的是为了将减法转化为加法，从而使正负数的加减运算转化为单一的加法运算，这样可减少电路的种类并有利于提高运算速度。

阎石主编的《数字电子技术基础》第五版教材(以下均称教材)第10页的式(1.4.1)补码的表示法，要注意看懂书中的含义。该式是指原码为$(n+1)$位除去最左边的一位符号位而剩下的数值部分(有效数字)为n位，该n位所表示的二进制数为N，它的补码$(N)_{COMP}$表示为

$$(N)_{COMP}=\begin{cases} N & (当原码代表为正数时) \\ 2^n-N & (当原码代表为负数时) \end{cases}$$

即该式只是计算将原码的数值部分变成补码的数值部分，而对于符号位部分，则将原码的符号位的符号值照抄，两者共同来完成补码。例如，原码为**0**0011010，其数值部分为0011010，符号位为**0**。变为补码时，其符号位照抄原码的符号位的符号，而数值部分因原码表示正数，故照抄原码的数值部分，即0011010。两者合起来，其补码即为**0**0011010。对于原码为**1**0011010变补码时，要先将原码的数值部分0011010变为补码，即按$(2^7)_{10}-(0011010)_2=(10000000-0011010)_2=(1100110)_2$得到，然后把原码的符号位的符号照抄作为补码的符号位，这样补码最后为**1**1100110。

值得注意的是，当一个二进制数用补码表示后，其最高位为符号位，当符号位为“0”时，其余各位所表示的是该二进制数的原值，但当符号位为“1”时，则其数值部分的各位就不是此二进制数的原值了，而是它的“补数”。这点一定要注意区分开来。

(4) 负数补码的简便求法：正数的补码与其原码相同。负数的补码则要用2^n-N(N为表示负数的原码的数值)来计算，不方便。简单的方法是将负数原码数值部分的各位变为其反码(即0变为1，1变为0)，然后在最低位加1，其符号位则与原码的符号位相同。在最低位加1后，有时会产生进位，此时应将进位完成才是最后结果。

(5) 二进制数的减法。当被减数和减数都用补码表示时，就可以把减法转换为加法，即将这两个补码相加。值得注意的是两补码相加后的结果，若结果的符号位为“0”时，则该结果就是原两数相减的结果；若结果的符号位为“1”时，则该结果不是原两数相减的结果，而是它的补码，它还必须再将补码转换成原码才能得到。

(6) 求负数补码的原码。当补码的符号位为“0”时，该补码对应的原码就是该补码；当补码的符号位为“1”时，则肯定为负数，此时求其对应的原码的方法是，符号位 1 不变，而其数值部分是将补码数值部分的每位取反，然后再在最低位加 1 得到。

(7) 在两个同符号数相加时，它们的绝对值之和不可超过其有效数字位所能表示的最大值，否则会使结果出错。例如 4 位(含符号位)字长的有效数字位为 3 位，它所能表示的最大值为 111。若有 0110＋0101，则结果变成 1011(显然错了)，这是一个负数，因为其符号位变成 1 了。同样，1011＋1101，其结果为 $\boxed{1}$ 0000，前面的 1 丢失，结果变成了 0000，结果是零(应为 －8)显然不对。为了避免出现此类情况，在表示补码的有效位数时应选足够多的位数，以保证相加后的结果不会超出所选位数能表达的最大值。如此例有效数字位为 3 位的两个二进制相加，其最大值为 111 ＋ 111 ＝ 1110，是个 4 位二进制数，再加上 1 位符号位应选用字长为 5 位的二进制码才不会出错。这个问题很重要，它涉及在数字电路设计时应该选用几位的加法器(是一种逻辑部件)来具体实现两个二进制数(补码)的加法。反过来一旦选定了字长为几位的加法器后，那么该数字电路的性能指标也就确定了，即表明它能够进行几位的二进制数加法而不会出错。例如，若选用了字长为 8 位(含 1 位符号位)的加法器，则该加法器用补码表示时其范围为 $+(2^7-1) \sim (-2^7)$，即 $+127 \sim -128$，其运算结果不可超出这个表达范围。

(8) 对有符号位的原码求反码要特别注意一点，就是不要一见到原码就立即按逐位求反，然后再加 1 的方法来求解，这是很容易出错的。从教材 P10 的式(1.4.2)可知，当原码表示正数时，其反码与原码是相同的。只有当原码表示负数时，这时反码才是将原码的数值部分的各位求反，而其符号位与原码相同。

