

第二章

1. 将下列十进制数转换成二进制和十六进制

$$\begin{aligned}(1) \quad 129.75 &= 100000001.11\text{B} \\ &= 81.\text{CH}\end{aligned}$$

$$\begin{aligned}(2) \quad 218.8125 &= 1101\ 1010.1101\text{B} \\ &= \text{DA.DH}\end{aligned}$$

$$\begin{aligned}(3) \quad 15.625 &= 1111.101\text{B} \\ &= \text{F.AH}\end{aligned}$$

$$\begin{aligned}(4) \quad 47.15625 &= 101111.00101\text{B} \\ &= 2\text{F.28H}\end{aligned}$$

2. 将下列二进制数转换成十进制和十六进制

$$\begin{aligned}(1) \quad 111010\text{B} &= 58 \\ &= 3\text{AH}\end{aligned}$$

$$\begin{aligned}(2) \quad 10111100.11\text{B} &= 188.875 \\ &= \text{BC.E H}\end{aligned}$$

$$\begin{aligned}(3) \quad 0.11011\text{B} &= 0.84375 \\ &= 0.\text{D8 H}\end{aligned}$$

$$\begin{aligned}(4) \quad 11110.01\text{B} &= 30.25 \\ &= 1\text{E.4H}\end{aligned}$$

4、完成下列 16 进制的加减法运算。

$$(1) 0\text{D}14\text{B H} \qquad (2) 9\text{C}28.\text{E H}$$

$$(3) 1678.\text{FC H} \qquad (4) -80\text{D7 H}$$

5. 计算下列表达式的值

$$\begin{aligned}(1) \quad 128.8125 + 10110101.1011\text{B} + 1\text{F.2H} \\ = (101010101.101) \text{ B}\end{aligned}$$

$$\begin{aligned}(2) \quad 287.68 - 10101010.11\text{B} + 8\text{E.EH} \\ = (103.\text{CE}) \text{ H}\end{aligned}$$

$$\begin{aligned}(3) \quad 18.9 + 1010.1101\text{B} + 12.6\text{H} - 1011.1001\text{B} \\ = (36.525) \text{ D}\end{aligned}$$

7. 写出下列以补码表示的二进制数的真值：

1) $[X]_{\text{补}} = 1000\ 0000\ 0000\ 0000$
 $?X = -1000\ 0000\ 0000\ 0000 = -215 = -32768$

(2) $[Y]_{\text{补}} = 0000\ 0001\ 0000\ 0001$
 $?Y = +0000\ 0001\ 0000\ 0001 = +257$

(3) $[Z]_{\text{补}} = 1111\ 1110\ 1010\ 0101$
 $?X = -000\ 0001\ 0101\ 1011 = -347$

(4) $[A]_{\text{补}} = 0000\ 0010\ 0101\ 0111$
 $?X = +0000\ 0010\ 0101\ 0111 = +599$

9、设有变量……..

$X+Y = 1B8\ H$ $Y+Z = 161\ H$

$Y+Z = 13B\ H$ $Z+V = 0CC\ H$

(1) 不正确 不正确 不正确 正确

(2) 正确 正确 正确 溢出

12. 试计算下列二进制数为无符号数、原码、反码、补码、8421BCD 码时分别代表的数值大小。若为非 8421BCD 数时请指出。

(1) 10001000B

无符号数: $27+23 = 136$

原码: $-000\ 1000 = -8$

反码: $-111\ 0111 = -119$

补码: $-111\ 1000 = -120$

8421BCD 码: 88

2) 00101001B

无符号数: $25+23+20 = 41$

原码: 41

反码: 41

补码: 41

8421BCD 码: 29

3) 11001001B

无符号数: $27+26+23+20 = 201$

原码: $-100\ 1001 = -73$

反码: $-011\ 0110 = -54$

补码: $-011\ 0111 = -55$

8421BCD 码: 非 8421BCD 码

(4) 10010011B

无符号数: $27+24+21+20 = 147$

原码: $-001\ 0011 = -19$

反码: - 110 1100 = - 108

补码: - 110 1101 = - 109

8421BCD 码: 93

第三章 80X86 微处理器

1. 简述 8086 / 8088CPU 中 BIU 和 EU 的作用, 并说明其并行工作过程。

答: (1) BIU 的作用: 计算 20 位的物理地址, 并负责完成 CPU 与存储器或 I/O 端口之间的数据传送。

(2) EU 的作用: 执行指令, 并为 BIU 提供所需的有效地址。

(3) 并行工作过程: 当 EU 从指令队列中取出指令执行时, BIU 将从内存中取出指令补充到指令队列中。这样就实现了取指和执行指令的并行工作。

2. 8086 / 8088CPU 内部有哪些寄存器?其主要作用是什么?

