

计算机组织结构

# 3 数据的机器级表示

任桐炜

2021年9月14日



南京大學  
NANJING UNIVERSITY

# 教材对应章节



## 第2章 数据的机器级表示



## 第9章 计算机算术

# 信息的二进制编码

- 在冯·诺依曼结构中，所有信息（代码和数据）都采用二进制编码
  - 编码**：用少量简单的**基本符号**对复杂多样的信息进行一定**规律**的组合
- 采用二进制的原因
  - 多种物理器件可以表示两种稳定的状态，用于表示0和1
  - 二进制的编码和运算规则简单
  - 1和0可以对应逻辑命题中的“真”和“假”
- K位的二进制编码至多表示 $2^k$ 个不同的值

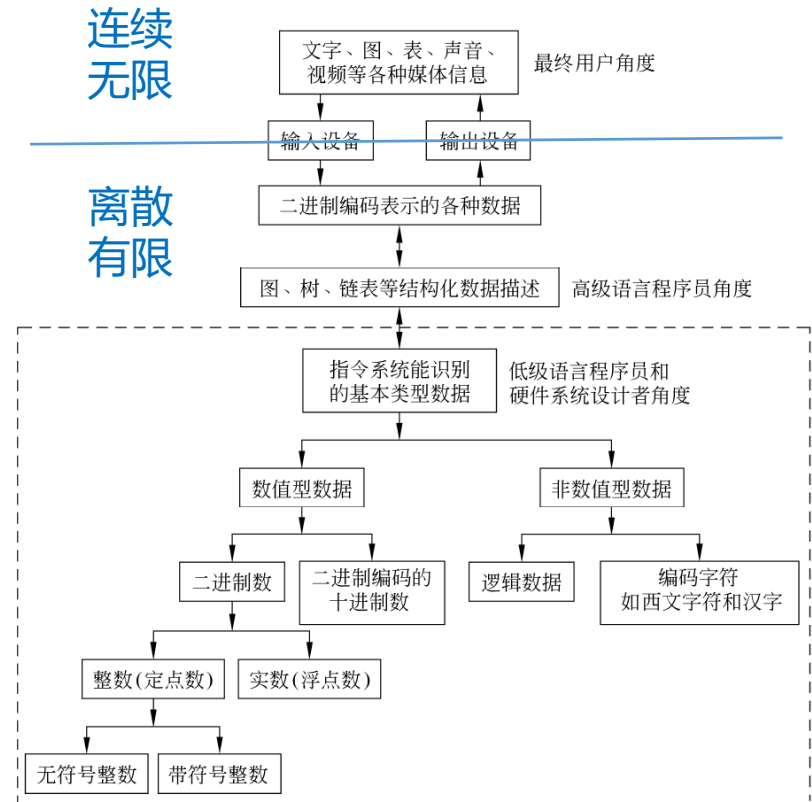


图 2.1 计算机外部信息与内部数据的转换

# 整数的二进制数表示

- 无符号整数
- 有符号整数：原码，反码，移码，补码
  - 原码、反码、移码在进行加法运算时都会造成不必要的硬件需求，因此目前计算机中普遍使用补码
  - 二进制补码的运算
  - 二进制-十进制转换



