

第一章

1. 什么是冯·诺依曼机?

答: 冯·诺依曼于 1945 年提出了存储程序的概念和二进制原理, 利用这种概念和原理设计的电子计算机系统统称为冯·诺依曼机。

它包括运算器、控制器、存储器、输入设备和输出设备五个组成部分。

早期的冯·诺依曼机结构上以运算器和控制器为中心, 随着计算机体系结构的发展, 现在已演化为以存储器为中心的结构。

2. 微处理器, 微型计算机, 微型计算机系统有什么联系与区别?

答: 微处理器是微型计算机系统的核心, 也称为 CPU (中央处理器)。主要完成: ①从存储器中取指令, 指令译码; ②简单的算术逻辑运算; ③在处理器和存储器或者 I/O 之间传送数

据; ④程序流向控制等。

微型计算机由微处理器、存储器、输入/输出接口电路和系统总线组成。

以微型计算机为主体, 配上外部输入/输出设备及系统软件就构成了微型计算机系统。

三者关系如下图:

3. 微处理器有哪些主要部件组成? 其功能是什么?

答: 微处理器是一个中央处理器, 由算术逻辑部件 ALU、累加器和寄存器组、指令指针寄存器 IP、段寄存器、标志寄存器、时序和控制逻辑部件、内部总线等组成。

算术逻辑部件 ALU 主要完成算术运算及逻辑运算。

累加器和寄存器组包括数据寄存器和变址及指针寄存器, 用来存放参加运算的数据、中间结果或地址。

指令指针寄存器 IP 存放要执行的下一条指令的偏移地址, 顺序执行指令时, 每取一条指令增加相应计数。

段寄存器存放存储单元的段地址, 与偏移地址组成 20 位物理地址用来对存储器寻址。

标志寄存器 flags 存放算术与逻辑运算结果的状态。

时序和控制逻辑部件负责对整机的控制: 包括从存储器中取指令, 对指令进行译码和分析, 发出相应的控制信号和时序, 将控制信号和时序送到微型计算机的相应部件, 使 CPU 内部及外部协调工作。

内部总线用于微处理器内部各部件之间进行数据传输的通道。

(3) $1238D=2326Q=4D6H$ (4) $92D=134Q=5CH$

12. 将下列十进制数转换为 8 位有符号二进制数。

(1) +32 (2) -12 (3) +100 (4) -9

答: (1) $[+32]_{原}=[+32]_{反}=[+32]_{补}=00100000B$

(2) $[-12]_{原}=10001100B$ $[-12]_{反}=11110011B$

(3) $[+100]_{原}=[+100]_{反}=[+100]_{补}=01100100B$

(4) $[-92]_{原}=11011100B$ $[-92]_{反}=10100011B$

13. 将下列十进制数转换为压缩和非压缩格式的 BCD 码。

(1) 102 (2) 44 (3) 301 (4) 1000

答: (1) $[102]_{压缩BCD}=00000001\ 00000010B$

$[102]_{非压缩BCD}=00000001\ 00000000\ 0$

(2) $[44]_{压缩BCD}=01000100B$

$[44]_{非压缩BCD}=00000100\ 0$

(3) $[301]_{压缩BCD}=00000011\ 00000001B$

$[301]_{非压缩BCD}=00000011\ 00000000\ 0$

(4) [1000]压缩 BCD=00010000 00000000B

[1000]非压缩 BCD=00000001 00000000 0

14. 将下列二进制数转换为有符号十进制数。

(1) 10000000B (2) 00110011B (3)

答：(1) 看作原码时真值为-0, 看作反码时真值为-127, 看作补码时真值为-128

(2) +51

(3) 看作

(4) 看作原码时真值为-9, 看作反码时真值为-118, 看作补码时真值为-119

第二章

1. 8086CPU 内部由哪两部分组成? 它们的主要功能是什么?

