

指令说明

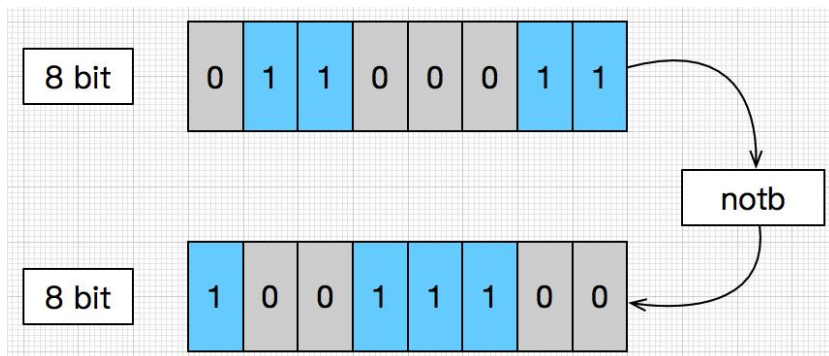
not 指令表示位取反。

not 指令分为 8 位 notb、16 位 notw、32 位 notl、64 位 notq。

not 指令可以操作寄存器、内存。

语法格式 `not aa` 表示 `aa = ~ aa`。

对应的位执行位取反操作。如果位为 1，则结果为 0；如果位为 0，则结果为 1。



用汇编代码分析

编写代码： `not_bit.s`

```
.data

str64 :
    .string " int64  %#llx \n"

str32 :
    .string " int32  %#X \n"

.text
.global main

main :
    pushq %rbp
    movq %rsp, %rbp
    subq $64, %rsp

    # 64 位。操作寄存器
    movq $0x0F0F0F0F0F0F0F0F, %rax # 64 位，寄存器
    notq %rax                      # 64 位，位非
    movq $str64, %rdi
    movq %rax, %rsi
```

```

callq printf

# 32 位。操作寄存器
movl $0xAAAA7777, %ecx # 32 位, 寄存器
notl %ecx               # 32 位, 位非
movq $str32, %rdi
movl %ecx, %esi
callq printf

# 32 位。操作栈内存
movl $0xAAAA7777, -8(%rbp) # 32 位, 栈
notl -8(%rbp)              # 32 位, 位非
movq $str32, %rdi
movl -8(%rbp), %esi
callq printf

# 16 位。操作寄存器
movl $0xAAAAAAAA, %ebx # 32 位, 占位
movw $0xCCCC, %bx      # 16 位, 寄存器
notw %bx                # 16 位, 位非
movq $str32, %rdi
movl %ebx, %esi
callq printf

# 8 位。操作寄存器
movl $0xAAAAAAAA, %ebx # 32 位, 占位
movb $0xCC, %bl        # 8 位, 寄存器
notb %bl                # 8 位, 位非
movq $str32, %rdi
movl %ebx, %esi
callq printf

addq $64, %rsp
popq %rbp
retq

```

编译代码:

```
gcc not_bit.s -o not_bit
```

运行代码:

```

[root@local binary]# ./not_bit
int64  0XF0F0F0F0F0F0F0
int32  0X55558888
int32  0X55558888
int32  0XAAAA3333
int32  0XAAAAAA33

```

分析结果:

汇编代码	结果和分析
------	-------

# 64 位。操作寄存器 movq \$0x0F0F0F0F0F0F0F, %rax # 64 位, 寄存器 notq %rax # 64 位, 位非	int64 0XF0F0F0F0F0F0F0F0 操作 64 位寄存器 rax。 ~ 0x0F0F0F0F0F0F0F0F = 0XF0F0F0F0F0F0F0F0
# 32 位。操作寄存器 movl \$0xAAAA7777, %ecx # 32 位, 寄存器 notl %ecx # 32 位, 位非	int32 0X55558888 操作 32 位寄存器 ecx。 ~ 0xAAAA7777 = 0X55558888
# 32 位。操作栈内存 movl \$0xAAAA7777, -8(%rbp) # 32 位, 栈 notl -8(%rbp) # 32 位, 位非	int32 0X55558888 操作栈内存。 ~ 0xAAAA7777 = 0X55558888
# 16 位。操作寄存器 movl \$0xAAAAAAAA, %ebx # 32 位, 占位 movw \$0xCCCC, %bx # 16 位, 寄存器 notw %bx # 16 位, 位非	int32 0XAAAA3333 操作 16 位寄存器 bx。 高 16 位, 没有变化, 都为 AAAA。 低 16 位, 发生变化, 从 CCCC 变成 3333。 ~ CCCC = 3333
# 8 位。操作寄存器 movl \$0xAAAAAAAA, %ebx # 32 位, 占位 movb \$0xCC, %bl # 8 位, 寄存器 notb %bl # 8 位, 位非	int32 0XAAAAAA33 操作 8 位寄存器 bl。 高 24 位, 没有变化, 都为 AAAAAA。 低 8 位, 发生变化, 从 CC 变成 33。 ~ CC = 33

十六进制 F 等于二进制 1111。

十六进制取反某数，等价于 F 减去该数。

十六进制取反 A，等价于 $F - A = 5$ 。

十六进制取反 7，等价于 $F - 7 = 8$ 。

十六进制取反 0xAAAA7777，等于 0X55558888。

十六进制取反 0x0F0F0F0F0F0F0F0F，等于 0XF0F0F0F0F0F0F0F0。