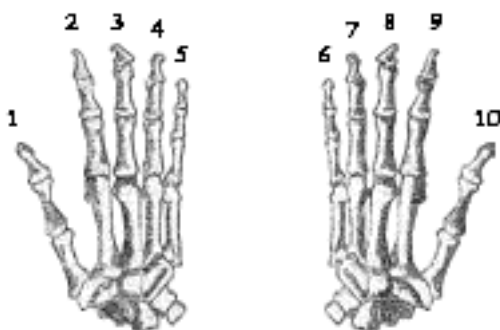


第8章 其他进位制记数法

10对我们来说是一个非常重要的数字。10是我们大多数人拥有的手指或脚趾的数目，我们当然希望所有人的手指脚趾都是10个。因为我们的手非常适合数数，因而我们人类已经适应了以10为基础的数字系统：



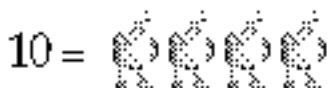
前面数章已经提到过，通常使用的数字系统称为以10为基础的数字系统或十进制。这个数字系统对我们来说非常自然，因而我们很难想像出还有其他的数字系统。事实上，当我们看到数字10的时候，不由自主地就会认为这个数是指下面这么多只鸭子：



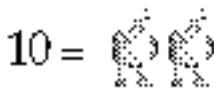
但是，数字10是指这么多只鸭子的唯一理由是因为这么多只鸭子与我们的手指数目相同。如果人类不是有那么多只手指，我们数数的方式就会有所不同，数字10就可能代表别的东西了。同样是数字10，可以指这么多只鸭子：



或这么多只鸭子：



甚至可以是这么多只鸭子：



当我们明白了10可以指只有两只鸭子的时候，也就可以解释开关、电线、灯泡、继电器（或干脆就叫计算机）是怎样表示数字的了。

如果人类像卡通人物那样，每只手上只有 4 个手指会怎样呢？我们可能永远都不会想到要发明一种以 10 为基础的数字系统的问题，取而代之的是我们可能会认为数字系统基于 8 是正常、自然、合理、必然的，是毫无疑问的，是非常合适的。这时，就不能称之为十进制了，得将它称作为以 8 为基础的数字系统或八进制。

如果数字系统是以 8 为基础组织起来的，就不需要这样的—个符号：

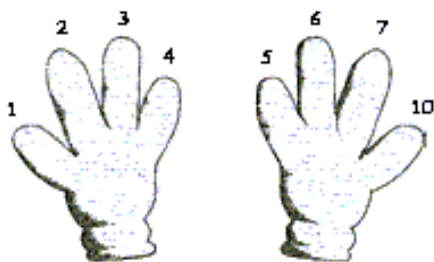
9

把这个符号拿给任何一个卡通人物看，都会有同样的反应：“那是什么？它是干什么用的？”如果再仔细想—会儿的话，你会发现连这样的—个字符也不需要：

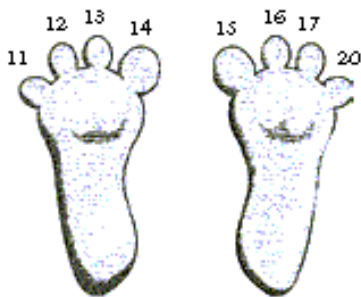
8

在十进制数字系统中，没有专门用来表示 10 的符号，所在在八进制数字系统中，也没有专门用来表示 10 的符号。

在十进制数字系统中数数的方式是 0、1、2、3、4、5、6、7、8、9，然后是 10。在八进制数字系统中数数的方式是 0、1、2、3、4、5、6、7，然后是什么呢？我们已经没有符号可用了，唯一的一个有意义的可用符号是 10，的确是那样。在八进制数中，7 之后紧接着的数字是 10，但是 10 并不是指人类的手指那么多的数目。在八进制数中，10 指的是卡通人物手指的数目：



继续数脚趾头：



使用非十进制的数字系统时，将数字“10”读作“么零”可以避免—些混淆。同样，“13”可以读作“么三”，“20”可以读作“二零”。要想真正避免混淆，可以将“20”读作“八进制二零”或“基于 8 的数二零”。

即使没有手指和脚趾帮忙，我们仍能够将八进制数继续数下去。除了要跳过那些含有 8 或 9 的数字以外，它基本上和数十进制的数是一样的。当然，相同的数字代表的数量是不同的：

0、1、2、3、4、5、6、7、10、11、12、13、14、15、16、17、
20、21、22、23、24、25、26、27、30、31、32、33、34、35、36、37、40、
41、42、43、44、45、46、47、50、51、52、53、54、55、56、57、60、61、62、
63、64、65、66、67、70、71、72、73、74、75、76、77、100...