答: 8086CPU 内部由执行单元 EU 和总线接口单元 BIU 组成。

主要功能为: 执行单元 EU 负责执行指令。它由算术逻辑单元(ALU)、通用寄存器组、16 位标志寄存器(FLAGS)、EU 控制电路等组成。EU 在工作时直接从指令流队列中取指令代码, 对其译码后产生完成指令所需要的控制信息。数据在 ALU 中进行运算, 运算结果的特征保留在标志寄存器 FLAGS 中。

总线接口单元 BIU 负责 CPU 与存储器和 I/O 接口之间的信息传送。它由段寄存器、指令指针寄存器、指令队列、地址加法器以及总线控制逻辑组成。

2. 8086CPU 中有哪些寄存器? 各有什么用途?

答: 8086CPU 内部包含 4 组 16 位寄存器, 分别是通用寄存器组、指针和变址寄存器、段寄存器、指令指针和标志位寄存器。

寄存器、指令指针和标志位寄存器。

(1) 通用寄存器组 包含 4 个 16 位通用寄存器 AX、BX、CX、DX, 用以存放普通数据或地址, 也有其特殊用途。如 AX (AL) 用于输入输出指令、乘除法指令, BX 在间接寻址中作基址寄存器, CX 在串操作和循环指令中作计数器, DX 用于乘除法指令等。

(2) 指针和变址寄存器 BP、SP、SI 和 DI, 在间接寻址中用于存放基址和偏移地址。

(3) 段寄存器 CS、DS、SS、ES 存放代码段、数据段、堆栈段和附加段的段地址。

(4) 指令指针寄存器 IP 用来存放将要执行的下一条指令在现行代码段中的偏移地址。

(5) 标志寄存器 Flags 用来存放运算结果的特征。

3. 8086CPU 和 8088CPU 的主要区别是什么?

答: 8088CPU 的内部结构及外部引脚功能与 8086CPU 大部分相同, 二者的主要不同之处如下:

下:

(1) 8088 指令队列长度是 4 个字节, 8086 是 6 个字节。

(2) 8088 的 BIU 内数据总线宽度是 8 位, 而 EU 内数据总线宽度是 16 位, 这样对 16 位数的存储器读/写操作需要两个读/写周期才能完成。8086 的 BIU 和 EU 内数据总线宽度都是 16 位。

是 16 位。

(3) 8088 外部数据总线只有 8 条 AD7~AD0, 即内部是 16 位, 对外是 8 位, 故 8088 也称为准 16 位机。

(4) 8088 中, 用 IO/M 信号代替 M/IO 信号。

(5) 8088 中, 只能进行 8 位数据传输, BHE 不再需要, 改为 SS0, 与 DT/R 和 IO/M 一起决定最小模式中的总线周期操作。

4. 简要解释下列名词的意义: CPU, 存储器, 堆栈, IP, SP, BP, 段寄存器, 状态标志, 控制标志, 物理地址, 逻辑地址, 机器语言, 汇编语言, 指令, 内部总线, 系统总线。

答：CPU：中央处理器，是整个计算机系统的控制中心，主要功能是进行算术和逻辑运算，以及发出各种控制信号以协调整个系统正常工作。

存储器：是计算机系统记忆元件，用于存储指令和数据。

堆栈：在存储器中开辟的一个区域，用来存放需要暂时保存的数据。其操作特点是先进后出。

IP：指令指针寄存器，用来存放将要执行的下一条指令在现行代码段中的偏移地址。

SP：堆栈指针寄存器，用于指向当前栈顶单元。

BP：基址指针，间接寻址中用于存放基址，隐含段地址为 SS。

段寄存器：用于存放逻辑段的段地址。

状态标志：SF、ZF、OF、AF、PF 和 CF，反映运算结果的状态特征。

控制标志：IF、DF 和 TF，对可屏蔽中断、字符串操作指针变换方向和单步运行起控制作用。

物理地址：指存储器中存储单元的实际地址编码，是一种绝对地址，是 CPU 访问存储器的实际寻址地址，对于 8086 系统，地址范围为 00000H~FFFFFFH。

