

一、数据传输指令

它们在存储器、寄存器和寄存器、寄存器和输入输出端口之间传送数据。

1. 通用数据传送指令。

MOV 传送字或字节。

MOVSX 先符号扩展,再传送。

MOVZX 先零扩展,再传送。

PUSH 把字压入堆栈。

POP 把字弹出堆栈。

PUSHA 把 AX, CX, DX, BX, SP, BP, SI, DI 依次压入堆栈。

POPA 把 DI, SI, BP, SP, BX, DX, CX, AX 依次弹出堆栈。

PUSHAD 把 EAX, ECX, EDX, EBX, ESP, EBP, ESI, EDI 依次压入堆栈。

POPAD 把 EDI, ESI, EBP, ESP, EBX, EDX, ECX, EAX 依次弹出堆栈。

BSWAP 交换 32 位寄存器里字节的顺序

XCHG 交换字或字节。(至少有一个操作数为寄存器,段寄存器不可作为操作数)

CMPXCHG 比较并交换操作数。(第二个操作数必须为累加器 AL/AX/EAX)

XADD 先交换再累加。(结果在第一个操作数里)

XLAT 字节查表转换。

—— BX 指向一张 256 字节的表的起点, AL 为

表的索引值 (0-255, 即

0-FFH); 返回 AL 为查表结果. ([BX+AL]->AL)

2. 输入输出端口传送指令.

IN I/O 端口输入. (语法: IN 累加器, {端口号 | DX})

OUT I/O 端口输出. (语法: OUT {端口号 | DX}, 累加器)

输入输出端口由立即方式指定时, 其范围是 0-255; 由寄存器 DX 指定时,

其范围是 0-65535.

3. 目的地址传送指令.

LEA 装入有效地址.

例: LEA DX, string ; 把偏移地址存到 DX.

LDS 传送目标指针, 把指针内容装入 DS.

例: LDS SI, string ; 把段地址: 偏移地址存到 DS: SI.

LES 传送目标指针, 把指针内容装入 ES.

例: LES DI, string ; 把段地址: 偏移地址存到 ES: DI.

LFS 传送目标指针, 把指针内容装入 FS.

例: LFS DI, string ; 把段地址: 偏移地址存到 FS: DI.

LGS 传送目标指针, 把指针内容装入 GS.

例: LGS DI, string ; 把段地址: 偏移地址存到 GS: DI.

LSS 传送目标指针, 把指针内容装入 SS.

例: LSS DI, string ; 把段地址: 偏移地址存到 SS: DI.

4. 标志传送指令.

LAHF 标志寄存器传送,把标志装入 AH.
SAHF 标志寄存器传送,把 AH 内容装入标志寄存器.
PUSHF 标志入栈.
POPF 标志出栈.
PUSHD 32 位标志入栈.
POPD 32 位标志出栈.

二、算术运算指令

ADD 加法.
ADC 带进位加法.
INC 加 1.
AAA 加法的 ASCII 码调整.
DAA 加法的十进制调整.
SUB 减法.
SBB 带借位减法.
DEC 减 1.
NEG 求反(以 0 减之).
CMP 比较.(两操作数作减法,仅修改标志位,不回送结

果).

AAS 减法的 ASCII 码调整.
DAS 减法的十进制调整.
MUL 无符号乘法.
IMUL 整数乘法.

以上两条, 结果回送 AH 和 AL (字节运算), 或 DX 和 AX (字运算),

AAM 乘法的 ASCII 码调整.

DIV 无符号除法.

IDIV 整数除法.

以上两条, 结果回送:

商回送 AL, 余数回送 AH, (字节运算);

或 商回送 AX, 余数回送 DX, (字运算).

AAD 除法的 ASCII 码调整.

CBW 字节转换为字. (把 AL 中字节的符号扩展到 AH 中去)

CWD 字转换为双字. (把 AX 中的字的符号扩展到 DX 中去)

CWDE 字转换为双字. (把 AX 中的字符符号扩展到 EAX 中去)

CDQ 双字扩展. (把 EAX 中的字的符号扩展到 EDX 中去)

三、逻辑运算指令

AND 与运算.

OR 或运算.

XOR 异或运算.

NOT 取反.

TEST 测试. (两操作数作与运算, 仅修改标志位, 不回送结果).

SHL 逻辑左移.

SAL 算术左移. (=SHL)
SHR 逻辑右移.
SAR 算术右移. (=SHR)
ROL 循环左移.
ROR 循环右移.
RCL 通过进位的循环左移.
RCR 通过进位的循环右移.

以上八种移位指令,其移位次数可达 255 次.

移位一次时, 可直接用操作码. 如 SHL AX, 1.

移位>1 次时, 则由寄存器 CL 给出移位次数.

如 MOV CL, 04

 SHL AX, CL

四、串指令

DS: SI 源串段寄存器 :源串变址.

ES: DI 目标串段寄存器:目标串变址.

CX 重复次数计数器.

AL/AX 扫描值.