最后一个数字读作“么零零”，是卡通人物拥有的手指数自乘的结果（即平方）。

在写十进制或八进制数时，为避免混淆，可以借助使用特定的标记以区别表示数字系统。

下面用标记“TEN”表示十进制数，标记“EIGHT”表示八进制数。

这样，白雪公主遇到的小矮人的数目是 7_{TEN} 或 7_{EIGHT}

卡通人手的手指数是 8_{TEN} 或 10_{EIGHT}

贝多芬写的交响乐的首数是 9_{TEN} 或 11_{EIGHT}

人的手指的数目是 10_{TEN} 或 12_{EIGHT}

一年中的月份数是 12_{TEN} 或 14_{EIGHT}

两个星期所包含的天数是 14_{TEN} 或 16_{EIGHT}

“情人”的生日庆祝会是 16_{TEN} 或 20_{EIGHT}

一天中所包含的小时数是 24_{TEN} 或 30_{EIGHT}

拉丁字母表中的字符数是 26_{TEH} 或 32_{EIGHT}

与一夸脱液体相当的盎司数为 32_{TEN} 或 40_{EIGHT}

一副牌中含有的牌数是 52_{TEN} 或 64_{EIGHT}

国际象棋棋盘的方格数是 64_{TEN} 或 100_{EIGHT}

Sunset Strip 最著名的17牌号是 77_{TEN} or 115_{EIGHT}

美式足球场的面积是 100_{TEN} 或 144_{EIGHT}

参加温布尔登网球公开赛女单初赛的人数是 128_{TEN} 或 200_{EIGHT}

古埃及孟斐斯城市面积的平方英里数是 256_{TEN} or 400_{EIGHT}

注意，在上面一系列的八进制数中，有一些好整数，像 100_{EIGHT} 、 200_{EIGHT} 、 400_{EIGHT} 。好整数通常是指结尾有一些零的数。在结尾处有两个零的十进制数意味着它是 100_{TEN} 即 10_{TEN} 乘以 10_{TEN} ；在八进制数中，结尾处有两个零表示它是 100_{EIGHT} 即 10_{EIGHT} 乘以 10_{EIGHT} （或 8_{TEN} 乘以 8_{TEN} ，等于 64_{TEN} ）。

你可能已经注意到了，好的八进制整数 100_{EIGHT} 、 200_{EIGHT} 和 400_{EIGHT} 与十进制数 64_{TEN} 、 128_{TEN} 、 256_{TEN} 相等，它们都是2的整数次幂。例如， 400_{EIGHT} 等于 4_{EIGHT} 乘以 10_{EIGHT} 乘以 10_{EIGHT} ，所有这些数都是2的整数次幂。任何时候，将2的整数次幂和另一个2的整数次幂相乘，得到的仍是2的整数次幂。

下表给出了一些2的整数次幂的十进制及其对应的八进制的表示形式：

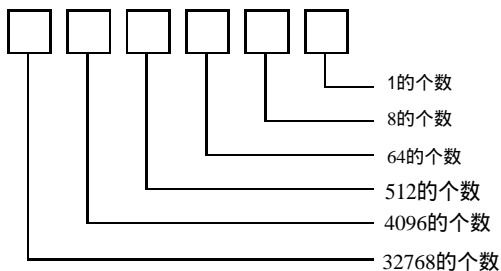
2的整数次幂	十进制数	八进制数
2^0	1	1
2^1	2	2
2^2	4	4
2^3	8	10
2^4	16	20
2^5	32	40
2^6	64	100

(续)

2^7	128	200
2^8	256	400
2^9	512	1000
2^{10}	1024	2000
2^{11}	2048	4000
2^{12}	4096	10000

最右边一列的好整数给我们一个暗示：十进制以外的数字系统可能对使用二进制有所帮助。

八进制数字系统和十进制数字系统在结构上没有什么差别，只是在细节上有一些差异。例如，八进制数的每一个位置代表的值是该位数字乘以 8 的整数次幂的结果：



这样，八进制数 3725_{EIGHT} 可以拆分成这样：

$$3725_{\text{EIGHT}} = 3000_{\text{EIGHT}} + 700_{\text{EIGHT}} + 20_{\text{EIGHT}} + 5_{\text{EIGHT}}$$