原码、补码和反码都是真值在电路中的具体表达方式，因此它们都是和相应的数字电路相对应的。在此搞清它们的内涵及细节对今后分析和设计与之有关的数字电路是非常有益的。

4. 二－十进制编码。凡是采用若干位二进制数码来表示十进制数的方法，称为二－十进制编码，所得的各种编码亦称为二－十进制代码(BCD 码)或称为十进制代码。

(1) 按有“权”和无“权”分，可分为恒权代码(即代码中每位所代表的“权”是固定的)和非恒权代码(即代码中的每位的 1 所代表的十进制数的含义是不固定的，即权不恒定)。常见恒权码有 8421 码、2421 码、5211 码、5421 码等；非恒权码有余 3 码、余 3 循环码等。

同样，不同的二－十进制编码对应不同的数字电路。具体选用哪种编码要根据对电路的功能和要求来考虑。例如要求电路过渡状态时出现噪声干扰小时可考虑选用“余 3 循环码”；又如要求电路能给出十进制的反码，这时可考虑采用“余 3 码”；若要求给出的十进制数比较好识别，这时应考虑采用有权码。

(2) 用BCD码实现多位十进制数的方法。用二进制表示的十进制数，有 0 ～ 9 十个不同的数值状态，因此十进制数的每一位(如“个”位、“十”位、“百”位等）至少需用4位二进制数来表示。这时就要特别注意用代码表示的十进制数的每位之间是按“逢十进一”规则，而十进制数每位内部的 4 位二进制数却是按“逢二进一”规则，并且它只允许有 10 种状态。当 4 位二进制数累加了 10 次时就一定要产生进位(逢十进一)，同时这 4 位二进制数还必须回复到原始状态(例如对 8421 码要回复到 0000 状态；余 3 码则应回复到 0011 状态)。另一方面，代码十进制的低位向高位进位对高位的内部 4 位二进制的状态影响也是不相同的。例如当采用 8421 码时，在十进制“个”位向“十”位进位后，“十”位内部的 4 位二进制的状态是由 0000 变成 0001。但是如果采用的是余 3 码时，十进制“个”位向“十”位进位后，“十”位内部的4位二进制的状态是由 0011 变成 0100(见教材 P13 的表 1.5.1)。这两个概念要特别引起注意，它们在以后的数字电路分析和设计时会经常遇到。

三、例题精选

例 1-1 将$(215.6531)_{10}$ 转化为二进制数。(精度达到 2%)

解 将十进制数转换成二进制数时，要注意两点，一是应先将整数部分和小数部分分别进行转换，然后再将它们组合起来构成整个的转换结果。二是整数部分转换时，采用“除 2 取余数法”。小数部分转换时，采用“乘 2 取整法”。注意取余和取整时它们方向是不同的。

		余数	
2	215	余数 = 1	k_0
2	107	余数 = 1	k_1
2	53	余数 = 1	k_2
2	26	余数 = 0	k_3
2	13	余数 = 1	k_4
2	6	余数 = 0	k_5
2	3	余数 = 1	k_6
2	1	余数 = 1	k_7
	0		

↑ 取余数后的排列方向

	取出整数
$0.6531 \times 2 = 1.3062$	取出整数 $k_{-1} = 1$
$0.3062 \times 2 = 0.6124$	取出整数 $k_{-2} = 0$
$0.6124 \times 2 = 1.2248$	取出整数 $k_{-3} = 1$
$0.2248 \times 2 = 0.4496$	取出整数 $k_{-4} = 0$
$0.4496 \times 2 = 0.8992$	取出整数 $k_{-5} = 0$
$0.8992 \times 2 = 1.7984$	取出整数 $k_{-6} = 1$

↓ 取整后的排列方向

由于精度要求达到2%，故此处要将二进制小数取 6 位，这时精度可达 $1/2^6 = 1/64 \approx 1.56\%$。这样两者合起来得

$$(215.6531)_{10} \approx (11010111.101001)_2$$

此转换结果的等值十进制数与原十进制数的误差为 0.0125，精度在 2% 以内。因

此，此例说明十进制小数不一定都能转换成完全等值的二进制小数，只能根据精度取相应的二进制小数位数来近似满足。

例 1-2　将下列八进制数转化为十六进制数(1)$(563)_8$；(2)$(0.764)_8$。

解　八进制数转换成十六进制数时，应先将八进制数转换成二进制数，然后再将二进制数转换成十六进制数。对于二进制整数转换成十六进制，应从二进制数的低位开始向左将每取 4 位二进制数的值就转换成一个十六进制数，如果最后在二进制的高位余下的不够 4 位时，则在有效数前面补充若干个"0"，使之变成 4 位二进制数，然后再转换成十六进制数。而对于二进制小数转换成十六进制数时，要从小数点后开始向右，每四位一组分别转换成对应的十六进制数，如果最后剩下的二进制位数不够 4 位，则在最低有效数后面补充若干个"0"，补足 4 位为止再转换。