逻辑地址：由段基址和偏移地址组成，均为无符号的 16 位二进制数，程序设计时采用逻辑地址，可由逻辑地址变换为物理地址，物理地址=段基址 \times 16+偏移地址。

机器语言：直接用二进制代码指令表达的计算机语言，指令是用 0 和 1 组成的一串代码，计算机可以直接识别，不需要进行任何翻译。每台机器的指令，其格式和代码所代表的含义都是硬性规定的，故称之为面向机器的语言，也称为机器语言，是第一代计算机语言。

汇编语言：使用助记符表示的二进制代码指令语言，是一种符号化的机器语言，必须经编译程序将汇编语言编译成机器语言，计算机才能识别。

指令：能被计算机识别并执行的二进制代码，规定了计算机能完成的某一操作。

内部总线：微处理器内部各个部件之间传送信息的通道。

系统总线：微处理机机箱内的底板总线，用来连接构成微处理机的各个插件板，如 ISA 总线、EISA 总线、PCI 总线等。

5. 要完成下述运算或控制，用什么标志位判别？其值是什么？

- (1) 比较两数是否相等
- (2) 两数运算后结果是正数还是负数
- (3) 两数相加后是否溢出
- (4) 采用偶校验方式，判定是否要补 1
- (5) 两数相减后比较大小
- (6) 中断信号能否允许

答：(1) ZF，两数相减，若 ZF=1，则相等。

(2) SF，SF=1 则为负，否则为正

(3) 对有符号数：OF，OF=1 为溢出；对无符号数：CF，CF=1 为溢出

(4) PF，PF=1，不补 1

(5) 对有符号数：无溢出时 (OF=0)，如 ZF=1，则两数相等；如 ZF=0 且 SF=0，则被减数大；如 ZF=0 且 SF=1，则减数大；有溢出时 (OF=1)，如 SF=1，则被减数大；如 SF=0，则减数大；对无符号数：如 ZF=1，则两数相等；如 CF=0，则被减数大；如 CF=1，则减数大

(6) IF，IF=1，允许中断

6. 8086 系统中存储器采用什么结构？用什么信号来选中存储体？

答：8086 系统中，存储器采用分体结构，1MB 的存储空间分成两个存储体：偶地址存储体和奇地址存储体，各为 512KB。

使用 A0 和 BHE 来区分两个存储体。当 A0=0 时，选中偶地址存储体，与数据总线低 8 位相连，从低 8 位数据总线读/写一个字节。

当 BHE=0 时，选中奇地址存储体，与数据总线高 8 位相连，从高 8 位数据总线读/写一个字节。

当 A0=0, BHE=0 时，同时选中两个存储体，读/写一个字。

9. 段寄存器装入如下数据，写出每段的起始和结束地址。

(1) 1000H (2) 1234H (3) 2300H (4) E000H (5) AB00H

答：(1) 10000H~1FFFFH

(2) 12340H~2233FH

(3) 23000H~32FFFH

(4) E0000H~EFFFFH

(5) AB000H~BAFFFH

10. 根据下列 CS: IP 的组合，求出要执行的下一条指令的存储器地址。

(1) CS: IP=1000H: 2000H (2) CS: IP=2000H: 1000H

(3) CS: IP=1A00H: B000H (4) CS: IP=3456H: AB09H

答：(1) 12000H (2) 21000H (3) 25000H (4) 3F069H

11. 求下列寄存器组合所寻址的存储单元地址：

(1) DS=1000H, DI=2000H (2) SS=2300H, BP=3200H

(3) DS=A000H, BX=1000H (4) SS=2900H, SP=3A00H

答：(1) 12000H (2) 26200H (3) A1000H (4) 2CA00H

12. 若当前 SS=3500H, SP=0800H, 说明堆栈段在存储器中的物理地址，若此时入栈 10 个字节，SP 内容是什么？若再出栈 6 个字节，SP 为什么值？

