

# 微机原理、汇编语言与接口技术 周杰英 张萍 习题答案汇总

微机原理、汇编语言与接口技术 周杰英 张萍 习题答案汇总

## 第1章 绪论

### 习题与参考答案(部分)

1. 把下列二进制数转换成十进制数、十六进制数及BCD码形式。

(1)  $10110010B =$

(2)  $01011101.101B =$

解:

(1)  $10110010B = 178D = B2H = (0001\ 0111\ 1000$

(2)  $01011101.101B = 93.625D = 5D.AH$

$= (1001\ 0011.0110\ 0010\ 0101)BCD$

2. 把下列十进制数转换成二进制数。

(1)  $100D =$

(2)  $1000D =$

(3)  $67.21D =$

解:

(1)  $100D = 01100100B$

(2)  $1000D = 1111101000B$

(3)  $67.21D = 1000011.0011B$

3. 把下列十六进制数转换成十进制数、二进制数。 )BCD

(1)

(2)

解:  $2B5H = 4CD.A5H =$

(1)  $2B5H = 693D = 0010\ 1011\ 0101B$

(2)  $4CD.A5H = 1229.6445D = 0100\ 1100\ 1101.1010\ 0101\ B$

4. 计算下列各式。

(1)  $A7H+B8H =$

(2)  $E4H-A6H =$

解:

(1)  $A7H+B8H = 15FH$

(2)  $E4H-A6H = 3EH$

5. 写出下列十进制数的原码、反码和补码。

(1)  $+89$

(2)  $-37$

解:

(1)  $[+89]$  原码、反码和补码为:  $01011001B$

(2)  $[-37]$  原码 =  $10100101\ B$

$[-37]$  反码 =  $11011010\ B$

$[-37]$  补码 =  $11011011\ B$

6(求下列用二进制补码表示的十进制数

(1)  $(01001101)$  补 =

(2)  $(10110101)$  补 =

解:

(1)  $(01001101)$  补 =  $77D$

(2)  $(10110101)$  补 =  $-75D$

7(请用 8 位二进制数写出下列字符带奇校验的 ASCII 码。

(1)C: 1000011 (2)O: 1001111

(3)M: 1001101 (4)P: 1010000

解:

(1)C:0 1000011 (2)O: 0 1001111

(3)M:1 1001101 (4)P: 1 1010000

8(请用 8 位二进制数写出下列字符带偶校验的 ASCII 码。

(1)+:0101011 (2)=: 0111101

(3)#:0100011 (4)>: 0111110

解:

(1)+:0 0101011 (2)=: 1 0111101

(3)#:1 0100011 (4)>: 1 0111110

9(叙述 CPU 中 PC 的作用。

解:PC 是 CPU 中的程序计数器,其作用是提供要执行指令的地址。

## 第 2 章 微处理器的结构

### 习题与参考答案

1. 8086, 8088 CPU 由哪两大部分组成, 请分别叙述它们的功能。

解:8086/8088 CPU 均由两个独立的逻辑单元组成, 一个称为总线接口单元 BIU(Bus Interface Unit), 另一个称为执行单元 EU(Execution Unit)。

总线接口单元 BIU 取指令时, 从存储器指定地址取出指令送入指令队

列排队; 执行指令时, 根据 EU 命令对指定存储器单元或 I/O 端口存取数据。

执行单元 EU 任务是执行指令, 进行全部算术逻辑运算、完全偏移地址的计算, 向总线接口单元 BIU 提供指令执行结果的数据和偏移地址, 并对通用寄存器和标志寄存器进行管理。

2. 8086, 8088 与传统的计算机相比在执行指令方面有什么不同?这样的设计思想有什么优点?

