限位数理论与运算器设计

姜咏江
对外经济贸易大学信息学院，北京（100013）
E-mail:accsys@126.com

摘　要：用数码按照“逢 N 进一”规则书写的限位数，最高进位会丢失，这种看似缺点的现象，恰为机器实现算术运算带来了方便。数码书写的限位数不但有序，而且总数一定，其中和为总数的两数中，较大的数如果代表较小数的相反数，不但能解决用限位数表示正负数，而且也能解决用机器实现算术运算的问题。限位数是计算机产生和发展的重要理论之一，运用它来进行计算机设计，简单易行。

关键词：限位数；补码；反码；运算器；计算机设计
中图分类号：TP301，TP302，TP303

1 引言

无论国内还是国外的计算机教材，涉及到运算器设计，都是用二进制运算与逻辑电路的转换，直接搭建了加减法运算器[1]，没有能够恰当地解释机器数值运算的道理。多数教材提到了数制理论，然而，又与计算机运算器的实际设计思想相差甚远，基本上都从实数理论对运算器的电路设计“进行解释，但不能应用于器件设计[2]”。

究竟计算机运算器的设计是根据什么理论来完成的？本文作者在 2004 年提出了限位数的概念和方法[3]，将二进制数的机器表示问题，提升到任意进制数，从而可以很好解释计算机设计中一些作法的合理性，也可以说明类似“+0 和-0”[4]概念的错误之处。

将限位数理论和方法，用到了计算机软硬件设计中[5]，不仅思路清晰，而且能够收到简单易行的效果。

2 限位数

数，一般都由数码记录。手写数不考虑数码有多少，而计算机却有位数的限定。

2.1 限位数的概念

用数码表示，位数一定的无符号整数，叫限位数。

限位数为了表示位数多少，无效 0 不能省略。例如，001，00001 是不同的限位数。

限位数按无符号整数进行运算，叫面值运算。

2.2 限位数特性

（1）限位数的数量一定。例如，3 位十进制数只有 1000 个，而 8 位二进制数只有 256 个。一般来说，n 位 N 进制数共有 N^n 个。总数 N^n 叫限数。

（2）最大的限位数是由最大的数码组成的，它比限数少 1。例如，4 位十进制的 9999 是最大数，而限数是 10000。显然限数是不能用限位数表示的最小正整数。

（3）限位数有一一对称性。

限位数的对称规则是“两数面值之和为限数”。例如四位十进制限位数，001 和 999、002 和 998、…、499 和 501，500 和 500 是自身对称。显然 000 没有对称的数。

限位数的对称数也叫补码。

（4）限位数进行面值算术运算，加法、乘法，超过位数限制的进位会丢失，不够减的

运算也不能进行。因而面值运算的变化一般不能用“=”连接。

我们规定，面值运算变化用“<=>”连接。例如，522×321<=>562，因为最高位向上进的167丢失了。

（5）假定限位数某一处有小数点，小数点移动，不影响限位数的特性（1）～（4）。

3 用限位数计算

限位数虽然运算范围有限，但却是计算机运算所必须依赖的形式。如果不发生减法运算，限位数面值运算的结果，发生了最高位进位，就说明结果超出范围了。如果用限位数表示正负数，那么面值运算最高位发生进位，值可以是正确的。

3.1 表示正负数

不用添加正负号，用限位数就可以表示正负数。这是利用限位数的对称性和正负数对称性一致实现的。由于限位数中，面值较小限位数的补码在面值较大的部分，而限位数的相反数与限位数的补码一一对应。因而可以规定：**比对称点大的数表示的是其补码的相反数，否则表示原数**。

例如，3位十进制限位数502表示的值是498的相反数-498。

由于偶进制的对称点正好是一个限位数，并且补码是自身，为避免二义性，就规定其为负数。如此偶进制限位数表达的正负数范围负数比正数多一个最小数。

限位数代表的十进制有符号数，叫这个限位数的值。

限位数值相等，用“=”号连接。

3.2 限位数加法可以代替值的加减运算

各进制整数如果可以用限位数表示，那么它们的加减运算，就可以用限位数的加法来完成。

例如，22-12=22+(-12)，由于2位限位数22 可以表示值22，而88可以表示值-12，所以用22+88<=>10限位数的加法，就得到了10这个限位数，它表示的值是10，可见结果正确。

3.3 溢出

用限位数面值加法，不一定能够完全代替十进制数的加减法运算。

例如，十进制的 40+20，用限位数面值加法运算表示，得到的值是-40。因为面值运算的结果虽然是60，但因为2位十进制限位数表示正负数范围是-50～49，根据值的规定，限位数60的值是-40，可见是错误的。