# 补码表示法的优势

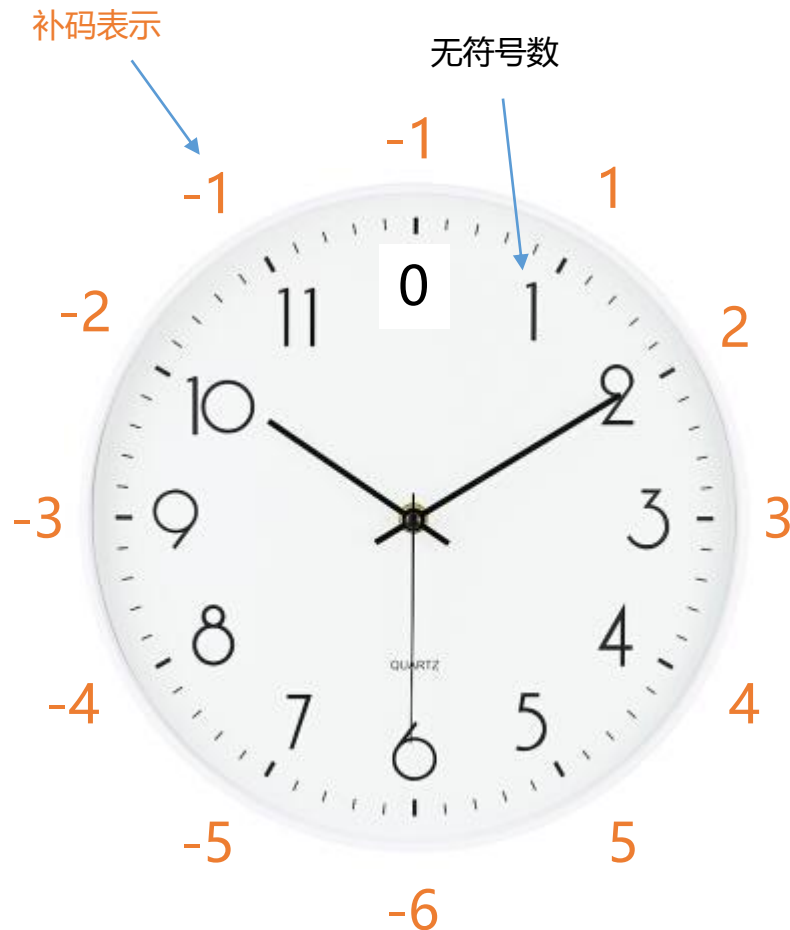
- 补码表示法 vs. 原码表示法

	补码表示法	原码表示法
$\begin{array}{r} 9 \\ + 8 \\ \hline 17 \end{array}$	$\begin{array}{r} 0000\ 1001 \\ + 0000\ 1000 \\ \hline 0001\ 0001 \end{array} \quad 17$	$\begin{array}{r} 0000\ 1001 \\ + 0000\ 1000 \\ \hline 0001\ 0001 \end{array} \quad 17$
$\begin{array}{r} 9 \\ + -8 \\ \hline 1 \end{array}$	$\begin{array}{r} 0000\ 1001 \\ + 1111\ 1000 \\ \hline 10000\ 0001 \end{array} \quad 1$	$\begin{array}{r} 0000\ 1001 \\ + 1000\ 1000 \\ \hline 1001\ 0001 \end{array} \quad -17$

无论同号还是异号都可以直接相加



# 补码表示



- 类比：时钟
  - 表示范围：0 ~ 11  $\rightarrow$  -6 ~ 5
  - 6 ~ 11  $\rightarrow$  -6 ~ -1 （变化：减12）
- 补码（相对于无符号数）
  - 000...000 ~ 011...111：表示的值不变
  - 100...000 ~ 111...111：表示的值由  $2^{(k-1)} \sim 2^k - 1$  变为  $-2^{(k-1)} \sim -1$ 
    - 真值为原来的无符号数对应的真值减去  $2^k$  （取反加1的由来）

# 补码的真值

- 补码的真值

$$[X]_C = X_n X_{n-1} \dots X_2 X_1$$

$$X = -X_n \times 2^{n-1} + \dots + X_2 \times 2^1 + X_1 \times 2^0$$

(这个式子的由来需要分情况 $X_n$ 为正和为负讨论)