答：物理地址：35000H~35800H。入栈 10 个字节后 SP 为 7F6H。再出栈 6 个字节后 SP 为 7FCH。

14. 存储器中每段容量最多 64K 字节，若用 debug 调试程序中的 r 命令，在屏幕上有如下显示：

示：

C: >debug

-r

AX=0000 BX=0000 CX=0079 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000

DS=10E4 ES=10F4 SS=21F0 CS=31FF IP=0100 NV UP DI PL NZ NA PO NC

(1) 试画出此时存储器分段示意图

(2) 写出状态标志 OF、SF、ZF、CF 的值

答：(1) 代码段首地址：31FF0H 当前指令地址：320F0H

数据段首地址：10E40H

堆栈段首地址：21F00H 堆栈段栈顶地址：31EEEH

附加段首地址：10F40H

(2) OF=SF=ZF=CF=0

15. 说明 8086 系统中“最小模式”和“最大模式”两种工作方式的主要区别是什么？

答：为了便于组成不同规模的系统，在 8086 芯片中设计了两种工作模式，即最小模式和最大模式。

最小模式用于单机系统，系统中所需要的控制信号全部由 8086 直接提供；最大模式用于多处理机系统，系统中所需要的控制信号由总线控制器 8288 提供。

16. 8086 系统中为什么要用地址锁存器？8282 地址锁存器与 CPU 如何连接？

答：为了减少引脚的数量，8086CPU 的地址引脚和数据引脚分时复用，为了保证在总线操作周期中地址信号能有效而稳定的输出，必须使用地址锁存器。

由于 8086 有 20 条地址线，故需使用三片 8282，其中 OE 接地，STB 与 CPU 的 ALE 相连，前两片 8282 的 DI0~DI7 分别与 CPU 的 AD0~AD15 相连，第三片 8282 的 DI0~DI3

分别与 CPU 的 AD16~AD19 相连，DI4 与 CPU 的 BHE 相连。

17. 哪个标志位控制 CPU 的 INTR 引脚？

答：IF，中断允许标志，IF=1 时，允许可屏蔽中断，IF=0 时，禁止可屏蔽中断。

18. 什么叫总线周期？在 CPU 读/写总线周期中，数据在哪个机器状态出现在数据总线上？

答：CPU 完成一次存储器访问或 I/O 端口操作所需要的时间称为一个总线周期，由几个 T 状态组成。在读/写总线周期中，数据在 T2~T4 状态出现在数据总线上。

20. 8086CPU 重新启动后，从何处开始执行指令？

答：重新启动后，CS=FFFFH，IP=0000H，故从物理地址为 FFFF0H 的位置开始执行指令。

21. 8086CPU 的最小模式系统配置包括哪几部分？

答：8086 最小模式系统配置包括：

8086CPU，存储器，I/O 接口芯片，1 片 8284 时钟发生器，3 片 8282 地址锁存器，2 片 8286 双向数据总线收发器。

第三章

1. 分别说明下列指令的源操作数和目的操作数各采用什么寻址方式。

- (1) MOV AX, 2408H (2) MOV CL, 0FFH (3) MOV BX, [SI]
- (4) MOV 5[BX], BL (5) MOV [BP+100H], AX (6) MOV [BX+DI], ' \$'
- (7) MOV DX, ES: [BX+SI] (8) MOV VAL[BX+DI], DX
- (9) IN AL, 05H (10) MOV DS, AX

答：(1) 立即数，寄存器 (2) 立即数，寄存器 (3) 寄存器间接，寄存器

- (4) 寄存器，寄存器相对 (5) 寄存器，寄存器相对 (6) 立即数，基址变址
- (7) 基址变址，寄存器 (8) 寄存器，相对基址变址
- (9) 直接端口寻址，寄存器 (10) 寄存器，寄存器

2. 已知：DS=1000H，BX=0200H，SI=02H，内存 10200H~10205H 单元的内容分别为 10H，2AH，3CH，46H，59H，6BH。下列每条指令执行完后 AX 寄存器的内容各是什么？