还可以写成另外几种不同的形式。下面就是其中的一种，采用十进制形式的 8 的整数次幂：

$$\begin{aligned} 3725_{\text{EIGHT}} &= 3 \times 512_{\text{TEN}} + \\ &\quad 7 \times 64_{\text{TEN}} + \\ &\quad 2 \times 8_{\text{TEN}} + \\ &\quad 5 \times 1 \end{aligned}$$

采用八进制形式的 8 的整数次幂的情况：

$$\begin{aligned} 3725_{\text{EIGHT}} &= 3 \times 1000_{\text{EIGHT}} + \\ &\quad 7 \times 100_{\text{EIGHT}} + \\ &\quad 2 \times 10_{\text{EIGHT}} + \\ &\quad 5 \times 1 \end{aligned}$$

还有另外的一种拆分形式：

$$\begin{aligned} 3725_{\text{EIGHT}} &= 3 \times 8^3 + \\ &\quad 7 \times 8^2 + \\ &\quad 2 \times 8^1 + \\ &\quad 5 \times 8^0 \end{aligned}$$

如果算出其十进制的结果，会得到 2005_{TEN} 。这就是将八进制数转换成十进制数的方法。

可以采用与做十进制加法和乘法相同的办法来做八进制数的加法和乘法。唯一真正的区别在于要采用不同的表格来对各个数字进行乘法或加法运算。下面是八进制数的加法表：

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	10
2	2	3	4	5	6	7	10	11
3	3	4	5	6	7	10	11	12
4	4	5	6	7	10	11	12	13
5	5	6	7	10	11	12	13	14
6	6	7	10	11	12	13	14	15
7	7	10	11	12	13	14	15	16

例如, $5_{\text{EIGHT}} + 7_{\text{EIGHT}} = 14_{\text{EIGHT}}$ 。可以采用与做十进制加法相同的方法将两个稍长一点儿的八进制数相加:

$$\begin{array}{r} 135 \\ + 643 \\ \hline 1000 \end{array}$$

先从最右边的一列做起, 5加上3等于10, 该位写下0, 向前进1; 1加3加4等于10, 该位写下0, 向前进1; 1加1加6等于10。

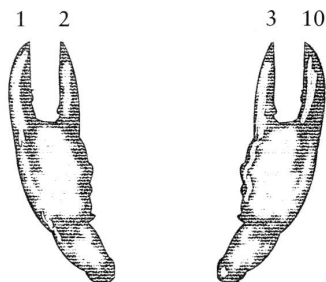
同样, 在八进制中, 2乘以2仍然等于4。但是3乘以3却不等于9, 那是多少呢? 3乘以3等于11_{EIGHT}, 此数与9_{TEN}所代表的数量相等。下图是完整的八进制数的乘法表:

x	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	10	12	14	16
3	0	3	6	11	14	17	22	25
4	0	4	10	14	20	24	30	34
5	0	5	12	17	24	31	36	43
6	0	6	14	22	30	36	44	52
7	0	7	16	25	34	43	52	61

这里, 4×6 等于 30_{EIGHT} , 也即表明 30_{EIGHT} 和 4×6 的十进制结果 24_{TEN} 是等值的。

八进制数字系统与十进制数字系统一样, 都是有效的, 但八进制数字系统在理解上更深了一层。既然我们已为卡通人物开发出了一套数字系统, 就再给龙虾开发一套适合它们用的数字系统吧。龙虾根本没有手指, 但它两只前爪的末端都有螯。适合于龙虾的数字系统是四

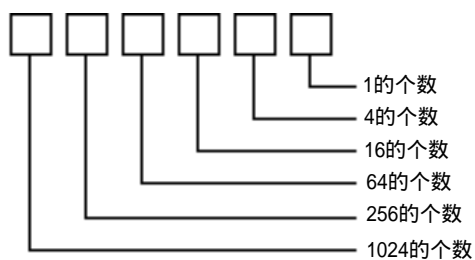
进制数字系统或称为基于4的数字系统：



四进制数可以这样来数：

0、1、2、3、10、11、12、13、20、21、22、23、30、31、32、33、100、101、102、103、110，等等。

这里不打算在四进制数上花太多的时间，因为还有更重要的事情要做。但我们还是要看一下四进制中的每一位是怎样和4的某个整数次幂相对应的：



四进制数31232可以写成：

$$31232_{\text{FOUR}} = 3 \times 256_{\text{TEN}} + \\ 1 \times 64_{\text{TEN}} + \\ 2 \times 16_{\text{TEN}} + \\ 3 \times 4_{\text{TEN}} + \\ 2 \times 1_{\text{TEN}}$$

