

数据的机器级表示与运算

数据在存储器中的存放格式

主讲人：邓倩妮

上海交通大学

部分内容来自：

1. 《深入理解计算机系统》第三版，机械工业出版社，作者：Bryant, R. E. 等
2. *Computer Organization and Design, 4th Edition*, Patterson & Hennessy



上海交通大学

SHANGHAI JIAO TONG UNIVERSITY



数据存储字节顺序

数据存储字节顺序(Endianness)

- 大数端(Big Endian)
 - 最低字节存储在高地址
- 小数端(Little Endian)
 - 最低字节存储在低地址
- 例:数据000F4240:

4	5	6	7
00	0F	42	40

(a) 大数端存储方式

4	5	6	7
40	42	0F	00

(b) 小数端存储方式



数据类型及其存储方式

- 数据对齐方式(Alignment)

地址	3	2	1	0
0				
4				
8				
12				
16				
...

(a) 字不对齐

字节	3	2	1	0
0				
4				
8				
12				
16				
...

(b) 字对齐



字不对齐方式下的数据存储

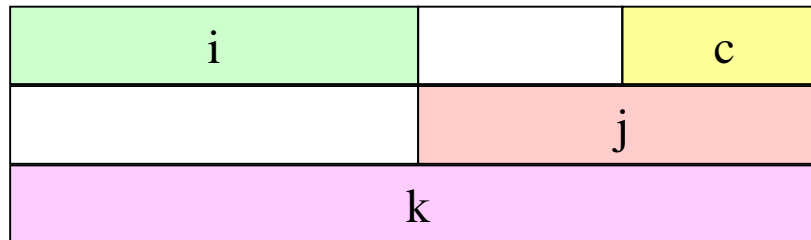
- 访存次数问题
 - char c;
 - short int i,j;
 - int k;
- 跨页问题

Short j	Short int i	Char c
int k		Short j
		int k



字对齐方式下的数据存储

- 空间浪费问题
 - char c;
 - short int i,j;
 - int k;

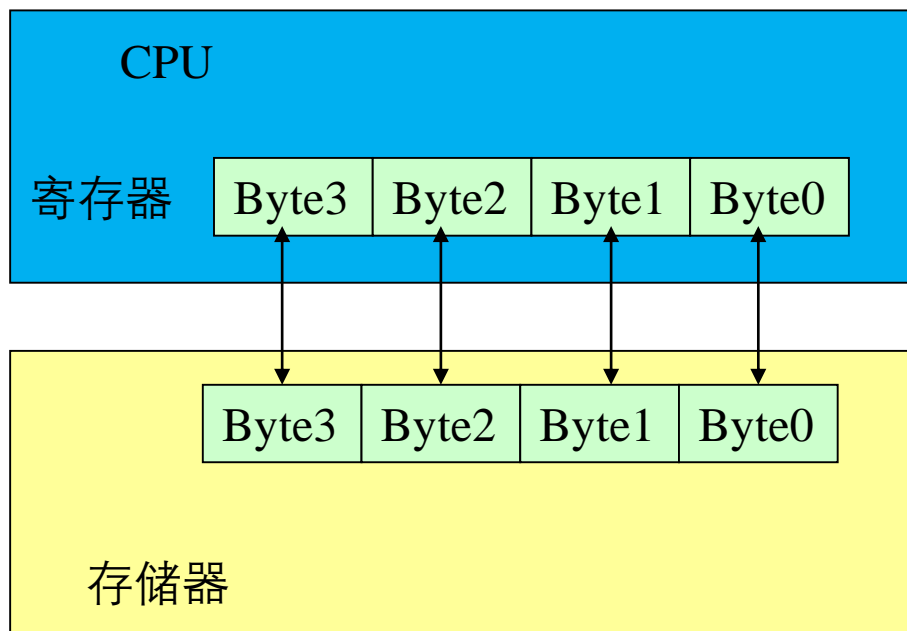


如何判断字对齐？



字对齐方式下的存储器访问

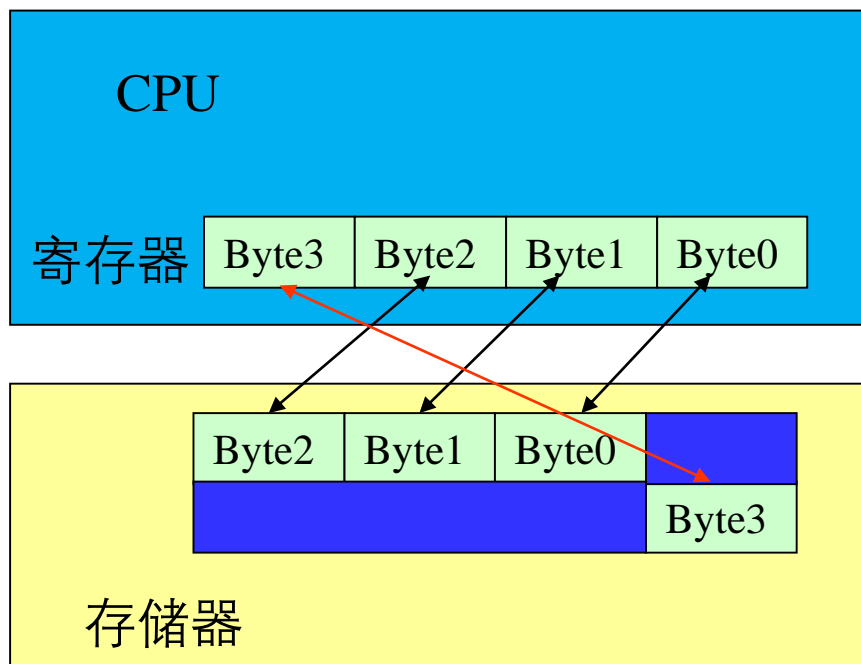
- 速度较高
- 接口较简单





字不对齐方式下的存储器访问

- 节省空间
- 访存速度慢
- 接口复杂





数据对齐的例子

- 试改变以下数据结构定义以减少对齐的开销

```
struct Loose {  
    short s;           // 16 bit  
    int i;             // 32bit  
    char c;            // 8 bit  
    Foo* p;            // 64 bit  
};
```

```
struct Tight {  
    Foo* p; // 64 bit  
    int i;           // 32bit  
    short s; // 16 bit  
    char c; // 8 bit  
};
```




分析题

- 某计算机字长为32位，按字节编址，采用小端（little Endian）方式存放数据，假定有一个double型变量，其机器数表示为1122 3344 5566 7788H，存放在0000 8040H开始连续存储单元中，其存储单元 0000 8046H 中存放的是（）
- A. 22H B. 33H C. 66H D. 77H

(A)



分析题

某计算机存储器按字节编址，采用小端方式放数据。假定编译器规定int 型和 short型 长度分别为 32 位和 16 位，并且数据按边界对齐存储。语言程序段如下：

```
Struct  
{ int a;  
  char b;  
  short c;  
} record;  
record.a=273;
```

若record变量的首地址是0xC008，
则地址0xC008 中的内容，以及
record.c的地址是（）

- A. 0x00, 0xC00D
- B. 0x00, 0xC00E
- C. 0x11, 0xC00D
- D. 0x11, 0xC00E

谢谢！