- (1) MOV AX, 0200H (2) MOV AX, [200H] (3) MOV AX, BX
- (4) MOV AX, 3[BX] (5) MOV AX, [BX+SI] (6) MOV AX, 2[BX+SI]

答：(1) 0200H (2) 2A10H (3) 0200H

- (4) 5946H (5) 463CH (6) 6B59H

3. 设 DS=1000H，ES=2000H，SS=3500H，SI=00A0H，DI=0024H，BX=0100H，BP=0200H，数据段中变量名为 VAL 的偏移地址值为 0030H，试说明下列源操作数字段的寻址方式是什么？物理地址值是多少？

- (1) MOV AX, [100H] (2) MOV AX, VAL (3) MOV AX, [BX]
- (4) MOV AX, ES: [BX] (5) MOV AX, [SI] (6) MOV AX, [BX+10H]
- (7) MOV AX, [BP] (8) MOV AX, VAL[BP][SI]
- (9) MOV AX, VAL[BX][DI] (10) MOV AX, [BP][DI]

答：(1) 直接，10100H (2) 直接，10030H (3) 寄存器间接，10100H

- (4) 寄存器间接，20100H (5) 寄存器间接，100A0H (6) 寄存器相对，10110H
- (7) 寄存器间接，35200H (8) 相对基址变址，352D0H
- (9) 相对基址变址，10154H (10) 基址变址，35224H

4. 写出下列指令的机器码

- (1) MOV AL, CL (2) MOV DX, CX (3) MOV [BX+100H], 3150H

答：(1) 10001010 11000001B

(2) 10001011 11010001B

(3) 11000111 10000111 00000000 00000001 01010000 00110001B

5. 已知程序的数据段为：

DATA SEGMENT

A DB '\$',10H

B DB 'COMPUTER'

C DW 1234H, 0FFH

D DB 5 DUP(?)

E DD 1200459AH

DATA ENDS

求下列程序段执行后的结果是什么。

MOV AL, A

MOV DX, C

XCHG DL, A

MOV BX, OFFSET B

MOV CX, 3[BX]

LEA BX, D

LDS SI, E

LES DI, E

答：MOV AL, A AL=24H

MOV DX, C DX=1234H

XCHG DL, A DL=24H, A=34H

MOV BX, OFFSET B BX=2

MOV CX, 3[BX] CX=5550H

LEA BX, D BX=000EH

LDS SI, E DS=1200H, SI=459AH

LES DI, E ES=1200H, DI=459AH

6. 指出下列指令中哪些是错误的，错在什么地方。

(1) MOV DL, AX (2) MOV 8650H, AX (3) MOV DS, 0200H

(4) MOV [BX], [1200H] (5) MOV IP, 0FFH (6) MOV [BX+SI+3], IP

(7) MOV AX, [BX][BP] (8) MOV AL, ES: [BP] (9) MOV DL, [SI][DI]

