



第二讲 计算机编码



河北升学规划行业联盟

主讲人：吕毅飞



主要内容



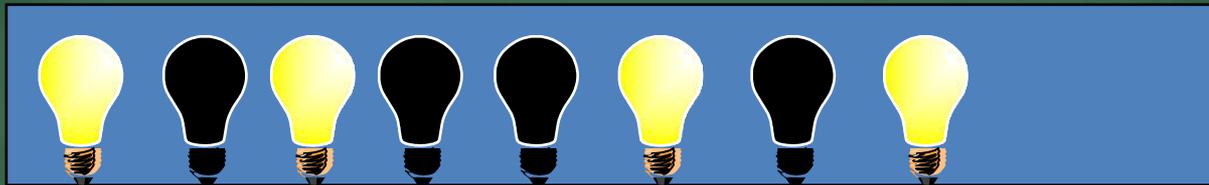
- 2.1 计算机存储信息的单位
- 2.2 数值在计算机中的表示
- 2.3 整数与浮点数的表示
- 2.4 八进制与十六进制
- 2.5 字符信息的表示
- 2.6 中文信息编码及其他编码



计算机与二进制



在计算机内部，数据的存储与处理都采用二进制



电压：高低
电流：有无
门电路：导通/截止

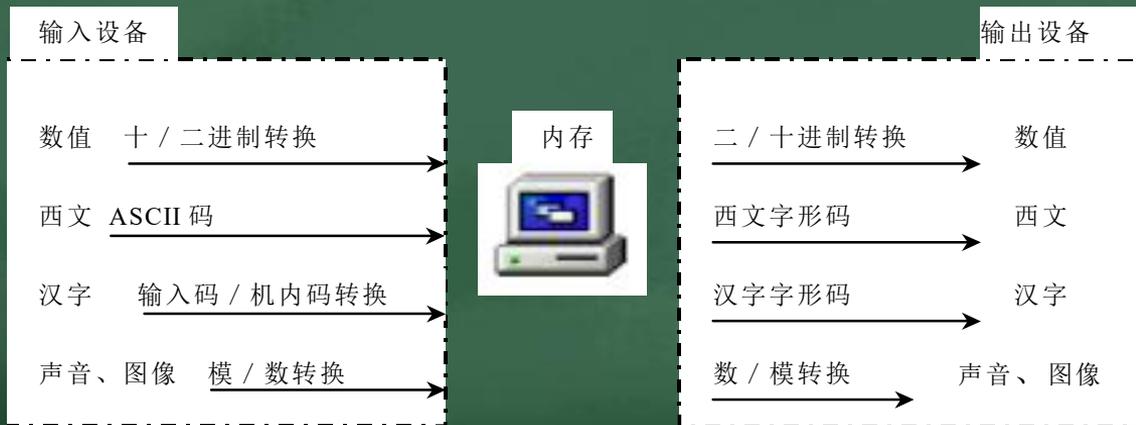


利用二进制对信息进行编码



说明：

用0和1的组合表示数值、英文字母、汉字、色彩和声音



2.1 计算机存储信息的基本单位

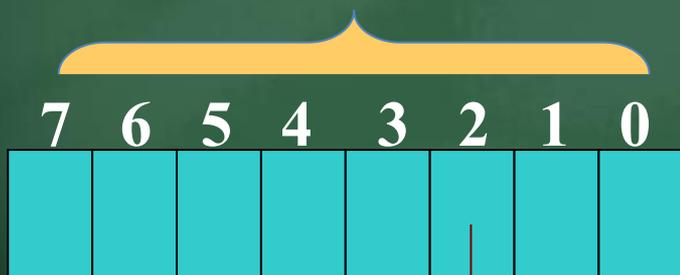


说明:



位 (Bit) 是计算机表示数据的最小单位, 表示一个0或1。
字节 (Byte) 是表示数据的基本单位, 由八个位组成。

字节BYTE



位bit

$$1\text{KB}=2^{10}\text{B}=1024\text{B}$$

$$1\text{MB}=2^{10}\text{KB}=1024\text{KB}$$

$$1\text{GB}=2^{10}\text{MB}=1024\text{MB}$$

$$1\text{TB}=2^{10}\text{GB}=1024\text{GB}$$



计算机内，数据表示的有限性



说明：

数据在计算机内部通常用8、16、32位的等八的整数倍来进行表示，是有限的

存储单位	可表示的状态数
1 bit	0和1两种状态
1 Byte	2^8 ，共256种状态
2 Byte	2^{16} ，共65526种状态
4 Byte	2^{32} 种状态



2.2数值在计算机内的表示



计算机表示数值型数据需考虑

1.确定字的长度

数据长度以字节为单位计算

2.确定字的符号

数的最高位（左边第一位）来表示数的正负号，并约定“0”表示正号，“1”表示负号

3.小数点的表示方法

计算机隐含规定小数点的位置。根据小数点的位置是否固定，数可以分为定点和浮点两种表示方法



2.3 整数与浮点数的表示



进位计数制说明：

任何n进制数均可表示为：数码乘以各自权的累加

十进制数的进位计数制表示：

$$7531 = 7 \times 10^3 + 5 \times 10^2 + 3 \times 10^1 + 1 \times 10^0$$

二进制数的进位计数制表示：

$$1101 = \textcircled{1} \times \mathbf{2}^3 + 1 \times \textcircled{2}^2 + 0 \times \mathbf{2}^1 + 1 \times \textcircled{2}^0$$

数码

基数

位权



进位计数制的基本概念



1. 数码

一组用来表示某种数制的符号

如十进制采用 $0, 1, 2, \dots, 9$, 十六进制的 $0, \dots, 9, A, \dots, F$

2. 基数

数制所用的数码的个数, 如十进制有10个数码, 则基数为10

3. 位权

用于表示不同位置上的数的权值, 由基数和对应指数构成



各种进制数的表示



进位制	二进制	八进制	十进制	十六进制
规则	逢二进一	逢八进一	逢十进一	逢十六进一
基数	$r = 2$	$r = 8$	$r = 10$	$r = 16$
基本符号	0,1	0,1,⋯,7	0,1,⋯,9	0,1,⋯,9,A,B,⋯,F
权	2^i	8^i	10^i	16^i
字母代号	B	O	D	H



十进制的二进制表示



十进制	二进制	十进制	二进制
0	00000000	8	00001000
1	00000001	9	00001001
2	00000010	10	00001010
3	00000011	11	00001011
4	00000100	12	00001100
5	00000101	13	00001101
6	00000110	14	00001110
7	00000111	15	00001111



十进制转化为二进制的方法

说明：任意十进制转为二进制的方法：除2取余

* $(26)_D = (?)_B$

	余数	
2 26	0	二进制的低位
2 13	1	↑
2 6	0	
2 3	1	
2 1	1	
0		



负数在计算机内部的表示



说明:

数的正负用高位字节的最高位来表示，用0表示正，用1表示负，其余位表示数值。这种数称为机器数。

67的原码表示:

0	1	0	0	0	0	1	1
---	---	---	---	---	---	---	---

-67的原码表示:

1	1	0	0	0	0	1	1
---	---	---	---	---	---	---	---



当符号位参与运算



说明：

当最高位符号位时，符号位也要参与运算，若用原码表示负数，将引起运算结果错误。

$$3 + -2 = ?$$

3	0	0	0	0	0	0	1	1
-2	1	0	0	0	0	0	1	0



思考



由于 $1 + -1$ 的结果为0。那么，只要找到-1在计算机内部的正确表示方法，就可推导出负数在计算机内部的正确表示方法。

1	0	0	0	0	0	0	0	1
+	-1	?	?	?	?	?	?	?
<hr/>								
0	0	0	0	0	0	0	0	0



思考



$$\begin{array}{r} 1 \quad 0 \quad 1 \\ + \quad -1 \quad 1 \\ \hline 1 \quad 0 \end{array} =$$

溢出的位，自然丢失

1111 1111就是-1在机器内的正确表示，即补码



如何找出负数的补码?