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

-128

+2

+1

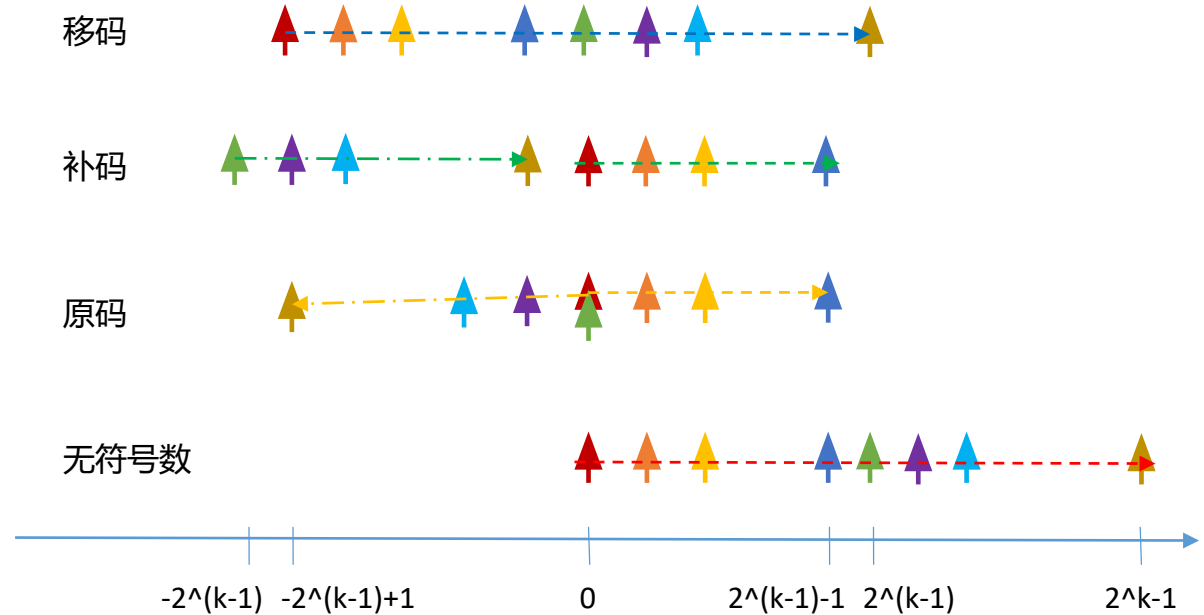
= -125

- 值的范围

$$-2^{n-1} \leq X \leq 2^{n-1} - 1$$



# 扩展：不同的整数编码





# 浮点数的二进制数表示

- 实数表示
- 定点表示法表示值的范围是有限制的
- 科学计数法

$$\pm S \times B^E$$

- $\pm$  (符号) : 正或负
- S (尾数/有效值)
- B (底/基) : 对所有的数都是相同的, 不需要存储的 (隐含的)
- E (阶码/指数)



# 规格化数

- 任何浮点数都能以多种样式来表示

$$0.1110 \times 2^5, 1110 \times 2^2, 0.01110 \times 2^6$$

- 规格化表示

$$\pm 1. bbb \dots b \times 2^E$$

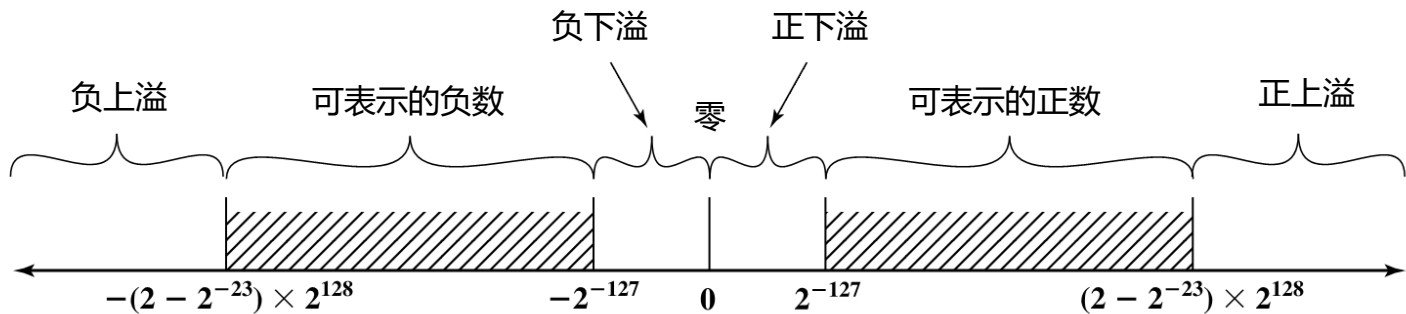
- 符号总是位于字的第1位
- 尾数 (S) 的第1位总是1, 不需要存于尾数字段中 (默认省略)
- 阶码 (E) 的真实值加127后, 再存入阶码字段中 (移码)
- 底 (B) 默认为2



# 规格化数的值的范围

- 值的范围

- 介于  $-(2 - 2^{-23}) \times 2^{128}$  和  $-2^{-127}$  之间的负数
- 介于  $2^{-127}$  和  $(2 - 2^{-23}) \times 2^{128}$  之间的正数



[袁睿, 131250088]



# 规格化数的变化

- 对于一定长度的规格化数，表示范围和精度之间存在权衡
  - 增加阶码 (E) 位数：扩大表示范围，降低表示精度
  - 增加尾数 (S) 位数：提高表示精度，减少表示范围
  - 采用更大的底 (B)：实现更大的范围，降低表示精度

