

P119-123

1. 数据格式

| C 声明   | Intel 数据类型 | 汇编代码后缀 | 大小(字节) |
|--------|------------|--------|--------|
| char   | 字节         | b      | 1      |
| short  | 字          | w      | 2      |
| int    | 双字         | l      | 4      |
| long   | 四字         | q      | 8      |
| char*  | 四字         | q      | 8      |
| float  | 单精度        | s      | 4      |
| double | 双精度        | l      | 8      |

图 3-1 C 语言数据类型在 x86-64 中的大小。在 64 位机器中，指针长 8 字节

字表示 16 位数据类型。因此 32 位为双字，64 位为四字。比如 long 类型实现为 64 位，范围较大。

浮点数有两种：单精度值（4 字节），对应 c 语言是 float；双精度 8 字节，对应 double。

在汇编代码指令中，都有一个字符的后缀，表示操作数的大小。

比如，汇编代码用“l”来表示 4 字节整数和 8 字节双精度浮点数，这不会产生歧义，因为二者用的完全不同的指令和寄存器。

1. 访问信息：

- a. 一个 cpu 包含一组 16 个存储 64 位值的通用目的寄存器，这些寄存器用来存储数据和指针。

|      |       |       |       |   |        |
|------|-------|-------|-------|---|--------|
| 63   | 31    | 15    | 7     | 0 |        |
| %rax | %eax  | %ax   | %al   |   | 返回值    |
| %rbx | %ebx  | %bx   | %bl   |   | 被调用者保存 |
| %rcx | %ecx  | %cx   | %cl   |   | 第4个参数  |
| %rdx | %edx  | %dx   | %dl   |   | 第3个参数  |
| %rsi | %esi  | %si   | %sil  |   | 第2个参数  |
| %rdi | %edi  | %di   | %dil  |   | 第1个参数  |
| %rbp | %ebp  | %bp   | %bpl  |   | 被调用者保存 |
| %rsp | %esp  | %sp   | %spl  |   | 栈指针    |
| %r8  | %r8d  | %r8w  | %r8b  |   | 第5个参数  |
| %r9  | %r9d  | %r9w  | %r9b  |   | 第6个参数  |
| %r10 | %r10d | %r10w | %r10b |   | 调用者保存  |
| %r11 | %r11d | %r11w | %r11b |   | 调用者保存  |
| %r12 | %r12d | %r12w | %r12b |   | 被调用者保存 |
| %r13 | %r13d | %r13w | %r13b |   | 被调用者保存 |
| %r14 | %r14d | %r14w | %r14b |   | 被调用者保存 |
| %r15 | %r15d | %r15w | %r15b |   | 被调用者保存 |

图 3-2 整数寄存器。所有 16 个寄存器的低位部分都可以作为字节、字(16 位)、双字(32 位)和四字(64 位)数字来访问

指令可以对寄存器中低位字节存放不同大小的数据进行操作，根据操作位数不同，访问寄存器的位数也就不同。

不同寄存器扮演着不同的角色，比如栈指针%rsp，用来指明运行时栈的结束位置。有些程序会明确读写这个寄存器。其他的寄存器使用更加灵活，因此会有一组编程规范控制着如何使用寄存器来管理和操作数据。

#### ○ 操作指示符