答: 8086 / 8088CPU 内部共有 14 个寄存器, 可分为 4 类: 数据寄存器 4 个, 地址寄存器 4 个, 段寄存器 4 个和控制寄存器 2 个。其主要作用是:

(1) 数据寄存器: 一般用来存放数据, 但它们各自都有自己的特定用途。
AX, BX, CX, DX

(2) 地址寄存器: 一般用来存放段内的偏移地址。SP, BP, SI, DI

(3) 段寄存器: 用于存放段地址。CS, DS, ES, SS

(4) 控制寄存器 , FLAGS, IP

具体寄存器略。

3. 8086 / 8088CPU 中有哪些寄存器可用来指示操作数在存储器中某段内的偏移地址?

答: 可用来指示段内偏移地址的寄存器共有 6 个: IP、SP、BP、BX、SI、DI。

4. 8086 / 8088CPU 中标志寄存器 FLAGS 有哪些标志位?它们的含义和作用如何?

答: 标志寄存器中的标志共有 9 个, 分为两类: 状态标志 6 个和控制标志 3 个。其作用是:

(1) 状态标志：用来反映算术和逻辑运算结果的一些特征。CF,AF,PF,ZF,SF,ZF

(2) 控制标志位：用来控制 CPU 的操作，由程序设置或清除。DF,IF,TF.

具体寄存器位介绍略。

5. 8086 / 8088CPU 的地址总线有多少位?其寻址范围是多少?

答：8086/8088 地址总线有 20 根，可寻址 1MB 空间，范围：00000H~0FFFFFFH.

6. 什么叫指令队列?8086 / 8088CPU 中指令队列有什么作用?其长度分别是多少?

答：(1) 指令队列：采用“先进先出”原则，按顺序存放预执行指令的缓冲器称为指令队列。

(2) 指令队列的作用：存放 EU 将要执行的指令，使 CPU 的取指和执行指令能并行工作。

(3) 指令队列的长度：8086 为 6 个字节，8088 为 4 个字节。

7. Intel8086 与 8088 有何区别?

答：8086 与 8088 的区别有三处：

(1) 外部数据总线位数不同（即地址 / 数据分时复用引脚条数不同）；
8086 为 16 位：AD15~AD0 。

8088 为 8 位：AD7~AD0 。

(2) 内部指令队列缓冲器长度不同；

8086 有 6 个字节。当指令队列出现 2 个空字节时，BIU 将取指补充。

8088 有 4 个字节。当指令队列出现 1 个空字节时，BIU 将取指补充。

(3) 外部某些控制总线定义不同。

① 8086 的 28 号引脚定义为 M / IO(S2)，8088 定义为 IO / M(S2)

② 8086 的 34 号引脚定义为 BHE / S7，8088 定义为 SS0 / (HIGH)

第 4 章作业

1、指出源操作数的寻址方式

- (1) MOV BX, 2000H ; 立即数寻址
- (2) MOV BX, [2000H]; 直接寻址
- (3) MOV BX, [SI] ; 寄存器间接寻址
- (4) MOV BX, [SI+2000H] ; 寄存器相对寻址
- (5) MOV [BX+SI], AL ; 寄存器寻址
- (6) ADD AX, [BX+DI+80] ; 基址变址相对寻址
- (7) MUL BL ; 寄存器寻址
- (8) JMP BX ; 段内间接寻址
- (8) SUB AX, BX ; 寄存器寻址
- (9) IN AL, DX ; 端口间接寻址
- (10) INC WORD PTR [BP+10H] ; 寄存器相对寻址
- (11) MOV CL,LENGTH VAR ; 立即数寻址
- (12) MOV BL, OFFSET VAR1 ; 立即数寻址

2、指出下列指令是否正确

- (1) MOV DS, 0100H; 错误。源操作数是立即数时，目的操作数不能是段寄存器
- (2) MOV BP, AL; 错误。操作数类型不一致
- (3) XCHG AH, AL ; 正确。
- (4) OUT 310H, AL; 错误。端口直接寻址的范围应在 0~FFH 之间
- (5) MOV BX, [BX] ; 正确。
- (6) MOV ES:[BX+DI] , AX ; 正确。

