

计算机组成原理 Homework11 (11.16)

中国人民大学 信息学院 崔冠宇 2018202147

1. 编写程序，将偶数地址单元 200H, 202H, ..., 20EH 中 8 个数据，传送到奇数地址单元 301H, 303H, ..., 30FH 中。

解: 本题可以用循环结构。需要注意每次循环后源地址和目标地址增加 2。8086 汇编 (DOS debug) 代码如下所示:

```
1 MOV CX, 8          ; 循环8次
2 MOV SI, 200        ; 源地址
3 MOV DI, 301        ; 目标地址
4 L:                 ; 不输入标签, 仅记住指令地址
5 MOV AL, [SI]
6 MOV [DI], AL
7 INC SI             ; 增加2
8 INC SI
9 INC DI             ; 同上
10 INC DI
11 DEC CX            ; 循环次数减1
12 JNZ L             ; 跳转 改为上面的指令地址, 下同
13 INT 3
```

在 DEBUG 环境中, 为了观察指令效果, 首先用 -e 命令将地址为 200H~20FH 的单元的数据修改为非 0 值, 然后用 -a 键入代码并用 -g 命令执行, 最后用 -d 观察执行效果。部分截图如下所示:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
073F:0360 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0370 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
-e 200
073F:0200 00.01 00.02 00.03 00.04 00.05 00.06 00.07 00.08
073F:0208 00.09 00.0A 00.0B 00.0C 00.0D 00.0E 00.0F 00.10

-d 200
073F:0200 01 02 03 04 05 06 07 08-09 0A 0B 0C 0D 0E 0F 10 .....
073F:0210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
-d 300
073F:0300 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0310 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0320 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0330 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0340 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0350 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0360 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0370 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

图 1: 设置内存单元的值

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
073F:0109 MOV AL, [SI]
073F:010B MOV [DI], AL
073F:010D INC SI
073F:010E INC SI
073F:010F INC DI
073F:0110 INC DI
073F:0111 DEC CX
073F:0112 JNZ 109
073F:0114 INT 3
073F:0115
-g=100
AX=000F BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0210 DI=0311
DS=073F ES=073F SS=073F CS=073F IP=0114 NU UP EI PL ZR NA PE NC
073F:0114 CC INT 3
-d 300
073F:0300 00 01 00 03 00 05 00 07-00 09 00 0B 00 0D 00 0F .....
073F:0310 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0320 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0330 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0340 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0350 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0360 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0370 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

图 2: 键入代码并执行后内存单元的值

根据程序运行结果可见代码功能正确。

2. 编写程序，将从 200H 开始的 8 个单元存 00H, 01H, ..., 07H 的数据，移动到 202H 开始的 8 个单元中。

解：本题可以用循环结构。因为数据存放的新地址和旧地址有重叠，所以要从高地址向低地址逐个传送。8086 汇编 (DOS debug) 代码如下所示：

```
1 MOV CX, 8      ; 循环8次
2 MOV SI, 207    ; 源地址
3 MOV DI, 209    ; 目标地址
4 L:
5 MOV AL, [SI]
6 MOV [DI], AL
7 DEC SI
8 DEC DI
9 DEC CX
10 JNZ L
11 INT 3
```

在 DEBUG 环境中，首先用 -e 命令将地址为 200H~207H 的单元的数据修改为 00H~07H，然后用 -a 键入代码并用 -g 命令执行，最后用 -d 观察执行效果。部分截图如下所示：

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-e 200
073F:0200 00.00 01.01 02.02 03.03 04.04 05.05 06.06 07.07

-d 200
073F:0200 00 01 02 03 04 05 06 07-09 0A 0B 0C 0D 0E 0F 10 .....
073F:0210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....

-a 100
073F:0100 MOV CX, 8
073F:0103 MOV SI, 207
073F:0106 MOV DI, 209
073F:0109 MOV AL, [SI]
073F:010B MOV [DI], AL
073F:010D DEC SI
073F:010E DEC DI
073F:010F DEC CX
073F:0110 JNZ 109
073F:0112 INT 3
073F:0113
```

图 3: 设置内存单元的值

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
073F:0103 MOV SI, 207
073F:0106 MOV DI, 209
073F:0109 MOV AL, [SI]
073F:010B MOV [DI], AL
073F:010D DEC SI
073F:010E DEC DI
073F:010F DEC CX
073F:0110 JNZ 109
073F:0112 INT 3
073F:0113

-g=100
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=01FF DI=0201
DS=073F ES=073F SS=073F CS=073F IP=0112 NU UP EI PL ZR NA PE NC
073F:0112 CC INT 3

-d 200
073F:0200 00 01 00 01 02 03 04 05-06 07 0B 0C 0D 0E 0F 10 .....
073F:0210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

图 4: 键入代码并执行后内存单元的值

根据程序运行结果可见代码功能正确。

3. 编写程序，在 200H 开始的 8 个数据单元存放 00H, 01H, ..., 07H 8 个数据，利用交换指令 XCHG，要求程序运行之后，颠倒 8 个数据的存放顺序为: 07H, 06H, ..., 00H。

解: 本题可以用循环结构。想要颠倒 8 个数据，只需要进行 4 次交换。8086 汇编 (DOS debug) 代码如下所示:

