作者：代兴　邮箱：503268771@qq.com　Vx 公众号：东方架构师

寄存器种类繁多，这里列出常用的寄存器。

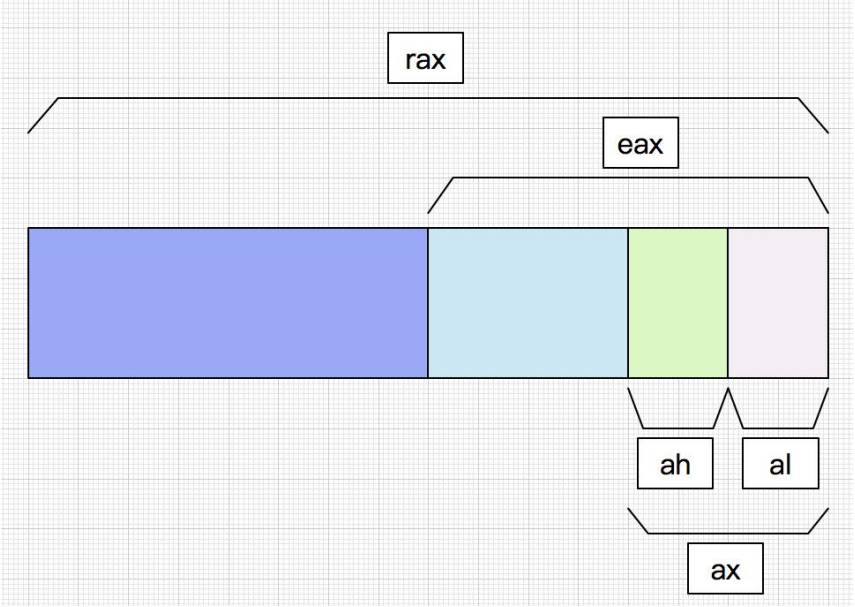
整数寄存器

整数寄存器的长度为 64 位、32 位、16 位、8 位。

64 位	32 位	16 位	次低 8 位	低 8 位
rax	eax	ax	ah	al
rbx	ebx	bx	bh	bl
rcx	ecx	cx	ch	cl
rdx	edx	dx	dh	dl
rdi	edi	di		dil
rsi	esi	si		sil
r8	r8d	r8w		r8b
r9	r9d	r9w		r9b
r10	r10d	r10w		r10b
r11	r11d	r11w		r11b
r12	r12d	r12w		r12b
r13	r13d	r13w		r13b
r14	r14d	r14w		r14b
r15	r15d	r15w		r15b

指令使用后缀，操作对应位数的寄存器。
64 位指令的后缀是 q，表示 quad，4 倍的字。比如，movq、addq。
32 位指令的后缀是 l，表示 long，长型的字。比如，movl、addl。
16 位指令的后缀是 w，表示 word，2 个字节。比如，movw、addw。
8 位指令的后缀是 b，表示 byte，1 个字节。比如，movb、addb。

以 mov 为例。
`movq $700, %rax` 把 700 赋给 rax。
`movl $700, %eax` 把 700 赋给 eax。
`movw $700, %ax` 把 700 赋给 ax。
`movb $7, %ah` 把 7 赋给 ah。



浮点数寄存器

128 位
xmm0
xmm1
xmm2
xmm3
xmm4
xmm5
xmm6
xmm7
xmm8
xmm9
xmm10
xmm11
xmm12
xmm13
xmm14
xmm15