(7) MOV AX, [SI+DI] ; 错误。存储器寻址中有效地址不能由两个变址寄存器组成

(8) MOV SS:[BX+SI+100H],BX ; 正确。

(9) AND AX, BL ; 错误。操作数类型不一致

(10) MOV DX, DS:[BP] ; 正确

(11) ADD [SI], 20H ; 错误,目的操作数类型不明确。

(12) MOV 30H, AL ; 错误, 目的操作数不能为立即数

(13) PUSH 2000H; 错误。堆栈指令的操作数不能是立即数

(14) MOV [SI], [2000H]; 错误。两个操作数不能同时为存储器操作数

(15) MOV SI, AL ; 错误。操作数类型不一致

(16) ADD [2000H], 20H; 错误,目的操作数类型不明确

(17) MOV CS, AX; 错误, 目的操作数不能为代码段寄存器

(18) INC [DI] ; 错误,目的操作数类型不明确

(19) OUT BX, AL; 错误。端口间接寻址的寄存器只能是 DX 寄存器

(20) SHL BX, 3 ; 错误。移位次数大于 1 时应该用 CL 寄存器

(21) XCHG CX, DS ; 错误。交换指令中不能出现段寄存器

(22) POP AL ; 错误。堆栈指令的操作数只能是字操作数(即 16 位操作数)

3. 写出下列指令中存储器操作数物理地址的计算表达式:

1) MOV AL, [DI] $(DS) \times 10H + (DI)$

(2) MOV AX, [BX+SI] $(DS) \times 10H + (BX) + (SI)$

(3) MOV 5[BX+DI],AL $(DS) \times 10H + (BX) + (DI) + 5$

(4) ADD AL, ES:[BX] $(ES) \times 10H + (BX)$

5) SUB AX, [1000H] $(DS) \times 10H + 1000H$

(6) $\text{ADC AX, [BX+DI+2000H]}$ $(\text{DS}) \times 10\text{H} + (\text{BX}) + (\text{DI}) + 2000\text{H}$

(7) MOV CX, [BP+SI] $(\text{SS}) \times 10\text{H} + (\text{BP}) + (\text{SI})$

(8) INC BYTE PTR[DI] $(\text{DS}) \times 10\text{H} + (\text{DI})$

4. 若 $(\text{DS}) = 3000\text{H}$, $(\text{BX}) = 2000\text{H}$, $(\text{SI}) = 0100\text{H}$,
 $(\text{ES}) = 4000\text{H}$, 计算下列各指令中存储器操作数的物理地址。

1) MOV [BX], AL

$$(\text{DS}) \times 10\text{H} + (\text{BX}) = 32000\text{H}$$

(2) $\text{ADD AL, [BX+SI+1000H]}$

$$(\text{DS}) \times 10\text{H} + (\text{BX}) + (\text{SI}) + 1000\text{H} = 33100\text{H}$$

(3) MOV AL, [BX+SI]

$$(\text{DS}) \times 10\text{H} + (\text{BX}) + (\text{SI}) = 32100\text{H}$$

(4) SUB AL, ES:[BX]

$$(\text{ES}) \times 10\text{H} + (\text{BX}) = 42000\text{H}$$

5. 若 $(\text{CS}) = \text{E}000\text{H}$, 说明代码段可寻址物理存储空间的范围。

代码段物理地址为: $(\text{CS}) : (\text{IP})$

所以可寻址物理地址范围为:

$(\text{CS}) \times 10\text{H} + (\text{IP})$, 其中 IP 的为 16 位寄存器, 取值范围为:

$0000\text{H} \sim 0\text{FFFFH}$

所以有寻址范围为: $0\text{E}0000\text{H} \sim 0\text{EFFFFH}$

6. 设 $(\text{SP}) = 2000\text{H}$, $(\text{AX}) = 3000\text{H}$, $(\text{BX}) = 5000\text{H}$, 问执行下面程序段后:

PUSH AX

PUSH BX

POP AX

$(\text{SP}) = 1\text{FFE}\text{H}$, $(\text{AX}) = 5000\text{H}$, $(\text{BX}) = 5000\text{H}$

7、试比较 SUB AL, 09H 与 CMP AL, 09H 这两条指令的异同。若 $(\text{AL}) = 08\text{H}$, 分别执行上述两条指令后, $(\text{AL}) = ?$, $\text{CF} = ?$, $\text{OF} = 0$, $\text{ZF} = ?$

(1) 相同点: 两条指令都能完成 $(\text{AL}) - 09\text{H}$ 的功能, 并且都影响六个状态标志位;

不同点: SUB 指令将运算结果回送到 AL 寄存器中, 而 CMP 指令不回送。

(2) ① $(\text{AL}) = \text{FFH}$, ② $(\text{AL}) = 08\text{H}$, $\text{CF} = 1$, $\text{OF} = 0$, $\text{ZF} = 0$

