

1.1 信息存储

1.1.1 十六进制表示法

一、记忆：

- A = 10
- C = 12
- F = 15

二、2 的整数次幂与 16 进制相互转化

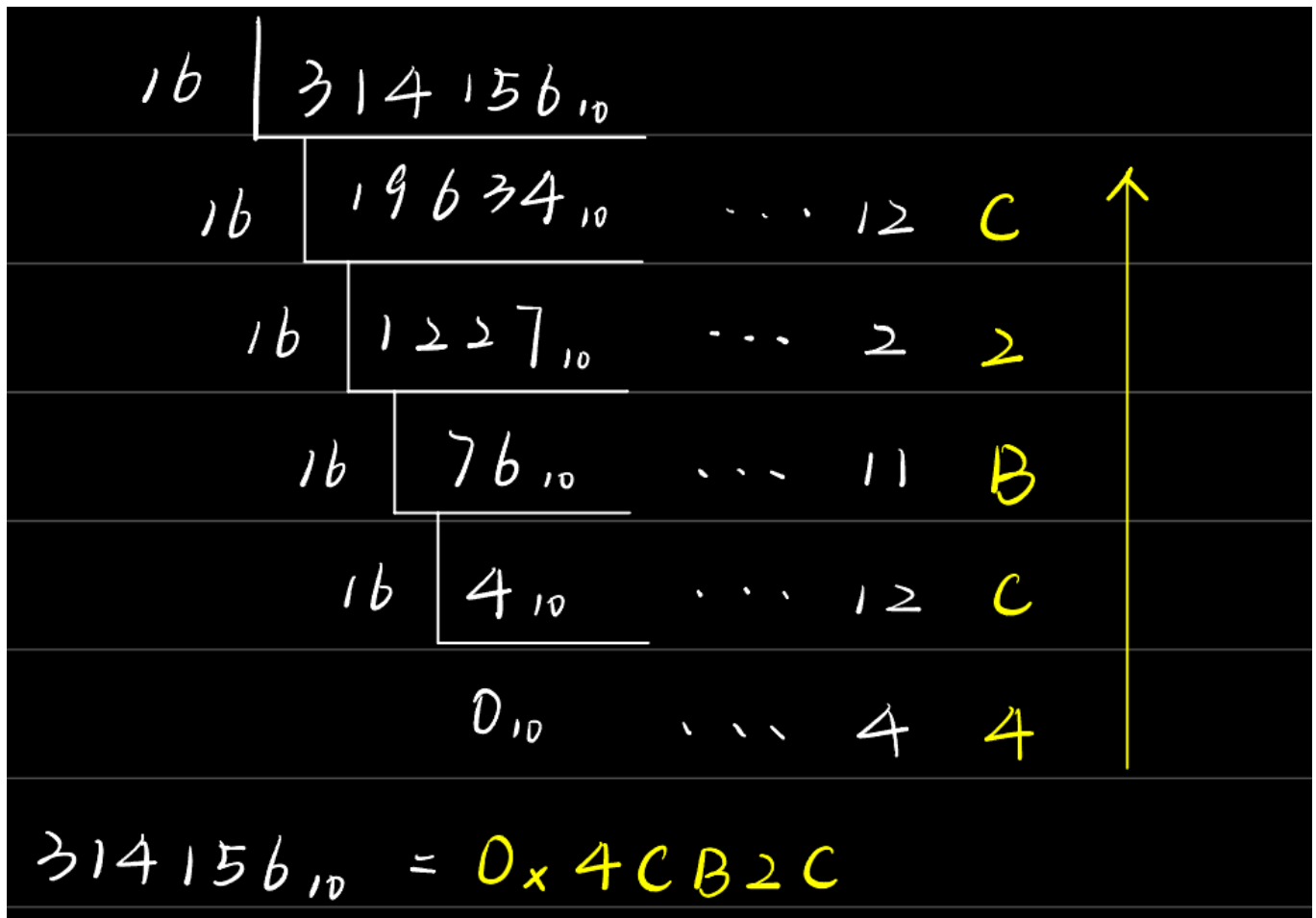
e.g.

$$n = 2_{10}^k, \quad k = i + 4j \quad (0 \leq i \leq 3)$$

$$n = 0x2_{Hex}^i 0 \dots 0, \text{ 共 } j \text{ 个 } 0$$

三、16 进制转化 10 进制

e.g.



1.1.2 字数据大小

C声明		字节数	
有符号	无符号	32 位	64 位
[signed] char	unsigned char	1	1
short	unsigned short	2	2
int	unsigned	4	4
long	unsigned long	4	8
int32_t	uint32_t	4	4
int64_t	uint64_t	8	8
pointer		4	8
float		4	4
double		8	8

在使用 char 的时候应当习惯性标注上是否为 signed
使用 int 的时候最佳方式是使用 int32_t 和 int64_t

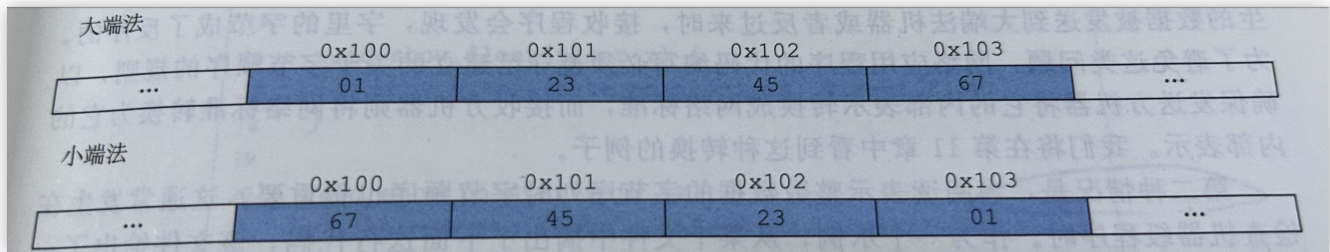
1.1.3 寻址和字节顺序

考虑一个 w 位的整数，其位表示为 $[x_{w-1}, x_{w-2}, \dots, x_2, x_1, x_0]$ ，其中 x_{w-1} 为最高有效位， x_0 为最低有效位，假设 w 是8的倍数，则这些位就能被分为字节，其中最高有效字节包含 $[x_{w-1}, x_{w-2}, \dots, x_{w-8}]$ ，最低有效字节包含 $[x_7, x_6, \dots, x_0]$ 。

- 大端法：最高有效字节存放在地址最前面的方式
- 小端法：最低有效字节存放在地址最前面的方式

e.g.

假设变量 `int x`，位于地址 `0x100` 处， x 的16进制值为 `0x01234567`，地址范围 `0x100~0x103` 的字节顺序依赖于机器的类型：



注意：高位字节为 `0x01`，低位字节为 `0x67`。