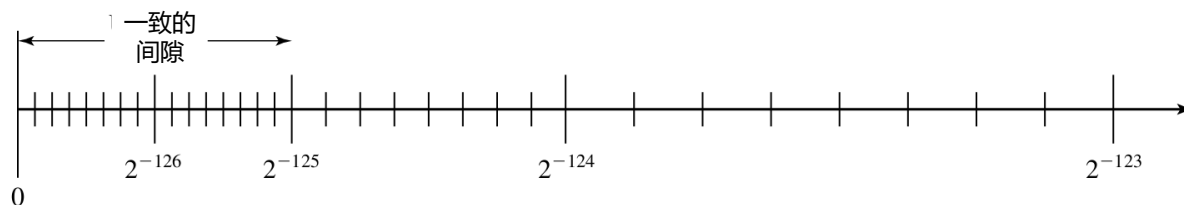


# 非规格化数

- 处理规格化数中的下溢情况
- 当结果的阶值太小时，通过右移进行非规格化；每次右移阶值增，直到阶值落在可表示范围内



(a)没有非规格化数的32位格式



(b)有非规格化数的32位格式

# IEEE 754 标准

- 定义32位的单精度和64位的双精度两种格式



- 定义两种拓展格式
  - 在阶值字段提供更多的位（拓展范围）和在有效值字段提供更多的位（拓展精度）
  - 减少过度舍入误差和计算过程中溢出的机会



# IEEE 754 标准 (cont.)

- 格式参数

参数	格式			
	单精度	单精度拓展	双精度	双精度拓展
字宽（位数）	32	$\geq 43$	64	$\geq 79$
阶值位宽（位数）	8	$\geq 11$	11	$\geq 15$
阶值偏移量	127	未指定	1023	未指定
最大阶值	127	$\geq 1023$	1023	$\geq 16383$
最小阶值	-126	$\leq -1022$	-1022	$\leq -16382$
数的范围（底为10）	$10^{-38}, 10^{+38}$	未指定	$10^{-308}, 10^{+308}$	未指定
有效值位宽（位数）	23	$\geq 31$	52	$\geq 63$
阶值的数目	254	未指定	2046	未指定
小数的数目	$2^{23}$	未指定	$2^{52}$	未指定
值的数目	$1.98 \times 2^{31}$	未指定	$1.99 \times 2^{63}$	未指定



# IEEE 754 标准 (cont.)

符号: 0

用法: 表示未初始化的值, 用于捕获异常

符号: 1

用法: 表示未定义的算术结果, 如除数等于0

	单精度 (32位)				双精度 (64位)			
	符号	移码阶值	小数	值	符号	移码阶值	小数	值
正零	0	0	0	0	0	0	0	0
负零	1	0	0	-0	1	0	0	-0
正无穷大	0	255 (all 1s)	0	$\infty$	0	2047 (all 1s)	0	$\infty$
负无穷大	1	255 (all 1s)	0	$-\infty$	1	2047 (all 1s)	0	$-\infty$
静默式非数	0 or 1	255 (all 1s)	$\neq 0$	NaN	0 or 1	2047 (all 1s)	$\neq 0$	NaN
通知式非数	0 or 1	255 (all 1s)	$\neq 0$	NaN	0 or 1	2047 (all 1s)	$\neq 0$	NaN
正的规格化非零数	0	$0 < e < 255$	f	$2^{e-127}(1.f)$	0	$0 < e < 2047$	f	$2^{e-1023}(1.f)$
负的规格化非零数	1	$0 < e < 255$	f	$-2^{e-127}(1.f)$	1	$0 < e < 2047$	f	$-2^{e-1023}(1.f)$
正的非规格化数	0	0	$f \neq 0$	$2^{e-126}(0.f)$	0	0	$f \neq 0$	$2^{e-1022}(0.f)$
负的非规格化数	1	0	$f \neq 0$	$-2^{e-126}(0.f)$	1	0	$f \neq 0$	$-2^{e-1022}(0.f)$





# IEEE 754 标准 (cont.)

- 例子

$$0.5 = 0.100\dots0B = (1.00\dots0)_2 \times 2^{-1}$$

0 01111110 000...00 (23)

$$-0.4375 = -0.01110\dots0B = - (1.110\dots0)_2 \times 2^{-2}$$

1 01111101 110...00 (21)



# 二进制编码的十进制数表示

- 浮点运算的问题
  - 精度限制
  - 转换成本高
- 应用需要
  - 长数字串的计算：会计， .....
- 解决方法
  - 用4位二进制编码十进制 (BCD) 表示0, 1, ..., 9, 直接计算



# 二进制编码的十进制数表示 (cont.)

- 自然BCD码 (NBCD, 8421 码)
  - 0 ~ 9: 0000 ~ 1001
  - 符号: 使用四个最高有效位
    - 正: 1100 / 0
    - 负: 1101 / 1
  - 例子
    - +2039: **1100** 0010 0000 0011 1001 / **0** 0010 0000 0011 1001
    - -1265: **1101** 0001 0010 0110 0101 / **1** 0001 0010 0110 0101
- 其他BCD码
  - 2421, 5211, 4311, ...



# 总结

- 信息的二进制编码
- 整数的二进制表示
  - 补码表示的优势，表示方法，真值计算
  - 不同的整数二进制表示
- 浮点数的二进制表示
  - 浮点数表示方法，规格化数，非规格化数，IEEE 754标准
- 二进制编码的十进制数表示
  - NBCD码表示方法



# 谢谢

rentw@nju.edu.cn



南京大學  
NANJING UNIVERSITY