8、分别执行下列指令, 试求 AL 的内容及各状态标志位的状态。

(1) MOV AL, 19H ;

ADD AL, 61H ; (AL)=7AH

OF=0 SF=0 ZF=0 AF=0 PF=0 CF=0

(2) MOV AL, 19H ;

SUB AL, 61H ; (AL)=0DCH

OF=0 SF=1 ZF=0 AF=1 PF=0 CF=1

MOV AL, 19H ;

SUB AL, 61H ; (AL)=0B8H

OF=0 SF=1 ZF=0 AF=0 PF=1 CF=1

(3) MOV AL, 5DH ;

ADD AL, 0C6H ; (AL)=23H

OF=0 SF=0 ZF=0 AF=1 PF=0 CF=1

(4) MOV AL, 7EH ;

SUB AL, 95H ; (AL)=0E9H

OF=1 SF=1 ZF=0 AF=0 PF=0 CF=1

9. 选用最少的指令，实现下述要求的功能。

1) AH 的高 4 位清 0

AND AH, 0FH

(2) AL 的高 4 位取反

XOR AL, 0F0H

(3) AL 的高 4 位移到低 4 位，高 4 位清 0

MOV CL, 4

SHR AL, CL

(4) AH 的低 4 位移到高 4 位，低 4 位清 0

MOV CL, 4

SHL AL, CL

11. 设初值 (AX) = 0119H，执行下列程序段后 (AX) = ?

MOV CH, AH


```

ADD  AL, AH          (AX) =011AH
DAA                      (AX) =0120H  压缩型 BCD 码调整
XCHG AL, CH          (AL) =01H
ADC  AL, 34H          (AX) =0135H
DAA                      (AX) =0135H
MOV  AH, AL           (AH) =35H
MOV  AL, CH           (AL) =20H

```

结果: (AX) =3520H

12. 指出下列程序段的功能

1)

```

MOV  CX, 10
CLD
LEA  SI, First
LEA  DI, Second
REP  MOVSB

```

功能: 将 First 开始的 10 个字节数据复制到 Second 开始的内存中

2)

```

CLD
LEA  DI, [0404H]
MOV  CX, 0080H
XOR  AX, AX
REP  STOSW

```

功能: 将 0404H 开始的 80H 个字 (100H 个字节) 的内存填充 0

16. 已知 (DS) =091DH, (SS) =1E4AH, (AX) =1234H,

(BX) =0024H, (CX) =5678H, (BP) =0024H,

(SI) =0012H, (DI) =0032H, (09226H) =00F6H, (09228H) =1E40H, (1EAF6H) =091DH, 试求单独执行下列指令后的结果。

```

1) MOV  CL, 20H[BX][SI]      ; (CL) =      0F6H
(2) MOV  [BP][DI], CX        ; (1E4F6H) =    56F6H
(3) LEA  BX, 20H[BX][SI]     ; (BX) =      0056H
    MOV  AX, 2[BX]            ; (AX) =      1E40H
(4) LDS  SI, [BX][DI]        ;
    MOV  [SI], BX             ; ((SI)) =    0024H
(5) XCHG CX, 32H[BX]
    XCHG 20H[BX][SI], AX     ; (AX) =      5678H; (09226H) = 1234H

```

20. 设 (IP) =3D8FH, (CS) =4050H, (SP) =0F17CH,

当执行:

```
CALL 2000: 0094H
```

后, 试指出 (IP)、(CS)、(SP)、((SP))、

((SP) +1)、((SP) +2) 和 ((SP) +3) 的内容。

(IP) = 0094H

```

(CS) = 2000H
(SP) = 0F178H          压 4 个字节进栈, - 4H
((SP)) = 8FH
((SP) + 1) = 3DH
((SP) + 2) = 50H
((SP) + 3) = 40H

```

第五章

4. 假设程序中的数据定义如下:

```

LNAME          DB  1, 2, 3, '123'
ADDRESS        DB  30 DUP (?)
CTTY            DB  15 DUP (?)
CODE_LIST      DB  1, 7, 8, 3, 2

```

- 1) 用一条 MOV 指令将 LNAME 的偏移地址放入 BX
- (2) 用一条指令将 CODE_LIST 的头两个字节的內容放入 SI
- (3) 写一条伪指令定义符使 CODE_LENGTH 的值等于 CODE_LIST 域的实际长度。