也可以写成：

$$31232_{\text{FOUR}} = 3 \times 10000_{\text{FOUR}} + \\ 1 \times 1000_{\text{FOUR}} + \\ 2 \times 100_{\text{FOUR}} + \\ 3 \times 10_{\text{FOUR}} + \\ 2 \times 1_{\text{FOUR}}$$

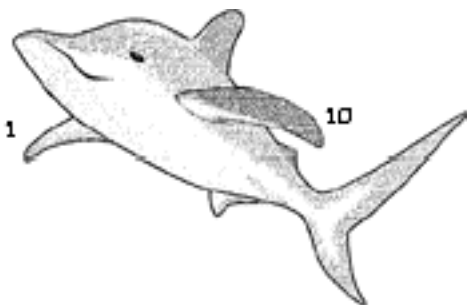
还可以写成：

$$31232_{\text{FOUR}} = 3 \times 4^4 + \\ 1 \times 4^3 + \\ 2 \times 4^2 + \\ 3 \times 4^1 + \\ 2 \times 4^0$$

如果以十进制数的形式计算其结果，就会发现 31232_{FOUR} 等于 878_{TEN} 。

现在，我们要做一个跳跃并且是最远的一跳。假定我们是海豚，并且必须用两鳍来数数。则这个数字系统就是基于2的数字系统或二进制的。这样似乎只需要两个数字，即0和1。

现在，0和1已是你要处理的全部问题，需要练习一下才能习惯使用二进制数。二进制数最大的问题是数字用完得很快。例如，下图是海豚怎样用它的鳍数数的例子：



是的，在二进制中，1后面的数字是10。这是令人惊讶的，但也并不奇怪。无论使用哪种数字系统，当单个位的数字用完时，第一个两位数字都是10。在二进制系统中，可以这样来数数：

0、1、10、11、100、101、110、111、1000、1001、1010、
1011、1100、1101、1110、1111、10000、10001、.....

这些数看起来好像很大，实际上并不是这样。更准确地说二进制数长度增长的速度要快过二进制数增大的速度：

每个人的头的个数为 1_{TEN} 或 1_{TWO}

海豚身上的鳍的个数为 2_{TEN} 或 10_{TWO}

一个大汤匙中包括的小茶匙的数目为 3_{TEN} 或 11_{TWO}

正方形的边数为 4_{TEN} 或 100_{TWO}

每个人一只手的手指数为 5_{TEN} 或 101_{TWO}

一种昆虫的腿数为 6_{TEN} 或 110_{TWO}

一周的天数为 7_{TEN} 或 111_{TWO}

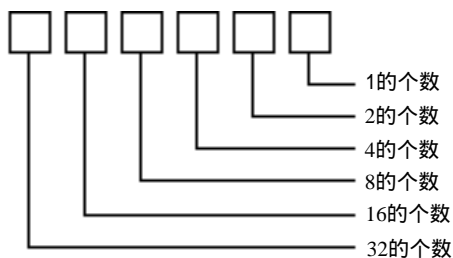
八重奏中音乐家的个数为 8_{TEN} 或 1000_{TWO}

太阳系中的行星（包括冥王星在内）总数为 9_{TEN} 或 1001_{TWO}

牛仔帽重量以加仑计算为 10_{TEN} 或 1010_{TWO}

等等。

在多位二进制数中，数字的位置和2的整数次幂的对应关系为：



因此，任何时候由一个1后跟几个零构成的二进制数一定是2的整数次幂。2的幂与二进制

数中零的个数相等。下面是扩充的 2 的各次幂的表，可用来说明这条规则：

2 的幂	十进制数	八进制数	四进制数	二进制数
2^0	1	1	1	1
2^1	2	2	2	10
2^2	4	4	10	100
2^3	8	10	20	1000
2^4	16	20	100	10000
2^5	32	40	200	100000
2^6	64	100	1000	1000000
2^7	128	200	2000	10000000
2^8	256	400	10000	100000000
2^9	512	1000	20000	1000000000
2^{10}	1024	2000	100000	10000000000
2^{11}	2048	4000	200000	100000000000
2^{12}	4096	10000	1000000	1000000000000