限位数面值运算得到错误结果，这种情况叫溢出，之所以叫溢出，是因为实际的运算结果，不在限位数表值的范围之内。

3.4 消除溢出

限位数面值加法运算的错误是由于结果值超出范围造成的，因而用扩充位数的方法，一定能在更大的范围内得到正确的表示。要不要扩大限位数的位数，关键在于能够知道是否溢出。

如何判断和消除溢出？可以证明：**限位数加减运算结果的值符号与第一个数值符号不同，一定溢出，否则不溢出**[6]。

例如，-41-10 用限位数表示为 59+90<=>49。由于 49 表示的是正数，而 59 表示的是负数，限位数 49 与 59 的符号不同，则一定溢出。实际上，-41-10 = -59，结果超出了 2 位十进制数的表数范围。

扩大限位数的位数就可以消除溢出。保值扩充限位数的位数，方法是：**表示正数的限位数前面加数码“0”，表示负数的限位数前面加最大数码。**

例如，-41-10，用限位数表示为 59+90，保值扩大成 3 位是 959＋990<=>949 = -51，它的值为-51，可见不再溢出了。

总而言之，用限位数加法运算可以在需要的范围内，替代十进制数的加减法运算。

4 机器求补码

计算机用限位数加法，解决了数值的减法运算，其关键是将减法运算变成了减数的补码表示。那么如何求一个数的补码，就成了必须解决的问题。虽然根据定义，可以用总数与限位数作减法得到一个限位数的补码，但这又回到了减法运算，故需另外寻求方法。

4.1 反码与反码数

用机器求补码要用到反码的概念。

两个数码之和为最大数码，其中一个叫另一个的反码。一个数的全部数码换成其反码所成的数，叫原数的反码数，有时也称反码。

4.2 用反码求补码

显然，两个反码数的和是最大数码组成的面值最大的限位数。而最大数码组成的限位数比总数小 1。

不妨设 a，b 分别是限位数 c 的反码和补码，那么有 b+c = c+a+1；则 b = a+1。这就是说，**一个限位数的补码等于其反码加一**。

例如，45108 的反码是 54891，而它的补码是 54891+1=54892。

4.3 机器求反码的电路

由于机器直接使用的是二进制数，二进制的反码，用一个逻辑非门就可以求出。如果想控制一位数是否求反，用一个异或门就可以控制，称之为反向控制器。反向控制器的真值表如表 1 所示。

表 1　反相控制器真值表

a（输入）	k（控制）	a 与 k 的异或	注释
0	0	0	原码
1	0	1	
0	1	1	反码
1	1	0	

由表 1 可见，控制端 k=0，输出端得到的是输入端的原码；而当 k=1 时，输出端得到的是输入端的反码。

5 用限位数设计运算器的例子

二进制全加器的进位 d 和本位和的逻辑表达式是：

d = ab+bc+ac

s = a’b’c+a’bc’+ab’c’+abc

用二进制全加器可以解决两个一位二进制数的带下端进位的加法。如果用补码，就可以完成这两个数的减法运算。其中补码使用“反码加一”得到。

图 1 是典型应用限位数设计运算器的例子，其中在 b 端接入一个反向控制器，并将下进位与控制线连在一起，就得到了一个一位的加减法运算器。当 k=1 时，b 端通过反向控制器进入全加器的是 b 的反码 b’，又由于 c 与 k 同值，故通过全加器实现了 a+b’+1，即 a 与 b 的补码相加。

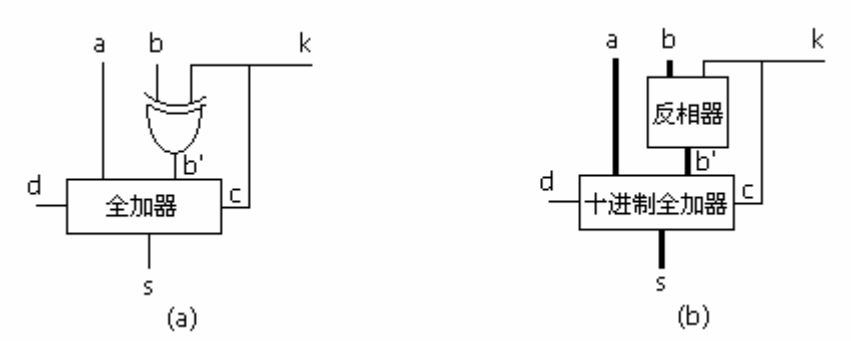

图 1　一位加减法运算器电路

图 1 (a)是二进制电路，图 1(b)是十进制电路。其实任何进制的一位加减法运算器的电路都如图 1(b)所示，只是全加器和反相器的设计要比二进制复杂一些罢了。

图 2 是四位十进制加减法运算器的构造，为了监控限位数所表示值的符号，在第一个数的最高位和输出的最高位，引出了负数判断的符号标志，并通过异或门得到溢出标志，通过溢出标志线来及时处理溢出。