解:总线接口部分和执行部分是分开的, 每当 EU 执行一条指令时, 造成指令队列空出 2 个或空出一个指令字节时, BIU 马上从内存中取出下面一条或几条指令, 以添满他的指令队列。这样, 一般

情况下, CPU 在执行完一条指令后, 便可马上执行下一条指令, 不像以往 8 位 CPU 那样, 执行完一条指令后, 需等待下一条指令。即 EU 从指令队列中取指令、执行指令和 BIU 补充指令队列的工作是同时进行的。这样大大提高了 CPU 利用率, 降低了系统都对存储器速度的要求。

3. 状态标志和控制标志有何不同, 8086, 8088 的状态标志和控制标志分别有哪些,

解:8086/8088 CPU 设立了一个两字节的标志寄存器, 共九个标志。其中标志位 O、S、Z、A、P、C 是反映前一次涉及 ALU 操作结果的状态标志, D、I、T 是控制 CPU 操作特征的控制标志。

4. 8086, 8088 CPU 寻址存储器时, 什么是物理地址, 逻辑地址, 它们之间有何关系,

解:8086/8088 系统中每个存储单元都有一个物理地址, 物理地址就是存储单元的实际地址编码。在 CPU 与存储器之间进行任何信息交换时,

需利用物理地址来查找所需要的访问的存储单元。逻辑地址由段地址和偏移地址两部分组成。

逻辑地址的表示格式为:段地址:偏移地址。知道了逻辑地址, 可以求出它对应的物理地址:物理地址=段地址 $\times$ 10H+偏移地址。

5. 段寄存器 CS, 1200H, 指令指针寄存器 IP, FF00H, 此时, 指令的物理地址为多少, 指向这一物理地址的 CS 值和 IP 值是唯一的吗,

解:根据:物理地址=段地址 $\times$ 10H+偏移地址, 指令的物理地址为 21F00H。指向这一物理地址的 CS 值和 IP 值不唯一。

6. 8086 CPU 中 BHE 信号和 A0 信号是通过怎样的组合解决存储器和外

设端口的读, 写的, 这种组合决定了 8086 系统中存储器偶地址体及奇地址体之间应该用什么信号区分, 怎样区分,

解:若存取一个字节的数据, 总是用一个总线周期来完成该操作;若存取一个字, 则依该字是规则字还是不规则字需用一个或两个总线周期来完成。对规则字, 只用一个总线周期来完成 16 位数据的传送, 而对于非规则字则用相邻两个总线周期来完成该字的存储操作, 先取其奇地址字节(即数据的低位字节), 然后存取偶地址字节。在组成存储系统时, 总是偶地址单元的数据通过 AD0~AD7 传送, 而奇地址单元的数据通过 AD8~AD15 传送, 即通过总线高字节传送。BHE 信号和 A0 信号控制存储器读写见下表:

### 第 3 章 80x86 指令系统

#### 习题与参考答案

1、指出下列指令中操作数的寻址方式(8086 系统)。

- (1) MOV AX, 100 (2) MOV AX, [100]  
(3) MOV DL, [BP+SI] (4) MOV [BX], CX  
(5) MOV DX, [SI] (6) MOV 1234H[BX], DS  
(7) MOV [DI+5678H], AL (8) MOV 12[BP][DI], BH  
(9) POP CX (10) MOV AX, [BX+SI+10] 解:

2、指出下列指令中操作数的寻址方式(80386 系统)。

- (1) MOV [EBX+12H], AX (2)  
[EBP+EDI\*2], EBX  
(3) MOV [EBX\*4], BX (4) MOV EDX

[EAX+ESI\*8-12H]

解:

MOV,

3、指出以下指令中，哪些指令是非法的，并说明为什么(8086 系统)。

- (1) PUSH 1234H (2) MOV CS, AX
- (3) IN AX, 300 (4) MOV AX, [DX+12]
- (5) MOV BX, [BX] (6) MOV DS, 1000H
- (7) XCHG AL, AL (8) MOV AL, 100H
- (9) MOV DX, AL (10) LEA BL, [BX+5]
- (11) LEA DX, BX (12) MOV [1000H], 12H
- (13) ADD AX, DS (14) SUB [0100H], BYTE PTR [0001]
- (15) SHL BL, 2 (16) SHR CL, CL
- (17) MUL AL, BL (18) INT 400