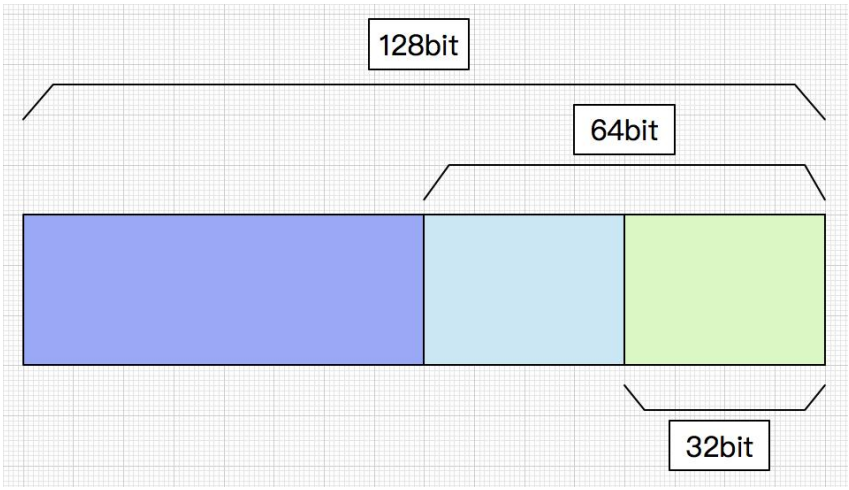
浮点数使用 xmm 寄存器。
xmm 寄存器有 128 位。常用 32 位 float，64 位 double。
32 位浮点数指令，使用 ss 后缀。比如，movss、addss、subss。
64 位浮点数指令，使用 sd 后缀。比如，movsd、addsd、subsd。

定义 2 个浮点数变量，然后使用浮点数寄存器。

```
float32:
    .float 11.22

float64:
    .double 33.55
```

```
movss float32(%rip), %xmm0
movsd float64(%rip), %xmm0
addss float32(%rip), %xmm15
subsd float64(%rip), %xmm15
```



栈寄存器

64 位	32 位
rbp	ebp
rsp	esp

函数功能，操作栈帧。
`pushq %rbp` 把 rbp 的值压入栈顶。
`movq %rsp, %rbp` 把 rsp 的值赋给 rbp。
`popq %rbp` 弹出栈顶，赋给 rbp。
`retq` 退出函数。

指令指针寄存器

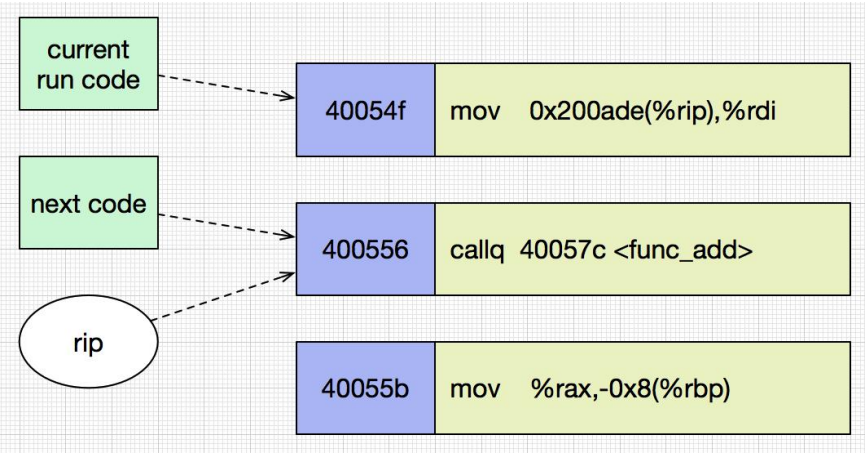
64 位	32 位
rip	eip

rip 存放下一个指令的地址。执行汇编指令，rip 的值跟着变化。
rip 相对寻址。在指定的内存位置，为某个符号写入 rip 的相对偏移。

调用函数的代码。

```
 40054f:   48 8b 3d de 0a 20 00    mov    0x200ade(%rip),%rdi        # 601034 <num_int64>
 400556:   e8 21 00 00 00          callq  40057c <func_add>
 40055b:   48 89 45 f8             mov    %rax,-0x8(%rbp)
```

假设 CPU 当前执行 40054f 对应的指令，则下一个指令的地址为 400556，rip 的值为 400556。



使用 rip 相对寻址，得出绝对地址。
40054f 行使用了 rip 相对寻址 0x200ade(%rip)。400556 行是下一个指令。
计算公式：绝对地址 = rip 的值 + 偏移 。

```
    uint64_t tmp = 0x400556 + 0x200ade;
    printf("  %#llX  \n", tmp);
```

状态寄存器

64 位	32 位
rflags	eflags

加、减、乘、除、比较等操作，更新状态寄存器。
跳转等操作，依赖状态寄存器的值。

```
subq $55, %rax  # rax = rax - 5
cmpq $77, %rcx  # compare rcx and 77
je cond_equal   # if equal jump to cond_equal
```