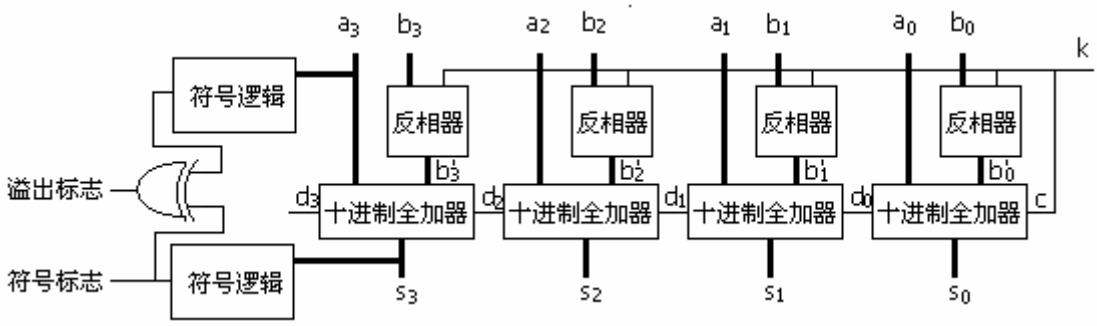

图 2　四位十进制加减法运算器

十进制加减法器输入输出都是 4 位二进制数表示的十进制数，进位线、控制线和标志线都是一位的二进制变量。设计中要注意，不论多少进制的加法运算，单位 1 都是相同的，进位不会超过 1。

十进制数的符号逻辑是“最高位小于 5 是正数或零，否则是负数”。而十进制反相器是依据（0，9）、（1，8）、（2，7）、（3，6）、（4，5）的互反关系设计的。十进制全加器的进位“1”是在 a 端、b 端和下进位端相加，结果为 10 至 19 的情况下得出的。

十进制加减法运算器的各器件，也是通过逻辑电路搭建起来的，虽然与二进制相比，复杂度较高，但作为运算器的部件，体现限位数的方法是完全一致的。

6 计算机的精确计算

计算机运算器的位数是固定的，因而它能够进行的算术运算，一般都是限位数面值运算。

由于限位数表达数值的范围有限，似乎一定位数的计算机不可能进行精确计算。其实，用限位数运算的方法，可以借助存储器实现任何精确的计算。

6.1 超长数的加减运算

借助于存储器的运算问题，多半都是软件的设计问题。对于加减法运算的超长数，我们可以用运算器的位长来分段存储，然后用加减法运算器分段计算。

限位数运算将数值的减法运算变成了加法。因而，超长数值计算的问题，都变成了加法运算的问题。限位数的分段加法，主要是要考虑“带进位加法”和“不带进位加法”。为此，要在图 2 的“c”端连入进位控制电路，还要将每次运算的进位，用一个一位寄存器保存起来，以便供带进位的加法使用。

图 3 是带进位控制的十进制加减法运算器，可以设计相应指令，完成超长数加减法运算。图中，k 是取反控制端，yi 是“1”控制端，jw 是带进位运算控制端，L 是寄存器 REG 的前次运算输入控制端。各控制端变化的真值表如表 2 所示。

表 2　带进位运算控制真值表

说明	运算类型	k（取反控制）	yi	jw（进位）
超长数加法	不带进位加法	0	0	0
	带进位加法	0	0	1
超长数减法	不带进位减法	1	1	0
	带进位减法	1	0	1

