本节内容

浮点数标准

IEEE 754

跟王者农药学发音

双杀 double kill——英：ˈdʌbl kɪl，美：ˈdʌbl kɪl。

三杀 triple kill——英：ˈtrɪpl kɪl，美：ˈtrɪpl kɪl。

四杀——quadra kill——（英/美）kwɒdrə kɪl。

五杀——penta kill——英：pɛntə kɪl，美：ˈpɛntə kɪl。

triple

英 [ˈtrɪpl]　美 [ˈtrɪpl]

adj. 三倍的；三方的

n. 三倍数；三个一组

vi. 增至三倍

vt. 使成三倍

移码

移码：　补码的基础上将符号位取反。注意：移码只能用于表示整数

x= +19D

$[x]_{原}$ =**0**,0010011

$[x]_{反}$ =**0**,0010011

$[x]_{补}$ =**0**,0010011

$[x]_{移}$ =**1**,0010011

定点整数的表示

x= -19D

$[x]_{原}$ =**1**,0010011

$[x]_{反}$ =**1**,1101100

$[x]_{补}$ =**1**,1101101

$[x]_{移}$ =**0**,1101101

移码

真值(十进制)	补码	移码
-128	1000 0000	0000 0000
-127	1000 0001	0000 0001
-126	1000 0010	0000 0010
…	…	…
-3	1111 1101	0111 1101
-2	1111 1110	0111 1110
-1	1111 1111	0111 1111
0	0000 0000	1000 0000
1	0000 0001	1000 0001
2	0000 0010	1000 0010
3	0000 0011	1000 0011
…	…	…
124	0111 1100	1111 1100
125	0111 1101	1111 1101
126	0111 1110	1111 1110
127	0111 1111	1111 1111

真值增大

移码的定义：移码=真值+偏置值

此处8位移码的偏置值=128D=1000 0000B，即2^{n-1}

真值 -127 = -1111111B
移码 = -1111111 + 10000000 = 0000 0001

真值 -3= -11B
移码 = -11 + 10000000 = 0111 1101

真值 +0 = +0
移码 = +0 + 10000000 = 1000 0000

真值 +3 = +11B
移码 = +11 + 10000000 = 1000 0011

真值 +127 = +1111111B
移码 = +1111111 + 10000000 = 1111 1111

移码

偏置值=2^{n-1}

偏置值=$2^{n-1}-1$

真值(十进制)	补码	移码	移码
-128	1000 0000	0000 0000	1111 1111
-127	1000 0001	0000 0001	0000 0000
-126	1000 0010	0000 0010	0000 0001
…	…	…	…
-3	1111 1101	0111 1101	0111 1100
-2	1111 1110	0111 1110	0111 1101
-1	1111 1111	0111 1111	0111 1110
0	0000 0000	1000 0000	0111 1111
1	0000 0001	1000 0001	1000 0000
2	0000 0010	1000 0010	1000 0001
3	0000 0011	1000 0011	1000 0010
…	…	…	…
124	0111 1100	1111 1100	1111 1011
125	0111 1101	1111 1101	1111 1100
126	0111 1110	1111 1110	1111 1101
127	0111 1111	1111 1111	1111 1110

无符号数255

无符号数1

真值增大

无符号数254

移码的定义：移码=真值+偏置值

偏置值可以取其他值

令偏置值=127D=0111 1111B，即$2^{n-1}-1$

真值 -128 = -1000 0000B
移码 = -1000 0000 + 01111111 = 1111 1111

真值 -127 = -111 1111B
移码 = -111 1111 + 01111111 = 0000 0000

真值 -126 = -111 1110B
移码 = -111 1110 + 01111111 = 0000 0001

真值 +0 = +0
移码 = +0 + 01111111 = 0111 1111

真值 +127 = +1111111B
移码 = +111 1111 + 01111111 = 1111 1110

IEEE 754标准

阶码全1、全0
用作特殊用途

尾数部分，$\boxed{\text{用原码表示}}$ 隐藏表示最高位1

m_s	E	M
数符	阶码部分，$\boxed{\text{用移码表示}}$	尾数数值位

表示尾数1.M

偏置值=$2^{n-1}-1$

单精度浮点型

双精度浮点型

真值正常范围：
-126~127

	类型	数符	阶码	尾数数值	总位数	偏置值	
						十六进制	十进制
float	短浮点数	1	8	23	32	7FH	127
double	长浮点数	1	11	52	64	3FFH	1023
long double	临时浮点数	1	15	64	80	3FFFH	16383

float 1000 0001 1000 1010 0101 0000 1000 0000

double 1000 0001 1100 0010 0101 0000 1000 0000 0000 0000 0001 1111 0000 0000 0000 0000