(1) 把$(563)_8$中的 5、6、3 分别用 3 位二进制数表示如下：

$$\begin{array}{ccc} 5 & 6 & 3 \\ \downarrow & \downarrow & \downarrow \\ \underline{101} & \underline{110} & \underline{011} \end{array}$$

该二进制数共 9 位，当每四位一组分组时，在最高位缺 3 位，故添加 3 个"0"，得到十六进制数为

$$\begin{array}{ccc} \underline{0001} & \underline{0111} & \underline{0011} \\ \searrow & \downarrow & \swarrow \\ (1 & 7 & 3)_{16} \end{array}$$

(2)$(0.764)_8$ 变成二进制数为

$$\begin{array}{ccc} 0.7 & 6 & 4 \\ \downarrow & \downarrow & \downarrow \\ 0.\underline{111} & \underline{110} & \underline{100} \end{array}$$

因小数点后只有 9 位二进制小数，为转换成十六进制数，在最低位后添加 3 个"0"，得到十六进制数为

$$\begin{array}{c} 0.\underline{1111}\ \underline{1010}\ \underline{0000} \\ \searrow\ \swarrow\ \swarrow \\ (0.FA0)_{16} \end{array}$$

当$(0.FA0)_{16}$最低位"0"不是有效数字，故可去掉，变成$(0.FA)_{16}$，这不会影响原数的精度。

同样，把十六进制数转换成八进制数时，也是先将其变成二进制数，然后整数就从低位开始向左每三位一组分组，小数就从小数点后开始向右每三位一组分别转换成相应的八进制数。其在最高位(对整数)和最低位(对小数)添"0"的方法同八进制转换成十六进制的方法一样。但是在分组完成后可能会在最高位(对整数)或最低位(对小数)出现剩余的全为"0"的情况，如$(173)_{16}=(\underline{000}\ \underline{101}\ \underline{110}\ \underline{011})_2$，其最高 3 位全为"0"，又如$(0.FA8)_{16}=(0.\underline{111}\ \underline{110}\ \underline{101}\ \underline{000})_2$，其最低 3 位全为"0"。由于它们都不是有效数字，故可以去掉，则$(173)_{16}=(563)_8$，$(0.FA8)_{16}=(0.765)_8$。

例 1-3 求下列各数所对应的等值十进制数(1)$[00010110]_2$;(2)$[00010110]_8$;(3)$[00010110]_{16}$;(4)$[00010110]_{8421BCD}$。

解 (1)$[00010110]_2 = 0\times2^7+0\times2^6+0\times2^5+1\times2^4+0\times2^3+1\times2^2+1\times2^1+0\times2^0=[22]_{10}$

(2)$[00010110]_8 = 0\times8^7+0\times8^6+0\times8^5+1\times8^4+0\times8^3+1\times8^2+1\times8^1+0\times8^0=[4168]_{10}$

也可以先把$[00010110]_8$转换成二进制数,然后再转换成十进制数。这时应注意每一位八进制数要变成三位二进制数,即

$$[0\ 0\ 0\ 1\ 0\ 1\ 1\ 0]_8$$

$$[\underline{000}\ \underline{000}\ \underline{000}\ \underline{001}\ \underline{000}\ \underline{001}\ \underline{001}\ \underline{000}]_2$$

$$1\times2^{12}+1\times2^6+1\times2^3=[4168]_{10}$$

(3)$[00010110]_{16} = 0\times16^7+0\times16^6+0\times16^5+1\times16^4+0\times16^3+1\times16^2+1\times16^1+0\times16^0=[65808]_{10}$

当然也可仿照以上小题先转换成二进制数后再转换成十进制数。这时每一位十六进制数要变成4位二进制数,请读者自己完成。

(4)$[\underline{0001}\ \underline{0110}]_{8421BCD}=[16]_{10}$(参见教材P13表1.5.1)

注意:8421BCD码不能直接与二进制数转换,它必须先转换成十进制数后,再转换成相应的二进制数。

此例通过在形式上位数和数字相同的4个数来说明不同数制、数码所表示的数的本质区别。这种区别在今后的学习中还会看到,由它们所引起的数字电路也是完全不一样的,因此在此搞清楚对今后的学习会很有帮助。

