

《汇编语言程序设计》试题及答案

1. 对于有符号的数来说, 下列哪个值最大 (D)
A: 0F8H
B: 11010011B
C: 82
D: 123Q
2. 下列有关汇编语言中标号的命名规则中, 错误的是 (D)
A: 通常由字母打头的字符、数字串组成
B: 标号长度不能超过 31 个字符
C: ? 和 \$ 不能单独作为标号
D: . 号不可位于标号首
3. 8088/8086 存储器分段, 每个段不超过 (D)
A. 64K 个字
B. 32K 个字节
C. 1 兆个字节
D. 64K 个字节
4. 寻址指令 MOV CX, [BX + DI + 20] 使用的是哪一种寻址方式 (B)
A: 寄存器寻址
B: 相对基址变址寻址
C: 变址寻址
D: 基址变址寻址
5. 若 AX = -15 要得到 AX = 15 应执行的指令是 (A)
A. NEG AX
B. NOT AX
C. INC AX
D. DEC AX
6. 8086/8088 系统执行传送指令 MOV 时 (A)
A. 不影响标志位
B. 影响 DF 方向标志
C. 影响 SF 符号标志
D. 影响 CF 进位标志
7. 若要求一个操作数中的若干位维持不变, 若干位置 “1”, 可以使用 (B)
A: NOT
B: OR
C: AND
D: XOR
8. 下列指令中段默认为堆栈段的是 (C)
A. MOV AX, [BX+SI+10]
B. ADD AX, ES:[SI]
C. SUB [BX], [BP][DI]
D. MOV DX, [1000H]
9. 关于 8086/8088 微机系列, 下列说法哪个是正确的 (D)
A: 一个存储单元由 16 个二进制位组成, 简称字。
B: 当存储一个字数据时, 低字节放高地址位, 高字节放低地址位。

C: 在内存空间中, 可以无限分配段, 且段的大小不受限制。

D: 段与段之间可以邻接, 也可以重叠。

10. 下列关于堆栈的说法, 错误的是 (D)

A: 以“先入后出”为原则。

B: 栈区最高地址单元的前一个单元为栈底。

C: 运行中 SP 寄存器动态跟踪栈顶位置。

D: 压栈和弹出都是以字节为单位。

11. 表示过程定义结束的伪指令是 (A)

A. ENDP

B. ENDS

C. END

D. ENDM

12. BUF1 DB 3 DUP (0, 2 DUP (1, 2), 3)

COUNT EQU \$-BUF1

符号 COUNT 等价的值是 (B)

A. 6

B. 18

C. 16

D. 9

13. 下列标志位中, 可以用来判断计算结果正负的是 (B)

A: PF

B: SF

C: DF

D: OF

14. 下列指令正确的是 (CD)

A. MOV [100H], [BX]

B. MOV DS, ES

C. ADD V[BX], CX

D. MOV AX, 34H

15. 下列哪个寄存器是属于指针寄存器 (C)

A: SI

B: DX

C: SP

D: ES

二、填空题 (每小题 4 分, 共 20 分)

1. 下列程序段求数组 FLD 的平均值, 结果在 AL 中。请将程序填写完整 (不考虑溢出)

FLD DW 10, -20, 30, -60, -71, 80, 79, 56

_LEA SI, FLD_____

MOV CX, 8

XOR AX, AX

R1: _ADD AX, [SI]_____



扫码观看
视频讲解更清晰

ADD SI, 2

LOOP R1

MOV CL, 8

IDIV CL

HLT

2. MOV AX, 1090H

SUB AX, 4AE0H

JC L1

J0 L2

JMP L3

上述程序执行后 AX= C5B0 , 程序转向 L1

3. 假设寄存器 AX=1234H, DX=0A000H

MOV BX, 0

MOV CX, BX

SUB CX, AX

SBB BX, DX

MOV AX, CX

MOV DX, BX

上述程序段执行后 AX= 0EDCCH , DX= 5FFFH

4. MOV AX, 234H

MOV CL, 3

ROL AX, CL

DEC AX

MOV CX, 4

MUL CX

此段程序执行完后, (AX)= 467CH , (CX)= 4

5. DA1 DB 83H, 72H, 61H, 94H, 5AH

:

MOV CX, WORD PTR [DA1]



扫码观看
视频讲解更清晰

AND CX, 0FH

MOV AL, [DA1+3]

SHL AL, CL

上述指令序列执行后, AL=___0A0H___, CL=___03H___。

三、简答题 (每小题 4 分, 共 20 分)

1. 如 TABLE 为数据段中 0100 单元的符号名, 其中存放的内容为 0FF00H, 试问以下两条指令有什么区别? 指令执行后, AX 寄存器的内容是什么?