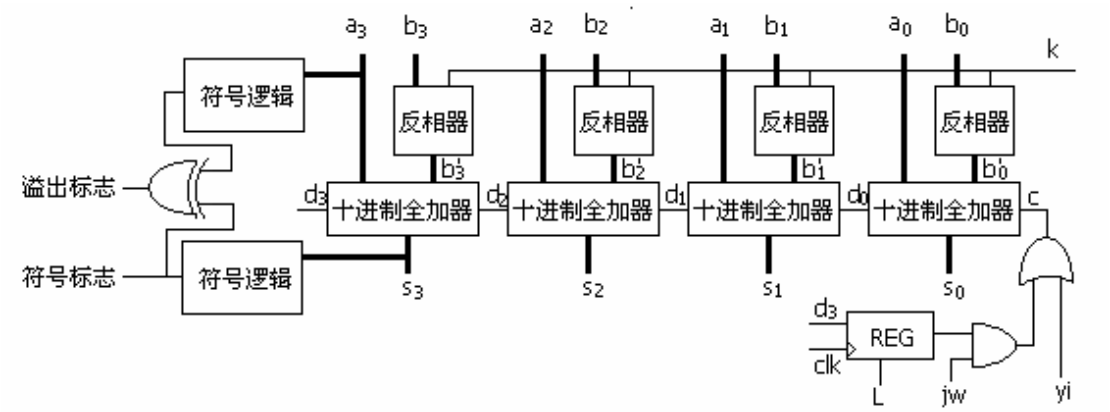

图 3　带进位控制的加减法运算器

超长数码记录的数，可以按着限位数分段存储，最高段位数不够，可以按值是正负数扩充的方法，得到更长的限位数。超长数分段运算，最右面一段不带进位，接下来的各段都需要带进位进行计算。

6.2 小数的处理

用数码表示的带小数，在一个具体的问题中总能找到最长的小数位数，例如 n 位小数。这时可以将所有数的小数点右移 n 位，将参与运算的数都变成整数。这种小数点右移的结果，使每个数都扩大了进制基数的 n 次方倍。使用超长数的运算方法进行整数计算，最终将结果的小数点再左移 n 位，就是带小数运算的结果。

例如，3452 - 4.876，右移小数点 3 位，得 3452000 - 4876，用 4 位限位数表示，得 0345_2000 + 9999_5124，可分成两次，用 4 位限位数加法运算器计算。不带进位加法是：

2000＋5124 <=> 7124

运算产生的进位是 0（带着进位，不用考虑溢出）；带进位的加法运算是：

0345+9999+0 <= >0344

最终结果是 3447.124。通过验证，可以说明结果的正确性。

注意，分段做限位数加法运算，最高段运算结束，要判断是否溢出，如果溢出，可再增加一段限位数处理。

有了图 3 的基本硬件结构，就可以设计相应的指令，然后通过程序设计，最终完成复杂的超长限位数的运算。

7 结论

限位数的概念和方法，不仅为计算机的运算器设计提供了切实可行的理论支持，而且也充分地体现了，数学认识的有限向无限转化的过程。限位数理论是计算机赖以产生的基础理论之一。

参考文献

[1] David Money Harris,Sarah L.Harris．数字设计和计算机系统结构（英文版）．北京，机械工业出版社，2007.10，p233-240．
[2] 金兰．计算机组织与结构．北京，高等教育出版社，1986.5，p80-141．
[3] 姜咏江．补码制理论的理解．北京，计算机工程与应用， 2004.5．
[4] 张功萱，顾一禾，邹建伟（等）．计算机组成原理．北京，清华大学出版社，2005.9，p25．
[5] 姜咏江．计算机原理综合课程设计．北京，清华大学出版社，2009.6，p75-114．
[6] 姜咏江．计算机设计的基础理论限位数．2009 中国计算机大会论文集，2009.10，p362．

The Theory of Fixed-Length Number and Design Arithmetical unit

Jiang Yongjiang
School of Information Technology & Management Engineering, University of International Business and Economics, Beijing (100013)

Abstract

Defers to with the numerical code “meets N to enter one” the rule writing fixed-length number highest carrying to be able to lose, this kind looked that resembles the shortcoming the phenomenon, realized the arithmetic operation for the machine to bring conveniently exactly.Not only digital writing fixed-length number order, moreover the total is certain, if sum of two fixed-length numbers is the total, the big number can express the other inverse.So that not can only solve with the fixed-length number shows the positive and negative number, moreover can also solve with the machine realizes the arithmetic operation question. The one computer production's important theory is the fixed-length digit, designs the computer with the fixed-length number will be simple and easy.

Keywords: fixed-length number;Base complement;Symmetric digital;Arithmetic device;Computer design

作者简介：

姜咏江 1945 年生。 中国计算机学会高级会员。计算机设计专家，中嵌联盟特聘专家。研究计算机理论与设计。成功设计了动态计算机核 PMC362 及多种计算机核结构。个人设计制作了教学计算机 PMC110 等。