例 1-4 真值X、Y均为$(n-1)$位二进制数(可正可负),试证明在字长为n位的存数单元中操作时,有$[X]_补+[Y]_补=[X+Y]_补$。

证 分4种情况,即$X\geqslant0,Y\geqslant0$;X或$Y\geqslant0$,Y或$X<0$但$X+Y\geqslant0$;X或$Y\geqslant0$,Y或$X<0$但$X+Y<0$;$X<0,Y<0$来分别证明。

(1) 当$X\geqslant0,Y\geqslant0$,若满足$2^{n-1}>(X+Y)\geqslant0$时,有

$$[X]_补=X \qquad (X\geqslant0)$$

$$[Y]_补=Y \qquad (Y\geqslant0)$$

$$[X]_补+[Y]_补=X+Y=[X+Y]_补 \qquad (\mathrm{mod}\ 2^n)$$

(2) 当$X\geqslant0,Y<0$,若满足$2^{n-1}>(X+Y)\geqslant0$时,由补码定义可得

$$[X]_补=X \qquad (X\geqslant0)$$

$$[Y]_补=2^n+Y \qquad (Y<0)$$

$$[X+Y]_补=X+Y \qquad (X+Y\geqslant0)$$

有 $$[X]_补+[Y]_补=X+2^n+Y=X+Y \qquad (X+Y\geqslant0)$$

所以 $$[X]_补+[Y]_补=[X+Y]_补 \qquad (\mathrm{mod}\ 2^n)$$

(3) 当$X\geqslant0,Y<0$,若满足$0\geqslant(X+Y)\geqslant-2^{n-1}$时,由补码定义可得

$$[X]_补 = X \qquad (X \geqslant 0)$$

$$[Y]_补 = 2^n + Y \qquad (Y < 0)$$

$$[X+Y]_补 = 2^n + (X+Y) \quad (X+Y \leqslant 0)$$

有 $$[X]_补 + [Y]_补 = X + 2^n + Y = 2^n + (X+Y)$$

所以 $$[X]_补 + [Y]_补 = [X+Y]_补 \qquad (\mathrm{mod}\ 2^n)$$

(4) 当 $X < 0, Y < 0$，若满足 $0 > (X+Y) \geqslant -2^{n-1}$ 时，由补码定义可得

$$[X]_补 = 2^n + X \qquad (X < 0)$$

$$[Y]_补 = 2^n + Y \qquad (Y < 0)$$

$$[X+Y]_补 = 2^n + (X+Y) \quad (X+Y < 0)$$

有 $[X]_补 + [Y]_补 = 2^n + X + 2^n + Y = 2^n + (X+Y)$ （其中一个 2^n 产生一个进位而舍去），所以

$$[X]_补 + [Y]_补 = [X+Y]_补 \qquad (\mathrm{mod}\ 2^n)$$

综上，当 $2^{n-1} > X \geqslant -2^{n-1}, 2^{n-1} > Y \geqslant -2^{n-1}$，且 $2^{n-1} > (X+Y) \geqslant -2^{n-1}$ 时，有