假定有一个二进制数 101101011010，它可以写成：

$$\begin{aligned}
 101101011010_{\text{TWO}} &= 1 \times 2048_{\text{TEN}} + \\
 &\quad 0 \times 1024_{\text{TEN}} + \\
 &\quad 1 \times 512_{\text{TEN}} + \\
 &\quad 1 \times 256_{\text{TEN}} + \\
 &\quad 0 \times 128_{\text{TEN}} + \\
 &\quad 1 \times 64_{\text{TEN}} + \\
 &\quad 0 \times 32_{\text{TEN}} + \\
 &\quad 1 \times 16_{\text{TEN}} + \\
 &\quad 1 \times 8_{\text{TEN}} + \\
 &\quad 0 \times 4_{\text{TEN}} + \\
 &\quad 1 \times 2_{\text{TEN}} + \\
 &\quad 0 \times 1_{\text{TEN}}
 \end{aligned}$$

也可以这样写：

$$\begin{aligned}
 101101011010_{\text{TWO}} &= 1 \times 2^{11} + \\
 &\quad 0 \times 2^{10} + \\
 &\quad 1 \times 2^9 + \\
 &\quad 1 \times 2^8 + \\
 &\quad 0 \times 2^7 + \\
 &\quad 1 \times 2^6 + \\
 &\quad 0 \times 2^5 + \\
 &\quad 1 \times 2^4 + \\
 &\quad 1 \times 2^3 + \\
 &\quad 0 \times 2^2 + \\
 &\quad 1 \times 2^1 + \\
 &\quad 0 \times 2^0
 \end{aligned}$$

如果将各个部分以十进制数的形式相加，得到 $2048+512+256+64+16+8+2 = 2906_{\text{TEN}}$ 。

将二进制数转换成十进制数非常简单，你可能更喜欢借助已准备好的模板进行转换：

× 128	× 64	× 32	× 16	× 8	× 4	× 2	× 1

+ + + + + + + + =

这个模板允许你转换最大长度为 8 的二进制数，但它扩充起来非常容易。使用时，将 8 个二进制数字放到上部的 8 个小盒子中，一个盒子放一个数字。做 8 个乘法运算，将结果分别放到底部的 8 个小盒子中。将 8 个盒子中的数字相加就得到最终结果。下面是将 10010110 转化成十进制数的例子：

1	0	0	1	0	1	1	0
× 128	× 64	× 32	× 16	× 8	× 4	× 2	× 1
128	0	0	16	0	4	2	0

+ + + + + + + + = 150

将十进制数转换成二进制数就没那么直接了。但这里也有一个帮助你将 0 ~ 225 范围内的十进制数转换成二进制数的模板：

÷ 128	÷ 64	÷ 32	÷ 16	÷ 8	÷ 4	÷ 2	÷ 1

实际转化过程要表面上看的麻烦得多，所以一定要仔细按照下面的指导来做。将整个十进制数（应小于等于 225）放在左上角的方格中。用除数（128）去除那个数（被除数），如下图所示。将商写在正下方的盒子中（即左下角的盒子中），余数写在右边的盒子中（即上面一行左数第二个盒子中）。用第一个余数再除以下一个算子 64。依照模板的顺序用同样的方法继续做下去。

记住，每次求得的商只能是 0 或者 1。如果被除数小于除数，商为 0，余数和被除数相等；如果被除数大于除数，商为 1，余数为被除数与除数之差。下面是将 150 转换成二进制数的过程：

150	22	22	22	6	6	2	0
÷ 128	÷ 64	÷ 32	÷ 16	÷ 8	÷ 4	÷ 2	÷ 1
1	0	0	1	0	1	1	0

如果要做两个二进制数的加法或乘法，也许直接采用二进制来做比转化成十进制再做还要简单。这将是真正喜欢二进制数的地方。如果只需记住下面的二进制加法表就可以做加法运算，也就不难想象掌握加法运算该有多快：