规格化的短浮点数的真值为：$(-1)^s \times 1.M \times 2^{E-127}$
规格化长浮点数的真值为：$(-1)^s \times 1.M \times 2^{E-1023}$

阶码真值=移码-偏移量

IEEE 754标准

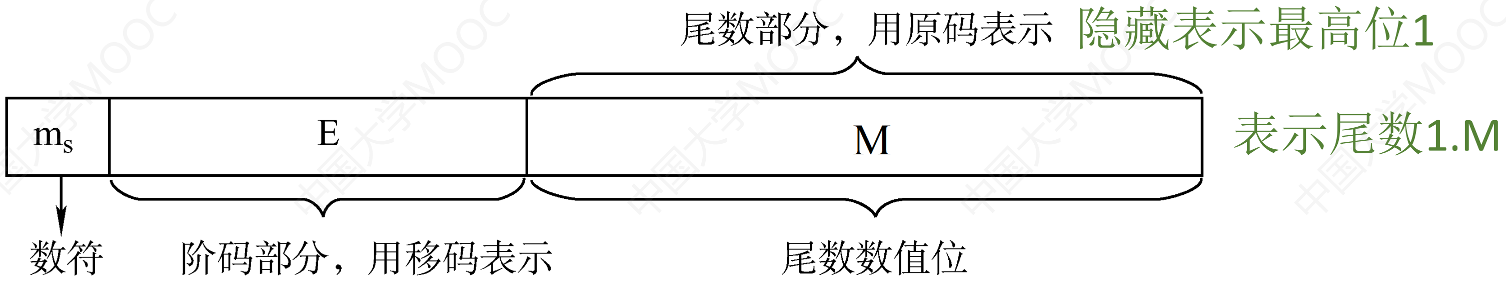
例：将十进制数 -0.75 转换为 IEEE 754 的单精度浮点数格式表示。

$(-0.75)_{10} = (-0.11)_2 = (-1.1)_2 \times 2^{-1}$

数符 = 1
尾数部分 = .1000000.....　（隐含最高位1）
阶码真值 = -1
单精度浮点型偏移量 = 127D
移码 = 阶码真值+偏移量 = -1 + 111 1111 = 0111 1110 （凑足8位）

→ 1 01111110 10000000000000000000000

IEEE 754标准



例：IEEE 754 的单精度浮点数 C0 A0 00 00 H 的值时多少。

C0 A0 00 00 H → 1100 0000 1010 0000 0000 0000 0000 0000

数符 = 1 → 是个负数

尾数部分 = .0100.... （隐含最高位1） → 尾数真值 = $(1.01)_2$

移码 = 10000001， 若看作无符号数= 129D

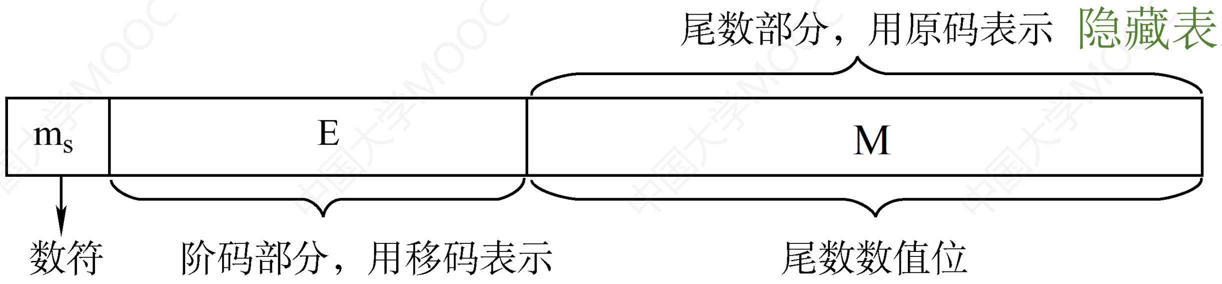
单精度浮点型偏移量 = 127D

阶码真值= 移码 - 偏移量 = 1000 0001 – 111 1111 = $(0000\ 0010)_2 = (2)_{10}$

→ 浮点数真值 = $(-1.01)_2 \times 2^2 = -1.25 \times 2^2 = -5.0$

IEEE 754标准

天然地完成了“规格化”