```
1 MOV CX, 4           ; 循环4次
2 MOV SI, 200
3 MOV DI, 207
4 L:
5 XCHG AL, [SI]       ; 3次交换
6 XCHG [DI], AL
7 XCHG AL, [SI]
8 INC SI
9 DEC DI
10 DEC CX
11 JNZ L
12 INT 3
```

在 DEBUG 环境中，首先用 -e 命令将地址为 200H~207H 的单元的数据修改为 00H~07H，然后用 -a 键入代码并用 -g 命令执行，最后用 -d 观察执行效果。部分截图如下所示：

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
073F:0200 00.00 01.01 00.02 01.03 02.04 03.05 04.06 05.07

-a 100
073F:0100 MOV CX, 4
073F:0103 MOV SI, 200
073F:0106 MOV DI, 207
073F:0109 XCHG AL, [SI]
073F:010B XCHG [DI], AL
073F:010D XCHG AL, [SI]
073F:010F INC SI
073F:0110 DEC DI
073F:0111 DEC CX
073F:0112 JNZ 109
073F:0114 INT 3
073F:0115
-d 200
073F:0200 00 01 02 03 04 05 06 07-06 07 0B 0C 0D 0E 0F 10 .....
073F:0210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

图 5: 设置内存单元的值

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
073F:0115
-d 200
073F:0200 00 01 02 03 04 05 06 07-06 07 0B 0C 0D 0E 0F 10 .....
073F:0210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....

-g=100
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0204 DI=0203
DS=073F ES=073F SS=073F CS=073F IP=0114 NU UP EI PL ZR NA PE NC
073F:0114 CC INT 3
-d 200
073F:0200 07 06 05 04 03 02 01 00-06 07 0B 0C 0D 0E 0F 10 .....
073F:0210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

图 6: 键入代码并执行后内存单元的值

根据程序运行结果可见代码功能正确。

4. 编写程序，利用换码指令 XLAT 完成表格的转换。在地址 200H 处输入一个表格，含 10 个表项为 0~9 的 ASCII 码: 30H~39H。在 210H~217H 有 8 个单元，每个单元存放一个 0~9 的数字 (数字大小自己设定)。要求，程序运行后，210H~217H 中转换为对应数字的 ASCII 码。

解: 本题可以用循环结构。8086 汇编 (DOS debug) 代码如下所示:

```
1 MOV CX, 8                ; 8次循环
2 MOV BX, 200              ; ASCII表格首地址
3 MOV DI, 0                ; 距离数字表头 210 的偏移量
4 L:
5 MOV AL, [210 + DI]        ; 把内存单元中的数字取出
6 XLAT                     ; 将数字作为ASCII表格偏移量查表
7 MOV [210 + DI], AL        ; 送回
8 INC DI                   ; 下一个数字
9 DEC CX
10 JNZ L
11 INT 3
```

在 DEBUG 环境中,首先用 -e 命令将地址为 200H~209H 的单元的数据修改为 30H~39H,再用 -e 命令将 210H~217H 设置为某些数字 (此处是 2、4、6、8、7、5、3、1),然后用 -a 键入代码并用 -g 命令执行,最后用 -d 观察执行效果。部分截图如下所示:

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

C:\>debug
-e 200
073F:0200 00.30 00.31 00.32 00.33 00.34 00.35 00.36 00.37
073F:0208 00.38 00.39

-e 210
073F:0210 00.02 00.04 00.06 00.08 00.07 00.05 00.03 00.01

-d 200
073F:0200 30 31 32 33 34 35 36 37-38 39 00 00 00 00 00 00 0123456789.....
073F:0210 02 04 06 08 07 05 03 01-00 00 00 00 00 00 00 00.....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
```

图 7: 设置内存单元的值

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG

073F:0103 MOV BX, 200
073F:0106 MOV DI, 0
073F:0109 MOV AL, [210+DI]
073F:010D XLAT
073F:010E MOV [210+DI], AL
073F:0112 INC DI
073F:0113 DEC CX
073F:0114 JNZ 109
073F:0116 INT 3
073F:0117
-g=100

AX=0031 BX=0200 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0008
DS=073F ES=073F SS=073F CS=073F IP=0116 NU UP EI PL ZR NA PE NC
073F:0116 CC INT 3
-d 200
073F:0200 30 31 32 33 34 35 36 37-38 39 00 00 00 00 00 00 0123456789.....
073F:0210 32 34 36 38 37 35 33 31-00 00 00 00 00 00 00 00.....
073F:0220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
073F:0270 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00.....
```

图 8: 键入代码并执行后内存单元的值

根据程序运行结果可见代码功能正确。