MOV AX, TABLE

MOV AX, OFFSET TABLE

2. 对于下面的数据定义

TAB1 DW 10 DUP(?)

TAB2 DB 10 DUP(?)

TAB3 DB '1234'

⋮

MOV AX, SIZE TAB1

MOV BL, SIZE TAB2

MOV CL, SIZE TAB3

三个寄存器中值分别为多少?

4. 已知某个人计算机 0000: 0080~0000: 00B0 区域的内容如下:

0000: 0080 CC A1 C8 00 80 89 A8 98 65 43 35 23 12 45 C4 00

0000: 0090 A4 78 98 56 90 00 00 78 49 03 28 B7 B8 9C A2 B8

0000: 00A0 CC A1 C3 00 80 89 A8 98 65 43 35 23 12 45 C4 00

0000: 00B0 2A 1B C8 04 00 33 11 23 DC 41 54 BB 90 45 33 01

试求执行软中断指令 INT 23H 时, 系统将转移到哪里进入中断服务程序。

5. 试计算逻辑地址 4017: 000A, 4015: 0002A 和 4010: 0007A 对应的物理地址。

这三个物理地址说明了什么问题?

四、编程题 (每小题 30 分, 共 30 分)

1. 试编制一程序, 统计 DT_W 字数组中正数、负数和 0 的个数, 并分别保存在

C1, C2 和 C3 变量中。

《汇编语言程序设计》试题及答案

一、 名词解释 (本大题共 5 小题, 每小题 3 分, 共 15 分) 试解释下列名词的含义。

1. 逻辑地址: 逻辑地址由两个 16 位的地址分量构成, 其中一个为段基值, 另一个为偏移量, 两个分量均为无符号数编码。物理地址=段基值 \times 16+偏移量
2. 物理地址: CPU 通过地址总线向外发送的 20 位地址, 在访问内存单元前, CPU 必须先将 CPU 内部的逻辑地址转换为物理地址。
3. 标志寄存器: 标志寄存器 (Flag Register) 又称程序状态字 (Program Status Word), 是 8086/088 中唯一按位操作的寄存器, 该寄存器为 16 位, 有效的标志位共 9 位。其中, CF, PF, AF, ZF, SF, OF 为状态标志, 用于反映最近一次影响标志位的算术或逻辑运算中运算过程、运算结果的一些性质; TF, IF, DF 为控制标志, 用于控制 CPU 对某些特定事件的处理方式以及 CPU 的工作模式。
4. 存储器寻址方式: 发生在指令的执行周期, 需要在执行周期内单独启动总线操作来完成。比寄存器寻址、立即数寻址速度都慢。
5. 补码: 计算机系统内部用于表示带符号数的一种编码, 正整数的补码为其真值本身, 负整数的补码为其真值变反加 1 的结果。

二、 计算题 (本大题共 5 小题, 每小题 4 分, 共 20 分) 试按照各小题的要求给出计算结果。

1. 将十进制数 100 分别转换为二进制、十六进制数, 它们的表示形式分别为多少?
解: 二进制为 01100100B, 十六进制为 64H。
2. 假设 (DS) = 0B00H, (BX) = 0210H, 对于指令 MOV DS: 120H[BX], AL, 其目的操作数的物理地址为多少?
解: EA = (BX) + 0120H = 0330H, 物理地址 = (DS) * 16 + EA = B330H
3. 假设 (BX) = 0210H, (DI) = 0060H, 对于指令 ADD DL, [BX][DI], 其源操作数的偏移量为多少?
解: 偏移量 = (BX) + (DI) = 0270H
4. 假设当前 (SP) = 0060H, 连续执行 5 条 PUSH 指令后, (SP) = ?
解: (SP) = 0060H - 5 * 2 = 0060H - 000AH = 0056H
5. 对于一个十进制数 -65, 其二进制补码表示形式为多少?
解: 先求二进制真值 01000001, 变反加一补码为 10111111

三、 排错题 (本大题共 4 小题, 每小题 5 分, 共 20 分) 每小題列出了一条指令, 判断该指令有无语法错误, 如果存在语法错误, 请指出具体的错误原因, 判断正确给 2 分, 分析正确给 3 分, 判断错误不给分。

1. PUSH 5588H 错误, 操作数不能用立即数;
2. MOV DS, 1720H 错误, MOV 指令不能对段寄存器直接传送立即数;
3. ADD AX, CL 错误, 寄存器长度不匹配;
4. AND AX, [DX] 错误, DX 不用于存储器寻址。

