

指令说明

mulsd、mulss 指令表示浮点数乘法。

mulsd 指令为 64 位。mulss 指令为 32 位。

mulsd、mulss 指令可以操作寄存器、内存。

语法格式 `mulsd ee, ff` 表示 `ee * ff = ff`。

语法格式 `mulss ee, ff` 表示 `ee * ff = ff`。

简单的乘法

编写代码： `mul.s`

```
.data

float64 :
    .double 111.11

float32 :
    .float 333.22

tmp32 :
    .float 200

str_64 :
    .string " float64 = %.5f  \n"

.text
.global main

main :
    pushq %rbp
    movq %rsp, %rbp

    # 乘法。64 位
    movsd float64(%rip), %xmm0 # 64 位，浮点数，xmm0
    movq $-5, %r9               # 64 位，整数，r9
    cvtsi2sd %r9, %xmm7         # 64 位，整数转浮点数
    mulsd %xmm7, %xmm0          # xmm0 = xmm0 * xmm7
    movq $str_64, %rdi
    callq printf

    # 乘法。32 位
```

```
movss float32(%rip), %xmm0 # 32 位, 浮点数, xmm0
mulss tmp32(%rip), %xmm0    # xmm0 = xmm0 * tmp32
movq $str_64, %rdi
cvtss2sd %xmm0, %xmm0      # 32 位浮点数转 64 位浮点数
callq printf

popq %rbp
retq
```

编译代码:

```
gcc mul.s -o mul
```

运行代码:

```
[root@local float]# ./mul
float64 = -555.55000
float64 = 66644.00000
```

分析结果:

汇编代码	结果和分析
<div># 乘法。64 位</div> <div>movsd float64(%rip), %xmm0 # 64 位, 浮点数, xmm0</div> <div>movq \$-5, %r9 # 64 位, 整数, r9</div> <div>cvtsi2sd %r9, %xmm7 # 64 位, 整数转浮点数</div> <div>mulsd %xmm7, %xmm0 # xmm0 = xmm0 * xmm7</div>	<div>float64 = -555.55000</div> <div>把 64 位浮点数变量 float64 的值 111.11 写到 xmm0 。</div> <div>把 64 位整数-5 写到 r9, 再转为浮点数写到 xmm7 。</div> <div>浮点数乘法, 寄存器 xmm0 乘以寄存器 xmm7, 结果写到 xmm0。</div> <div>111.11 * (-5) = -555.55000</div>
<div># 乘法。32 位</div> <div>movss float32(%rip), %xmm0 # 32 位, 浮点数, xmm0</div> <div>mulss tmp32(%rip), %xmm0 # xmm0 = xmm0 * tmp32</div>	<div>float64 = 66644.00000</div> <div>把 32 位浮点数变量 float32 的值 333.22 写到 xmm0 。</div> <div>浮点数乘法, 寄存器 xmm0 乘以内存变量 tmp32, 结果写到 xmm0。</div> <div>333.22 * 200 = 66644.00000</div>