-1的原码

1	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

-1的反码

1	1	1	1	1	1	1	0
---	---	---	---	---	---	---	---

-1的补码

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

保持符号位
不变, 其余
各位取反

反码基础上
加1



带符号数的表示



假定一个数在机器中占用8位。

(1) 原码

$[X]_{\text{原}} =$

→ 0X	$0 \leq X$	+7: 0000111	+0: 0000000
→ 1 X	$X <= 0$	-7: 1000111	-0: 1000000

(2) 反码

$[X]_{\text{反}} =$

→ 0X	$0 \leq X$	+7: 0000111	+0: 0000000
→ 1 $\overline{ X }$	$X <= 0$	-7: 1111100	-0: 1111111

$[X]_{\text{反}} =$

→ 0X	$0 \leq X$	+7: 0000111	+0: 0000000
→ 1 $\overline{ X } + 1$	$X <= 0$	-7: 1111101	-0: 0000000



数的各种码制



十进制数	原码	反码	补码
-8	10001000	11110111	11111000
-7	10000111	11111000	11111001
-6	10000110	11111001	11111010
-5	10000101	11111010	11111011
-4	10000100	11111011	11111100
-3	10000011	11111100	11111101
-2	10000010	11111101	11111110
-1	10000001	11111110	11111111
0	00000000	00000000	00000000
1	00000001	00000001	00000001
2	00000010	00000010	00000010
3	00000011	00000011	00000011
4	00000100	00000100	00000100
5	00000101	00000101	00000101
6	00000110	00000110	00000110
7	00000111	00000111	00000111



定点数的表示



小数点

定点整数

无符号位

小数点

S

定点小数



浮点数



浮点数的指数表示形式：

$$N = \pm d \times 10^{\pm p}$$

其中， d 是尾数，前面的“ \pm ”表示数符；
 p 是阶码，前面的“ \pm ”表示阶符。

同样，任意二进制浮点数的表示形式：

$$N = \pm d \times 2^{\pm p}$$



十进制小数转化为二进制



说明:

采用乘2取整法：用十进制小数乘2，当积为0或达到要求的精度时，将结果的整数部分由上而下排列。

例：将0.125转化为二进制形式

	0.125	
	$\times 2$	
0	0.250	
	$\times 2$	
0	0.500	
	$\times 2$	
1	1.000	小数部分为0 转换结束

小数点·整数

↓



2.4 八进制与十六进制



说明：

各种进制数按照以下公式进制转换：
(该公式只用于正数)