四、 程序分析题 (本大题共 6 小题, 每小题 5 分, 共 30 分) 每小題列出了一段小的程序片段和相关存储单元的初始值, 请按题目的要求分析各程序片段的运行结果。(寄存器中的内容请使用十六进制形式给出)

1. 阅读如下程序片段

MOV AL, 4CH

MOV BL, 0B5H

ADD AL, BL

执行上述程序片段后, (AL) = 01H (1分), (BL) = 0B5H (1分),

CF= 1 (1分), OF= 0 (1分), PF= 0 (1分)

2. 阅读如下程序片段

MOV AL, 0F3H

MOV DL, 0C4H

ADD AL, DL

AND AL, 0FH

执行上述程序片段后, (AL) = 07H (1分), (AF) = 不确定 (1分),

CF= 0 (1分), OF= 0 (1分), PF= 0 (1分)

3. 阅读如下程序片段

MOV AL, 7EH

MOV CL, 04H

ROL AL, CL

执行上述程序片段后, (AL) = 0F9H (2分), (CL) = 04H (1分),

CF= 1 (1分), OF= 无意义 (1分)

4. 阅读如下程序片段

MOV AX, 0632H

MOV BX, 0C202H

SUB AX, BX

INC AX

执行上述程序片段后, (AX) = 4431H (2分), (CF) = 1 (2分),

OF= 0 (1分)

5. 阅读如下程序片段, 假设 (DS) = 0100H, 字节单元 (01024H) = 56H, 字节单元 (01025H) = 28H

MOV BX, 0024H

LEA BX, [BX]

OR BX, 0

ADC BX, [BX]

执行上述程序片段后, (BX) = 287AH (3分), (CF) = 0 (2分),

OF= 0 (1分)

6. 阅读如下程序片段, 假设 (SP) = 0064H

MOV AX, 0001H

MOV BX, 0002H

PUSH AX

PUSH BX

POP AX

POP BX

执行上述程序片段后, (AX) = 0002H (2分), (BX) = 0001H (2分),

(SP) = 0064H (1分)

五. 程序设计题 (本大题共 2 小题, 第一小题 7 分, 第二小题 8 分, 共 15 分)

1. 试编写一程序片段, 实现 BL 高 4 位与低 4 位相互交换 (7 分)

解: MOV CL, 4;
ROL BL, CL

2. 试编写一程序片段, 不使用乘除法指令, 实现 $((AX) * 5 + 14) / 2$ 的功能 (AX 中的数据理解为补码) (8 分)

解: MOV BX, AX
MOV CX, 2
SAL AX, CX
ADD AX, BX
ADD AX, 14
SAR AX, 1

《汇编语言程序设计》试题及答案

六. 名词解释 (本大题共 5 小题, 每小题 3 分, 共 15 分) 试解释下列名词的含义。

6. 寄存器: 计算机系统内, 位于 CPU 内部的存储单元, CPU 访问此类存储单元不需要使用总线。
7. 物理地址: CPU 通过地址总线向外发送的 20 位地址, 在访问内存单元前, CPU 必须先将 CPU 内部的逻辑地址转换为物理地址。
8. 段基值: 逻辑地址的一个组成部分, 用于指示一个段在内存中的起始地址。
9. 寻址方式: 指令获取操作数的方式。
10. 补码: 计算机系统内部用于表示带符号数的一种编码, 正整数的补码为其真值本身, 负整数的补码为其真值变反加 1 的结果。

七. 计算题 (本大题共 5 小题, 每小题 4 分, 共 20 分) 试按照各小题的要求给出计算结果。

6. 将十进制数 100 分别转换为二进制、十六进制数, 它们的表示形式分别为多少?

答案: 二进制为 01100100B, 十六进制为 64H

7. 假设 (DS) = 0C00H, (BX) = 0210H, 对于指令 MOV DS: 100H[BX], AL, 其目的操作数的物理地址为多少?

答案: $EA = (BX) + 0100H = 0310H$, 物理地址 = $(DS) * 16 + EA = 0C000H + 0310H = 0C310H$

8. 假设 (BX) = 0100H, (SI) = 0020H, 对于指令 ADD DL, [BX][SI], 其源操作数的偏移量为多少?

答案: $EA = (BX) + (SI) + 0 = 0120H$

9. 假设当前 (SP) = 0060H, 连续执行 5 条 PUSH 指令后, (SP) = ?

答案: 每执行一条 PUSH 指令, (SP) 被减 2, 因此执行 5 条 PUSH 指令后, (SP) = 0060H - 5*2 = 0060H - 000AH = 0056H

10. 对于一个十进制数 -80, 其二进制补码表示形式为多少?