(10) MOV AX, OFFSET 0A20H (11) MOV AL, OFFSET TABLE

(12) XCHG AL, 50H (13) IN BL, 05H (14) OUT AL, 0FFEH

答：(1) 长度不匹配 (2) 立即数不能做目的操作数

(3) 段寄存器不能用立即数赋值 (4) 两个内存单元不能直接传送数据

(5) IP 不能用指令直接修改 (6) 指令中不能出现 IP

(7) BX/BP 应与 SI/DI 搭配 (8) 正确

(9) SI/DI 应与 BX/BP 搭配 (10) OFFSET 后应跟内存单元

(11) 偏移地址为 16 位，AL 长度不够 (12) 立即数不能用于 XCHG

(13) IN 必须用 AL/AX (14) 操作数顺序反向；地址应为 8 位

9. 下列指令完成什么功能？

(1) ADD AL, DH (2) ADC BX, CX (3) SUB AX, 2710H

(4) DEC BX (5) NEG CX (6) INC BL

(7) MUL BX (8) DIV CL

答: (1) AL+DH→AL (2) BX+CX+CF→BX (3) AX-2710H→AX

(4) BX-1→BX (5) 0-CX→CX (6) BL+1→BL

(7) AX*BX→DX, AX (8) AX/CL 商→AL, 余数→AH

14. 在下列程序段括号中分别填入以下指令

(1) LOOP NEXT (2) LOOPE NEXT (3) LOOPNE NEXT

试说明在这三种情况下, 程序段执行完后, AX, BX, CX, DX 的内容分别是什么。

START: MOV AX,01H

MOV BX,02H

MOV DX,03H

MOV CX,04H

NEXT: INC AX

ADD BX,AX

SHR DX,1

()

答: (1) AX=05H BX=10H CX=0 DX=0

(2) AX=02H BX=04H CX=03H DX=01H

(3) AX=03H BX=07H CX=02H DX=0

16. 中断向量表的作用是什么? 它放在内存的什么区域内? 中断向量表中的什么地址用于类型 3 的中断?

答: 中断向量表用来存放中断服务程序的入口地址。8086 的 256 类中断的入口地址要占用 1K 字节, 位于内存 00000H~003FFH 的区域内。中断向量表中 0000CH~0000FH 用于类型

3 的中断。

17. 设类型 2 的中断服务程序的起始地址为 0485: 0016H, 它在中断向量表中如何存放?

答: 物理地址 内容

00008H 16H

00009H 00H

0000AH 85H

0000BH 04H

18. 若中断向量表中地址为 0040H 中存放 240BH, 0042H 单元里存放的是 D169H, 试问:

(1) 这些单元对应的中断类型是什么?

(2) 该中断服务程序的起始地址是什么?

答: (1) 10H

(2) D169H:240BH

19. 简要说明 8086 响应类型 0~4 中断的条件是什么?

答: 类型 0: 除法错中断

执行除法指令时, 若除数为 0 或所得商超过寄存器能容纳的范围, 则自动产生类型 0 中断。

类型 1: 单步中断

若 CPU 的单步标志 TF=1, 则每执行完一条指令后, 自动产生类型 1 中断。

类型 2: 不可屏蔽中断 NMI

当 8086 的 NMI 引脚上接收到由低变高的电平变化时, 将自动产生类型 2 中断。

类型 3: 断点中断

若在程序中某位置设置断点，当程序执行到该断点时，则产生类型 3 中断。

类型 4：溢出中断

若溢出标志 OF=1，则可由溢出中断指令 INTO 产生类型 4 中断。

第四章

1. 下列变量各占多少字节？

A1 DW 23H, 5876H

A2 DB 3 DUP (?), 0AH, 0DH, '\$'

A3 DD 5 DUP(1234H, 567890H)

A4 DB 4 DUP(3 DUP(1, 2, 'ABC'))

答：A1 占 4 个字节

A2 占 6 个字节

A3 占 40 个字节

A4 占 60 个字节

2. 下列指令完成什么功能？

MOV AX, 00FFH AND 1122H+3344H

MOV AL, 15 GE 1111B

MOV AX, 00FFH LE 255+6/5

AND AL, 50 MOD 4

OR AX, 0F00FH AND 1234 OR 00FFH

答：(1) 将 0066H 传送给 AX

(2) 将 0FFH 传送给 AL

(3) 将 0FFFFH 传送给 AX

(4) AND AL, 02H

(5) OR AX, 00FFH

3. 有符号定义语句如下：

BUF DB 3,4,5,'123'

ABUF DB 0

L EQU ABUF-BUF

求 L 的值为多少？

答：L=6

4. 假设程序中的数据定义如下：

PAR DW ?

PNAME DB 16 DUP(?)

COUNT DD ?