$$[X]_补 + [Y]_补 = [X+Y]_补 \qquad (\mathrm{mod}\ 2^n)$$

此例说明，在进行补码加法运算时，不论相加两数的真值是正还是负，只要把它们表示成相应的补码形式，直接相加，其结果一定与真值直接相加后的补码是相等的。

例 1-5　判断下面各命题是否正确：

(1) 真值为 0100111，则其原码为 0100111；

(2) 真值为 1011011，则其补码为 00100101；

(3) 原码为 00111100，则其补码为 01000100；

(4) 原码为 10000001，则其反码为 01111110；

(5) 补码为 01110111，则其原码为 00001001；

(6) 补码为 10010011，则其原码为 11101101；

(7) 反码为 01111110，则其原码为 01111110；

(8) 在 8 位二进制存数单元中操作 $[11110100]_补 + [10000111]_补$，结果为 $[01111011]_补$。

解　(1) 错。因为该真值为正的 7 位二进制数，当表示为原码时，应在其最左边添加 1 位符号位，该符号位此处为“0”，即为**0**0100111。

(2) 错。因为该真值为正，其错把真值当成原码，从而逐位求反加 1 求出补码。

(3) 错。因为原码所表示的是个正数，所以其补码不应该将原码的数值部分各位求反后再加 1。应为 00111100。

(4) 错。原码所表示的是个负数，但是求反码时，只能将原码的数值部分各位求反，而符号位保持不变。应为 11111110。

(5) 错。因补码的符号位是“0”，故其原码是正数，由补码求原码时不应该按求反加 1 来求，两者应该相同，即原码为 01110111。

(6) 对。因补码的符号位为“1”，故原码的符号位也为“1”，而数值部分由补码的数值部分按逐位求反后再加 1 得到。

(7) 对。因反码符号位为“1”,故原码与反码相同。

(8) 错。因为相加的两数为同符号,且它们真值数值部分之和超出了 7 位所能表达的最大数字($2^7-1=127$),致使这两数操作的结果变成正数,还是不可能的。

本例将一些常见的、易犯的错误列出,请学习者能举一反三避免犯类似的错误。

例 1-6 已知$[X]_{补}$,求$[-X]_{补}$(在 n 位存数单元中操作)。

解 在做减法时,减一个数等于加上该数的负数。在进行补码运算时,若出现减法,这就需要先求出$[-X]_{补}$,然后利用例 1-4 的结论来求。

当 $0 \leqslant X < 2^{n-1}$ 时,即 $X \geqslant 0$,设 $X = x_1x_2\cdots x_{n-1}$,有

$[X]_{补} = 0x_1x_2\cdots x_{n-1} = X$

$-X = -x_1x_2\cdots x_{n-1}$

$[-X]_{原} = 1x_1x_2\cdots x_{n-1}$

$[-X]_{补} = 1\bar{x}_1\bar{x}_2\cdots\bar{x}_{n-1}+1$ (其中 $\bar{x}_1\bar{x}_2\cdots\bar{x}_{n-1}$ 表示对 $x_1x_2\cdots x_{n-1}$ 逐位取反)

比较$[X]_{补}$和$[-X]_{补}$可见,求$[-X]_{补}$是将$[X]_{补}$连同符号位的所有各位逐位求反再加 1 得到。

当 $-2^{n-1} \leqslant X < 0$,即 $X < 0$ 时,有

若 $[X]_{补} = 1x'_1x'_2\cdots x'_{n-1}$ (注意 $x'_1 \neq x_1, x'_2 \neq x_2, \cdots, x'_{n-1} \neq x_{n-1}$)

有 $[X]_{原} = 1\bar{x}'_1\bar{x}'_2\cdots\bar{x}'_{n-1}+1$ (由补码求原码)

则 $[-X]_{原} = 0\bar{x}'_1\bar{x}'_2\cdots\bar{x}'_{n-1}+1$ (注意 $-X > 0$)

$[-X]_{补} = 0\bar{x}'_1\bar{x}'_2\cdots\bar{x}'_{n-1}+1$

由此可见,当 $X < 0$ 时,比较$[X]_{补}$与$[-X]_{补}$,求$[-X]_{补}$也是将$[X]_{补}$中连同符号位一起逐位求反再加 1 得到。

综上,无论 $X \geqslant 0$ 还是 $X < 0$,当已知$[X]_{补}$求$[-X]_{补}$时,只要将$[X]_{补}$中连同符号位一起的所有各位逐位求反后再加 1 便可得到。请注意这与由原码求补码(或由补码求原码)是很不相同的,应注意区分开。

例 1-7 已知$[X]_{补}$和$[Y]_{补}$,试求下列各题的$[X-Y]_{补}$:(1)$[X]_{补}=011101$,$[Y]_{补}=001100$;(2)$[X]_{补}=000101$,$[Y]_{补}=111010$;(3)$[X]_{补}=101101$,$[Y]_{补}=111101$。

解 利用$[X-Y]_{补} = [X+(-Y)]_{补} = [X]_{补}+[-Y]_{补}$来求解。

(1) 由$[Y]_{补}=001100$,得$[-Y]_{补}=110100$

故 $[X-Y]_{补} = 011101+110100 = 010001$

(2) 由$[Y]_{补}=111010$,得$[-Y]_{补}=000110$

故 $[X-Y]_{补} = 000101+000110 = 001011$

(3) 由$[Y]_{补}=111101$,得$[-Y]_{补}=000011$

故 $[X-Y]_{补} = 101101+000011 = 110000$

例 1-8 若在 4 位存数单元中操作,试求下列各式的结果:(1)$(1011)_2+(0101)_2$;(2)$(0001)_{5211码}+(1)_{10}$;(3)$(1100)_{余3码}+(1)_{10}$;(4)$(1001)_{8421码}+(1)_{10}$。