答案 (1) MOV BX, OFFSET LNAME

(2) MOV SI, WORD PTR CODE_LIST

(3) CODE_LENGTH EQU \$ - CODE_LIST

5. 对于下面的数据定义, 试说明三条 MOV 指令的执行结果。

```

TABLEA        DW  10 DUP (?)
TABLEB        DB  10 DUP (?)
TABLEC        DB  '123'
MOV  AX, LENGTH TABLEA ; (AX) = 10
MOV  BL, LENGTH TABLEB ; (BL) = 10
MOV  CL, LENGTH TABLEC ; (CL) = 1

```

11. 试编写一程序, 要求比较两个字符串 STRING1 和 STRING2 所含字符是否相同, 若相同, 则显示 "MATCH", 若不相同则显示 "NOMATCH".

```

DATA SEGMENT

STRING1        DB  'ABCDEFGHJKLMNO'
STRING2        DB  'ABCDEFDDDDDD'
MSG1            DB  'MATCH','$'
MSG2            DB  'NOMATCH','$'

```

```

DATA ENDS
CODE SEGMENT
    ASSUME CS:CODE, DS:DATA
START:
    MOV AX, DATA
    MOV DS, AX
    MOV ES, AX
    LEA SI, STRING1

```

```

        LEA    DI,  STRING2
        MOV    CX,  15
        CLD
        REPE   CMPSB
        JZ     MATCH    ;相等
        LEA    DX,  MSG2
        JMP    DISPLAY
MATCH:
        LEA    DX,  MSG1
DISPLAY:
        MOV    AH, 09H
        INT    21H
        MOV    AX, 4C00H
        INT    21H
CODE ENDS
        END    START

```

16. 数据段中已定义了一个有 n 个字数据的数组 M ，试编写程序求出 M 中绝对值最大的数，把它放在数据段的 $M+2n$ 单元中，并将该数的偏移地址存放在 $M+2(n+1)$ 单元中。

```

DATA SEGMENT

n      EQU     6
M      DW      -66, 33, 55, -89, 28, 311
        DW      ?
        DB      ?

```

```

DATA ENDS
START:
        MOV    AX,  DATA
        MOV    DS,  AX
        LEA    BX,  M
        MOV    CX,  n
        MOV    SI,  0
        XOR    AX,  AX
        MOV    [BX][2*n],  AX
NEXT:
        MOV    AX,  [BX][SI]
        OR     AX,  AX
        JNS    L1
        NEG    AX
L1:
        CMP    [BX][2*n],  AX
        JG     L2
        MOV    [BX][2*n],  AX

```

```

        MOV     [BX][2*(n+1)], SI
L2:
        INC     SI
        INC     SI
        DEC     CX
        JNZ     NEXT

```

18. 已知数组 A 包含 15 个互不相等的整数, 数组 B 包含 20 个互不相等的整数。试编制一程序, 把既在 A 中又在 B 中出现的整数存放于数组 C 中。

DATA SEGMENT

```

A       DW     11, -22, 33, 88, -55, -67, 306, 39, 55, -90
          DW     28, 311, 65, 887, 911
B       DW     18, 25, 31, 88, -55, -65, 188, 30, -15, 77
          DW     252, 54, 102, 201, 87, -94, -22, -33, 911, 306
C       DW     15 DUP(0)

```

DATA ENDS

START:

```

        MOV AX, DATA
        MOV DS, AX
        MOV ES, AX
        LEA SI, A
        LEA BX, C

```

NEXT:

```

        LEA DI, B
        CLD
        MOV CX, 20
        MOV AX, [SI]
        REPNE SCASW
        JNZ     NOTFOND
        MOV [BX], AX
        INC     BX
        INC     BX

```

NOTFOND:

```

        INC     SI
        INC     SI
        CMP     SI, 30
        JL      NEXT

```

23. 编写程序, 将字节变量 BVAR 中的无符号二进制数(0~FFH)转换为 BCD 数, 在屏幕上显示结果。

DATA SEGMENT

```

BVAR    DB     165
BCD      DB     3 DUP(0) ;转换后以 3 位 ASCII 码保存

```

DB '\$'

DATA ENDS

START:

MOV AX, DATA

MOV DS, AX

MOV ES, AX

MOV AL, BVAR

LEA BX, BCD

MOV DI, 2

MOV CL, 10

NEXT:

MOV AH, 0

DIV CL

OR AH, 30H ;余数转换为 ASCII 码

MOV [BX+DI], AH

DEC DI

CMP AL, 0

JG NEXT

LEA DX, BCD

MOV AH, 09H

INT 21H ; 显示结果