+	0	1
0	0	1
1	1	10

用二进制加法表将两个二进制数相加：

$$\begin{array}{r}
 1100101 \\
 + 0110110 \\
 \hline
 10011011
 \end{array}$$

从最右边的一列开始做起：1 加上 0 等于 1；右数第 2 列：0 加上 1 等于 1；第 3 列：1 加上 1 等于 0，进位为 1；第 4 列：1（进位值）加上 0 再加上 0 等于 1；第 5 列：0 加上 1 等于 1；第 6 列：1 加 1 等于 0，进位为 1；第 7 列：1（进位值）加上 1 再加上 0 等于 10。

乘法表比加法表更简单，因为该表可以由两个基本的乘法规则推导出来：零乘以任何数都等于 0，1 与任何数相乘仍是那个数本身：

$$\begin{array}{r}
 \times \quad 0 \quad 1 \\
 0 \quad 0 \quad 0 \\
 1 \quad 0 \quad 1
 \end{array}$$

下面是 13_{TEN} 与 11_{TEN} 以二进制数的形式做乘法的过程：

$$\begin{array}{r}
 1101 \\
 \times 1011 \\
 \hline
 1101 \\
 1101 \\
 0000 \\
 1101 \\
 \hline
 10001111
 \end{array}$$

最后结果是 143_{TEN} 。

人们在使用二进制数的时候通常将它们写成带有前导零的形式（即第一个 1 的左边有零）。例如，0011 而不写成 11。这些零不会改变数字的值，只是起到一些装饰作用。例如，下面是二进制的前 16 个数以及和它们等值的十进制数：

二进制数	十进制数
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

让我们再仔细看看这些二进制数字。考虑一下这 4 个垂直列中每一列的 0 和 1，注意它们在一列中自上而下是以怎样的规律变化的：

- 最右边一列一直在0和1之间相互替换。
- 右数第2列在两个0和两个1之间相互替换。
- 右数第3列在四个0和四个1之间相互替换。
- 右数第4列在八个0和八个1之间相互替换。

这是很有规律的，难道不是吗？事实上，只要再重复这 16 个数字并且在每个数字的前面放一个1就可以很容易地写出后面的 16 个数字：

二进制数	十进制数
10000	16
10001	17
10010	18
10011	19
10100	20
10101	21
10110	22
10111	23
11000	24
11001	25
11010	26
11011	27
11100	28
11101	29
11110	30
11111	31

下面是看待这些数字的另一种方式：在数二进制数的时候，最右边的数字（也称最低位数字）是在0和1之间变化的。当它每次从1变到0时，右数第二位数字（也称次低位数字）也要发生变化，或者从0变到1，或者从1变到0。每次只要有一个二进制数位的值由1变到0，紧挨着的高位数字也会发生变化，要么从0变到1，要么从1变到0。

我们在写十进制中比较大的数字时，通常每三个数字之间留一点儿空隙，这样，我们一看就知道这个数的大概数值。例如，当你看到数字 12000000时，你可能不得不去数其中0的个数，但如果看到的是 12 000 000，则马上就能知道是一亿两千万。

二进制数的位长度增加得特别快。例如，一亿两千万的二进制表示为：101101110001101100000000。为了让它更易读，通常是每四个数字之间用连字符或空格来分开。例如；1011-0111-0001-1011-0000-0000或101101110001101100000000。本书的后面会讲到更简单的二进制数的表示方法。

通过将数字系统减少至只有0和1两个数字的二进制数字系统，我们已经在能够接受的范围内做了深入的讨论。不可能找到比二进制数字系统更简单的数字系统了。二进制数字系统架起了算术与电之间的桥梁。前面各章中，我们所看到的开关、电线、灯泡、继电器等物体都可以表示二进制数0和1：

电线可以表示二进制数字。有电流流过电线代表二进制数字1；如果没有，则代表二进制数字0。

开关可以表示二进制数字。如果开关闭合，代表二进制数字1；如果开关断开，代表二进制数字0。

灯泡可以表示二进制数字。如果灯泡亮着，代表二进制数字1；如果没亮，代表二进制数

字0。

电报继电器可以表示二进制数字。继电器闭合，代表二进制数字 1；继电器断开，代表二进制数字0。

二进制数与计算机密切相关！

大约在1948年，美国数学家John Wilder Tukey（生于1915年）提前认识到二进制数将在未来几年中随着计算机的流行而发挥更大的作用。他决定创造一个新的、更短的词来代替使用起来很不灵活的五音节词——binary digit。他曾经考虑用bigit或binit，但最后还是选用了短小、简单、精巧且非常可爱的单词bit(比特)来代替binary digit这个词。