解 (1)$(1011)_2+(0101)_2=(0000)_2$

此题若是人工在纸上运算,其结果就是$(10000)_2$。但是当在 4 位存数单元中进行二进制

加法时，由于它只能存放 4 位二进制数（正如算盘只有 4 格，只有“个、十、百、千”位，没有“万”位一样），所以在第 4 位产生向高位的进位 1 时就自动舍弃，结果就成为 $(0000)_2$，这种电路现象通常称为“溢出”。

(2) ～ (4) 小题是二 - 十进制代码作加法的情况。二 - 十进制代码所表示的是十进制数，因此它可以与十进制数相加，相加的结果仍为二 - 十进制代码所表示的十进制数。但是要注意它们的相加既不同于二进制数的加法，也不同于十进制数的加法。下面结合实例来说明，请学习者注意细心体会。

(2) $(0001)_{5211码} + (1)_{10} = (0100)_{5211码}$

此题可发现其相加不同于二进制相加，如果是二进制，则 $(0001)_2 + (1)_2$ 其结果是 $(0010)_2$。而这里 $(0001)_{5211码}$ 的第 1 位与 $(1)_{10}$ 相加是按二进制加法规则相加，其结果尽管使 5211 码的第 1 位变为“0”，而它的进位 1 是进到 5211 码的第 3 位，这是因为 5211 码第 3 位的“权”是“2”的缘故。由此可见，凡是在进行二 - 十进制代码的加减法运算时应按教材 P13 的表 1.5.1 中十进制数所对应的代码的编码来得到结果，这点对今后数字电路的设计与分析非常重要，请学习者注意。

(3) $(1100)_{余3码} + (1)_{10} = (0011)_{余3码}$

由教材 P13 表 1.5.1 可见，$(1100)_{余3码}$ 对应十进制数 $(9)_{10}$，故 $(9)_{10} + (1)_{10}$ 应等于 $(10)_{10}$，但对于只有 4 位的存数单元，它只能存放 $(10)_{10}$ 的个位 $(0)_{10}$，故结果得到的应该是余 3 码的 $(0)_{10}$ 的代码 $(0011)_{余3码}$，而不是 0000。在电路设计时，就要设法使电路能实现 $(1100)_{余3码} + (1)_{10}$ 后在 4 位存数单元中的状态是 0011，而不会是 0000。通过今后的学习能做到这一点。

(4) $(1001)_{8421码} + (1)_{10} = (0000)_{8421码}$

由教材 P13 表 1.5.1 可知 $(1001)_{8421码}$ 对应 $(9)_{10}$，所以它与 $(1)_{10}$ 相加后，在 4 位存数单元中所得的是 8421 码对应的 $(0)_{10}$ 的 4 位状态，即 0000，进位自动丢失。在电路设计时，要设法使 $(1001)_{8421码}$ 的第 1 位与 $(1)_{10}$ 相加后所产生的进位直接与第 4 位的“1”相加使其变成“0”（而进位丢失），从而使 4 位的状态由 1001 变成 0000。在今后的学习中可以找到实现它的方法。

例 1-9　分别用 8421BCD 码、余 3 码、5211 码代换下列十进制数。(1) $(23)_{10}$；(2) $(706)_{10}$；(3) $(91)_{10}$。

解　(1) $(23)_{10} = (\underline{0010}\ \underline{0011})_{8421BCD}$，$(23)_{10} = (\underline{0101}\ \underline{0110})_{余3码}$

$(23)_{10} = (\underline{0100}\ \underline{0103})_{5211码}$

(2) $(706)_{10} = (\underline{0111}\ \underline{0000}\ \underline{0110})_{8421BCD}$，$(706)_{10} = (\underline{1010}\ \underline{0011}\ \underline{1001})_{余3码}$

$(706)_{10} = (\underline{1100}\ \underline{0000}\ \underline{1001})_{5211码}$

(3) $(91)_{10} = (\underline{1001}\ \underline{0001})_{8421BCD}$，$(91)_{10} = (\underline{1100}\ \underline{0100})_{余3码}$

$(91)_{10} = (\underline{1111}\ \underline{0001})_{5211码}$

此例给出了二 - 十进制代码表示多位十进制数的方法。此时代码内的多位就可以进行由低位向高位的进位，例如 $(1100\ 0100)_{余3码} + (1100)_{余3码} = (0011\ 0011)_{余3码}$，即 $(91)_{10} + (9)_{10} = (00)_{10}$。