PLENTH EQU \$-PAR

求 PLENTH 的值为多少？表示什么意义？

答：PAR 的偏移地址为 0，PLENTH 当前偏移地址 $\$=2+16+4=22$ ， $\$-PAR=22$ ，故 PLENTH 的值为 22。

若在 PLENTH 所在行有变量定义，则 \$ 表示该变量的偏移地址，即 \$ 表示 PLENTH 所在行的当前偏移地址。故 PLENTH 表示从当前行到 PAR 之间定义的变量所占的字节个数。

5. 对于下面的数据定义，各条 MOV 指令执行后，有关寄存器的内容是什么？

DA1 DB ?

DA2 DW 10 DUP(?)

DA3 DB 'ABCD'


```
MOV AX, TYPE DA1
MOV BX, SIZE DA2
MOV CX, LENGTH DA3
```

答：AX=1,BX=20,CX=1

6. 下段程序完成后，AH 等于什么？

```
IN AL, 5FH
TEST AL, 80H
JZ L1
MOV AH, 0
JMP STOP
L1: MOV AH, 0FFH
STOP: HLT
```

答：讨论从端口 5FH 输入的数据最高位的情况。若最高位为 1，则 AH=0；若最高位为 0，则 AH=0FFH。

8. 编两个通过过程完成将 AX 中存放的二进制数转换成压缩型 BCD 码以及将 BCD 码转换成二进制数。

答：(1) 将 AX 中的二进制数先后除以 1000, 100 和 10，每次除法所得的商，即是 BCD 数的千位、百位和十位数，余数是个位数。

子程序名：B2TOBCD

输入参数：AX=十六位二进制数

输出参数：CF=0，则 AX=4 位压缩型 BCD 码。CF=1，则要转换的数大于 9999，AX 不变。

使用寄存器：CX：存放除数，DX：存放中间结果。

```
B2TOBCD PROC FAR
CMP AX,9999 ; AX>9999, 则 CF 置 1
JBE TRAN
STC
JMP EXIT
TRAN: PUSH CX
PUSH DX
SUB DX,DX ; DX 清 0
MOV CX,1000 ; 计算千位数
DIV CX
XCHG AX,DX ; 商在 DX 中, 余数在 AX 中
MOV CL,4
SHL DX,CL ; DX 左移 4 位
MOV CL,100 ; 计算百位数
DIV CL
ADD DL,AL ; 百位数加到 DL 中, DX 左移 4 位
MOV CL,4
SHL DX,CL
XCHG AL,AH ; 余数保留在 AL 中
SUB AH,AH
MOV CL,10 ; 计算十位数
```

```

ADD DL,AL          DL 中, DX 左移 4 位
MOV CL,4
SHL DX,CL
ADD DL,AH ; 加个位数
MOV AX,DX ; 结果送到 AX 中
POP DX
POP CX
EXIT: RET
B2TOBCD ENDP

```

(2) 将 AX 中 4 位 BCD 码顺序乘以 1000, 100, 10 和 1, 然后求和即得。

子程序名: BCDTOB2

输入参数: AX=4 位压缩 BCD 码

输出参数: AX=十六位二进制数

使用寄存器: BX: 暂存数据, CX: 存放乘数, DX: 存放中间结果

```
BCDTOB2 PROC FAR
```

```
PUSH BX
```

```
PUSH CX
```

```
PUSH DX
```

```
MOV BX,AX
```

```
MOV CL,4
```

```
ROL AX,CL
```

```
AND AX,000FH
```

```
MOV CX,1000 ; 乘以 1000
```

```
MUL CX
```

```
MOV DX,AX
```

```
MOV AX,BX
```

```
MOV CL,8
```

```
ROL AX,CL
```

```
AND AX,000FH
```

```
MOV CL,100 ; 乘以 100
```

```
MUL CL
```

```
ADD DX,AX
```

```
MOV AX,BX
```

```
MOV CL,4
```

```
SHR AX,CL
```

```
AND AX,000FH
```

```
MOV CL,10 ; 乘以 10
```

```
MUL CL
```

```
ADD DX,AX
```

```
AND BX,000FH
```

```
ADD DX,BX
```

```
MOV AX,DX
```

```
POP DX
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/816242140241010101>