隐藏表示最高位1

表示尾数1.M

IEEE 754 单精度浮点型能表示的最小绝对值、最大绝对值是多少？

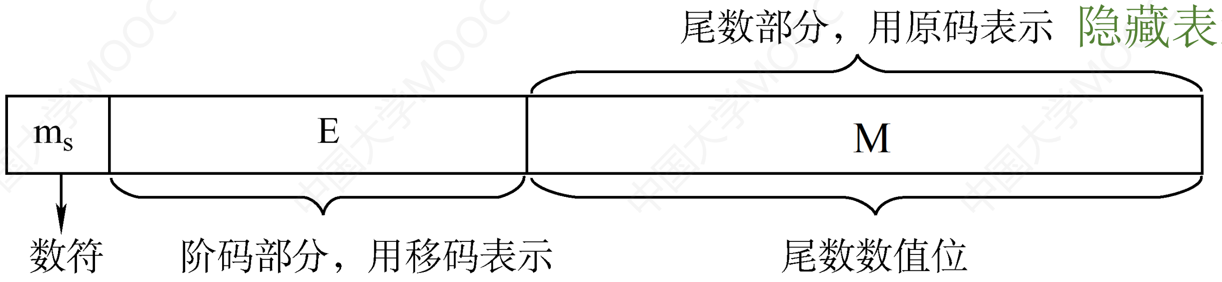
若要表示的数绝对值还要更小，怎么办？

最小绝对值：尾数全为0，阶码真值最小-126，对应移码机器数 0000 0001
此时整体的真值为 $(1.0)_2 \times 2^{-126}$

最大绝对值：尾数全为1，阶码真值最大 127，对应移码机器数 1111 1110
此时整体的真值为 $(1.111…11)_2 \times 2^{127}$

格　式	规格化的最小绝对值	规格化的最大绝对值
单精度	E=1，M=0：$1.0 \times 2^{1-127}=2^{-126}$	E=254，M=.11…1：$1.11…1 \times 2^{254-127}=2^{127} \times (2-2^{-23})$
双精度	E=1，M=0：$1.0 \times 2^{1-1023}=2^{-1022}$	E=2046，M=.11…1：$1.11…1 \times 2^{2046-1023}=2^{1023} \times (2-2^{-52})$

IEEE 754标准



隐藏表示最高位1

表示尾数1.M

阶码全1、全0
用作特殊用途

IEEE 754 单精度浮点型能表示的最小绝对值、最大绝对值是多少？

若要表示的数绝对值还要更小，怎么办？

最小绝对值：尾数全为0，阶码真值最小-126，对应移码机器数 0000 0001
此时整体的真值为 $(1.0)_2 \times 2^{-126}$

只有 $1 \leq E \leq 254$时，真值 $= (-1)^s \times 1.M \times 2^{E-127}$

隐含最高位变为0

阶码真值固定视为 -126

当阶码E全为0，尾数M不全为0时，表示非规格化小数 $\pm (0.xx...x)_2 \times 2^{-126}$

当阶码E全为0，尾数M全为0时，表示真值 ±0

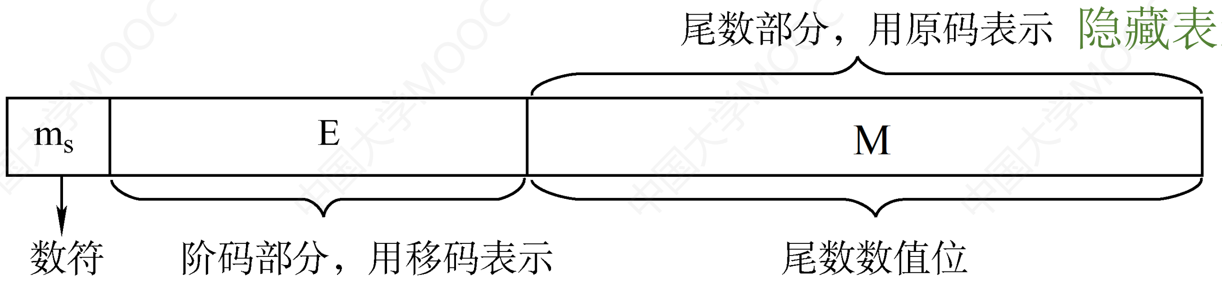
如：0/0、∞-∞等非法运算的结果就是 NaN

当阶码E全为1，尾数M全为0时，表示无穷大 ±∞

当阶码E全为1，尾数M不全为0时，表示非数值 “NaN” (Not a Number)

知识点回顾

阶码全1、全0
用作特殊用途



隐藏表示最高位1

表示尾数1.M

类　型	数　符	阶　码	尾 数 数 值	总 位 数	偏 置 值	
					十 六 进 制	十 进 制
短浮点数	1	8	23	32	7FH	127
长浮点数	1	11	52	64	3FFH	1023
临时浮点数	1	15	64	80	3FFFH	16383

由浮点数确定真值（阶码不是全0、也不是全1）：

1. 根据“某浮点数”确定数符、阶码、尾数的分布
2. 确定尾数 1.M　（注意补充最高的隐含位1）
3. 确定阶码的真值 = 移码 - 偏置值　（可将移码看作无符号数，用无符号数的值减去偏置值）
4. $(-1)^s \times 1.M \times 2^{E-\text{偏置值}}$