D 标志 0 表示重复操作中 SI 和 DI 应自动增量; 1 表示应自动减量.

Z 标志 用来控制扫描或比较操作的结束.

MOVS 串传送.

(MOVSB 传送字符. MOVSW 传送字. MOVSD

传送双字.)

CMPS 串比较.

(CMPSB 比较字符. CMPSW 比较字.)

SCAS 串扫描.

把 AL 或 AX 的内容与目标串作比较, 比较结果反映在标志位.

LODS 装入串.

把源串中的元素(字或字节)逐一装入 AL 或 AX 中.

(LODSB 传送字符. LODSW 传送字. LODSD

传送双字.)

STOS 保存串.

是 LODS 的逆过程.

REP 当 CX/ECX<>0 时重复.

REPE/REPZ 当 ZF=1 或比较结果相等, 且 CX/ECX<>0 时重复.

REPNE/REPZ 当 ZF=0 或比较结果不相等, 且 CX/ECX<>0 时重复.

REPC 当 CF=1 且 CX/ECX<>0 时重复.

REPNC 当 CF=0 且 CX/ECX<>0 时重复.

五、程序转移指令

1>无条件转移指令 (长转移)

JMP 无条件转移指令

CALL 过程调用

RET/RETF 过程返回.

2>条件转移指令 (短转移, -128 到+127 的距离内)

(当且仅当 $(SF \oplus OF) = 1$ 时, OP1

JA/JNBE 不小于或不等于时转移.

JAE/JNB 大于或等于转移.

JB/JNAE 小于转移.

JBE/JNA 小于或等于转移.

以上四条, 测试无符号整数运算的结果 (标志 C 和 Z).

JG/JNLE 大于转移.

JGE/JNL 大于或等于转移.

JL/JNGE 小于转移.

JLE/JNG 小于或等于转移.

以上四条, 测试带符号整数运算的结果 (标志 S, O 和 Z).

JE/JZ 等于转移.

JNE/JNZ 不等于时转移.

JC 有进位时转移.

JNC 无进位时转移.

JNO 不溢出时转移.

JNP/JPO 奇偶性为奇数时转移.

JNS 符号位为 "0" 时转移.

JO 溢出转移.

JP/JPE 奇偶性为偶数时转移.

JS 符号位为 "1" 时转移.

3>循环控制指令(短转移)

LOOP CX 不为零时循环.

LOOPE/LOOPZ CX 不为零且标志 Z=1 时循环.

LOOPNE/LOOPNZ CX 不为零且标志 Z=0 时循环.

JCXZ CX 为零时转移.

JECXZ ECX 为零时转移.

4>中断指令

INT 中断指令

INT0 溢出中断

IRET 中断返回

5>处理器控制指令

HLT 处理器暂停, 直到出现中断或复位信号才继续.

WAIT 当芯片引线 TEST 为高电平时使 CPU 进入等待状态.

ESC 转换到外处理器.

LOCK 封锁总线.

NOP 空操作.

STC 置进位标志位.

CLC 清进位标志位.

CMC 进位标志取反.

STD 置方向标志位.

CLD 清方向标志位.

STI 置中断允许位.

CLI 清中断允许位.

六、伪指令

DW 定义字 (2 字节).

PROC 定义过程.

ENDP 过程结束.

SEGMENT 定义段.

ASSUME 建立段寄存器寻址.

ENDS 段结束.

END 程序结束.

七、端口指令

端口是主机与外设进行数据交换的。(外设接口电路有专用于数据交互的寄存器。为了与 CPU 中的寄存器相区别，称之为“端口”)。端口有数据端口，状态端口和控制端口 3 种。

PC 机给予每一个端口分配了一个地址 (称为端口号)，形成一个独立于内存空间的 I/O 地址空间。在 8086/8088 中，端口地址的范围是 0000 至 FFFF。

CPU 对外设的操作通过专门的端口读写指令来完成。读端口用 IN 指令，写端口用 OUT 指令。

IN ACC, PORT; 从端口 PORT 读取数据到 ACC;

PORT 可以是立即数，也可以是 DX。

ACC 为 AL 时，从 PORT 读取 1 字节数据到 AL

ACC 为 AX 时，从 PORT 读取 2 字节数据到 AX

OUT PORT, ACC; 将 ACC 的数据写入端口 PORT;

例子:

IN AL, 21H; 表示从 21H 端口读取一字节数据到 AL

IN AX, 21H; 表示从端口地址 21H 读取 1 字节数据到 AL，从端口地址
22H 读取 1 字节到 AH

MOV DX, 379H

IN AL, DX ; 从端口 379H 读取 1 字节到 AL

OUT 21H, AL; 将 AL 的值写入 21H 端口

OUT 21H, AX; 将 AX 的值写入端口地址 21H 开始的连续两个字节。

(port [21H]=AL, port [22h]=AH)

MOV DX, 378H

OUT DX, AX ; 将 AH 和 AL 分别写入端口 379H 和 378H