$$K = \sum_{i=-m}^{n-1} a_i \times r^i$$

用八进制和十六进制简写二进制时：

因为 $2^3=8$ ，所以每一位八进制数可以用一个3位二进制数表示；

因为 $2^4=16$ ，所以每一位十六进制数可以用一个4位二进制数表示



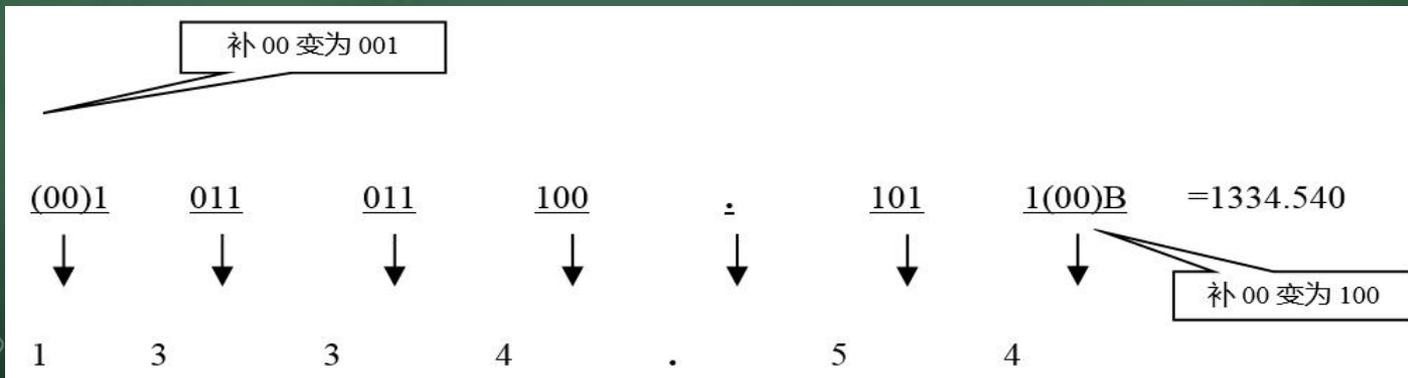
二进制转化为八进制



说明:

从小数点开始，整数部分从右向左3位一组，小数部分从左向右3位一组，若不足3位用“0”补足

例：将1011011100.1011B转换成八进制数



八进制转化为二进制



说明:

以小数点为界，向左或向右每一位八进制数用相应的3位二进制数取代，若中间位不足3位，在前面用0补足

例：将 $(2374.52)_8$ 转换为二进制数

$$(2374.52)_8 = 10011111100.10101B$$



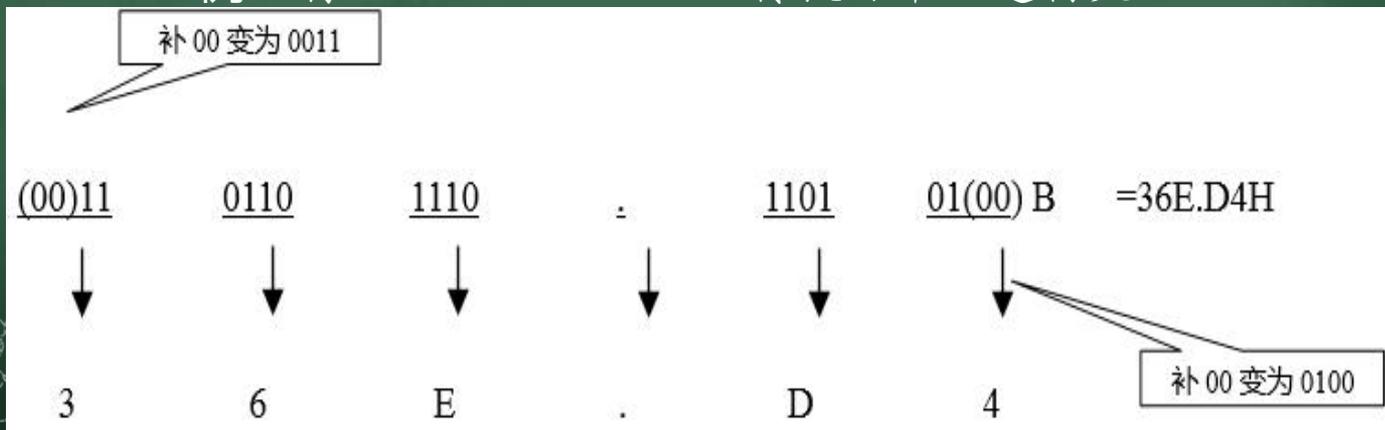
二进制转化为十六进制



说明:

从小数点开始，整数部分从右向左每4位一组；
小数部分从左向右每4位一组，不足4位用0补足

例：将1101101110.110101B转换为十六进制数



十六进制转化为二进制



说明:

以小数点为界，向左或向右每一位十六进制数用相应的4位二进制数取代

例：将36EF.A2H为二进制数

3	6	E	F	.	A	2H
0011	0110	1110	1111		1010	0010

=0011011011101111.10100010B



八进制和十六进制所对应的码制



八进制	0	1	2	3	4	5	6	7
二进制	000	001	010	011	100	101	110	111
十六进制	0	1	2	3	4	5	6	7
二进制	0000	0001	0010	0011	0100	0101	0110	0111
十六进制	8	9	A	B	C	D	E	F
二进制	1000	1001	1010	1011	1100	1101	1110	1111



2.5 字符表示的信息



说明:

ASCII码是ANSI（美国国家标准机构）制定的“美国标准信息交换代码”。

包括：128个常用字符，用7位二进制编码，从0到127。

控制字符：0~32, 127；普通字符：94个。

换行	0AH	10
回车	0DH	13
空格	20H	32
‘0’ ~ ‘9’	30H~39H	48~57
‘A’ ~ ‘Z’	41H~5AH	65~90
‘a’ ~ ‘z’	61H~7AH	97~122



2.6 中文信息编码及其他编码



汉字输入码:



音码类: 全拼、双拼、微软拼音、搜狗拼音等;

形码类: 五笔字型法、郑码输入法。

汉字国标码(GB2318-80):

每个汉字两个字节;

一级汉字: 3755个; 二级汉字: 3008个。

汉字 94×94 的矩阵, 即94个区和94个位, 由区号和位号构成区位码。

汉字的国标码与区位码的关系:

每个汉字的区号和位号各加32(20H)就构成了国标码

加32的原因: 为了与ASCII码兼容, 每个字节值大于32 (0~32为非图形字符码值)

区号	位号
----	----



机内码



说明:

汉字在设备或信息处理系统内部最基本的表达形式。

为了在计算机内部能够区分是汉字编码还是ASCII码，将国标码每个字节最高位设置为1(80H)。

区位码	国标码	机内码
中 (3630)H	(56 50)H=(01010110 01010000)B	(11010110 11010000)B=(D6 D0)H
华 (1B0A)H	(3B 2A)H=(00111011 00101010)B	(10111011 10101010)B=(BB AA)H

三种码之间的关系:

汉字机内码=汉字国标码+8080H=区位码+A0A0H

国标码=区位码+2020H



汉字字形码



点阵:



汉字字形点阵的代码; 有 16×16 、 24×24 、 32×32 、

48×48 等编码、存储方式简单、无需转换直接输出

放大后产生的效果差

矢量:

存储的是描述汉字字形的轮廓特征

矢量方式特点正好与点阵相反

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	十六进制码
0							●	●									0 3 0 0
1							●	●									0 3 0 0
2							●	●									0 3 0 0
3							●	●						●			0 3 0 4
4	●	●	●	●	●	●			●	●	●	●	●	●	●		F F F E
5							●	●									0 3 0 0
6							●	●									0 3 0 0
7							●	●									0 3 0 0
8							●	●									0 3 0 0
9							●	●	●								0 3 8 0
10							●	●		●							0 6 4 0
11											●						0 C 2 0
12												●	●				1 8 3 0
13													●	●			1 0 1 8
14														●	●		2 0 0 C
15	●	●													●	●	C 0 0 7



汉字的处理流程



说明：输入设备将外码送入计算机，再由汉字系统将其转换成内码存储、传送和处理，当需要输出时再由汉字系统调用字库中汉字的字形码得到结果



其他汉字编码



1.Unicode码: 另一国际标准, 采用双字节编码统一地表示世界上的主要文字。其字符集内容与UCS的BMP相同

2.GBK码: 2字节表示一个汉字。第一字节从81H~FEH, 最高位为1; 第二字节从40H~FEH, 第二字节的最高位不一定是1

3.BIG5编码: 台湾、香港地区普遍使用的一种繁体汉字的编码标准, 包括440个符号, 一级汉字5401个、二级汉字7652个



谢谢观赏