答案: 首先求得二进制真值为 -01010000B, 负数补码为变反加 1 的结果, 即 10110000B

八. 排错题 (本大题共 4 小题, 每小题 5 分, 共 20 分) 每小题列出了一条指令, 判断该指令有无语法错误, 如果存在语法错误, 请指出具体的错误原因, 判断正确给 2 分, 分析正确给 3 分, 判断错误不给分。

5. MOV CS, 0600H

参考答案: 错误, 不能使用 MOV 指令传递立即数给 CS

6. PUSH 074CH

参考答案: 错误, 单操作数指令不能使用立即数

7. ADD AX, CL

参考答案: 错误, 寄存器长度不匹配

8. OR BX, [CX]

参考答案: 错误, CX 不能用于存储器寻址

九. 程序分析题 (本大题共 6 小题, 每小题 5 分, 共 30 分) 每小题列出了一段小的程序片段和相关存储单元的初始值, 请按题目的要求分析各程序片的运行结果。(寄存器中的内容请使用十六进制形式给出)

7. 阅读如下程序片段

```
MOV AL, 4CH
MOV BL, 86H
ADD AL, BL
```

执行上述程序片段后, (AL) = 0D2H (1 分), (BL) = 86H (1 分),
CF = 0 (1 分), OF = 0 (1 分), PF = 1 (1 分)

8. 阅读如下程序片段

```
MOV AL, 0F3H
```



扫码观看
视频讲解更清晰

MOV DL, 0C4H
ADD AL, DL
AND AL, 0FH

执行上述程序片段后, (AL) = 07H (1分), (AF) = 不确定 (1分),
CF= 0 (1分), OF= 0 (1分), PF= 0 (1分)

注意: 逻辑运算指令中, 除 NOT 指令外, CF、OF 均置 0, AF 为不确定

9. 阅读如下程序片段

MOV AL, 6FH
MOV CL, 04H
ROL AL, CL

执行上述程序片段后, (AL) = 0F6H (2分), (CL) = 04H (1分),
CF= 0 (1分), OF= 不确定 (1分)

注意: 移位指令中, 当指令的移位次数大于 1 时, OF 为不确定

10. 阅读如下程序片段

MOV AX, 0632H
MOV BX, 0C202H
SUB AL, BL
SBB AH, BH
INC AX

执行上述程序片段后, (AX) = 4431H (2分), (CF) = 1 (2分),
OF= 0 (1分)

注意: INC 指令不影响 CF 标志位

11. 阅读如下程序片段, 假设 (DS)=0100H, 字节单元 (01024H)=56H, 字节单元 (01025H)=28H

MOV BX, 0024H
LEA BX, [BX]
AND BX, 0FFH
ADC BX, [BX]

执行上述程序片段后, (BX) = 287AH (3分), (CF) = 0 (2分),
OF= 0 (1分)

12. 阅读如下程序片段, 假设 (SP) = 0064H

```
MOV AX, 0001H
MOV BX, 0002H
PUSH AX
PUSH BX
POP AX
POP BX
```

执行上述程序片段后, (AX) = 0002H (2分), (BX) = 0001H (2分),
(SP) = 0064H (1分)

十. 程序设计题 (本大题共 3 小题, 每小题 5 分, 共 15 分) (注: 不是唯一答案)

3. 试编写一程序片段, 实现 BL 高 4 位与低 4 位相互交换 (5 分)

参考解答:

```
MOV CL, 4
ROL BL, CL
```

4. 试编写一程序片段, 不使用乘除法指令, 实现 $((AX) * 5 + 14) / 2$ 的功能 (AX 中的数据理解为补码) (5 分)

参考解答:

```
MOV BX, AX
MOV CL, 2
SAL AX, CL
ADD AX, BX
ADD AX, 14
SAR AX, 1
```

5. 试编写一完整程序, 实现 88242567H 与 0A435C218H 两个 32 位无符号数相加的功能。(要求编写完整源程序, 含程序框架) (5 分)

参考解答:

```
DATA SEGMENT
VAR1 DD 88242567H
VAR2 DD 0A435C218H
VAR3 DD ?
DATA ENDS
```

```
STACK1 SEGMENT STACK
        DW 40H DUP(0)
```

```
STACK1 ENDS
CODE SEGMENT
ASSUME CS:CODE, DS:DATA, SS:STACK1
BEGIN: MOV AX, DATA
        MOV DS, AX
        MOV AX, VAR1
        ADD AX, VAR2
        MOV VAR3, AX
        MOV AX, VAR1+2
        ADC AX, VAR2+2
        MOV VAR3+2, AX
        MOV AH, 4CH
        INT 21H
CODE ENDS
END BEGIN
```



扫码观看
视频讲解更清晰