解: (1) 非法, 8086 系统不允许 PUSH 指令的操作数为立即数;

(2) 非法, MOV 指令中 CS 不能作为目的操作数;

(3) 非法, 端口号大于 0FFH;

(4) 非法, DX 不能出现在中括号[]内;

(5) 合法;

(6) 非法, 不能把立即数赋给段寄存器;

(7) 合法;

(8) 非法, 源操作数大于 0FFH, 不能赋给 8 位寄存器 AL;

(9) 非法, 源操作数是 8 位寄存器, 目的操作数却是 16 位寄存器, 不匹配;

(10) 非法, LEA 指令得到的是 16 位偏移量, 但目的操作数是 8 位寄存器;

(11) 非法, LEA 指令的源操作数只能是存储器操作数;

- (12) 非法，指令存在歧义，无法确定传送的是字节还是字；
- (13) 非法，段寄存器不能参与算术运算；
- (14) 非法，源和目的寄存器不能同为存储器操作数；
- (15) 非法，在 8086 系统中，移位次数大于 1 时需要将移位次数存放到 CL 中；
- (16) 合法；
- (17) 非法，在 8086 系统中 MUL 指令没有双操作数的用法；
- (18) 非法，中断类型号大于 0FFH。

4、指出以下指令中，哪些指令是非法的，并说明为什么 (80386 系统)。

- (1) MOV AX, 12[EBX][SI\*16] (2) RCR EAX, 10
- (3) PUSH 5678H (4) POP 1000H
- (5) MOV [EAX+EAX], EAX (6) MOV AL, [ESP+ESP\*2]
- (7) MOV BL, [AX+12] (8) IMUL AL, BL, 12

解: (1) 非法，比例因子只能是 1、2、4、8，不能是 16；

- (2) 合法；
- (3) 合法；
- (4) 非法，POP 指令的目的操作数不能是立即数；
- (5) 合法；
- (6) 非法，ESP 只能作为基址寄存器，不能作为变址寄存器；
- (7) 非法，在 386 系统中 AX 不能作为基址寄存器，也不能作为变址寄存器；
- (8) 合法。

5、假设 (AX)=1234H, (BX)=5678H, (SP)=1000H, 指出执行下面的程序段后，各相关寄存器及堆栈段中的内容。

PUSH AX ; (AX)=?, (BX)=?, (SP)=?, 栈顶字节[SP]=?, 栈顶第二字节[SP+1]=?

PUSH BX ; (AX)=?, (BX)=?, (SP)=?, 栈顶字节[SP]=?, 栈顶第二字节[SP+1]=?

POP AX ; (AX)=?, (BX)=?, (SP)=?, 栈顶字节[SP]=?, 栈顶第二字节[SP+1]=?

解: (AX)=1234H, (BX)=5678H, (SP)=0FFE H, 栈顶字节[SP]=34H, 栈顶第二字节[SP+1]=12H

(AX)=1234H, (BX)=5678H, (SP)=0FFCH, 栈顶字节[SP]=78H, 栈顶第二字节[SP+1]=56H

(AX)=5678H, (BX)=5678H, (SP)=0FFE H, 栈顶字节[SP]=34H, 栈顶第二字节[SP+1]=12H

6、假设 A、B、C、D、X、Y 为字节变量，AA、BB、YY 为字变量，试利用算术运算指令编写程序段，完成以下各算术运算题。

(1) 计算  $YY \leftarrow A+B \times C$ ，其中 A、B、C 都是无符号数；

(2) 计算  $(AA+BB) / (C-D)$ ，商赋给 X，余数赋给 Y，其中 AA、BB、C、D 都是带符号数；

(3) 计算  $YY \leftarrow (A-B) \times C$ ，其中 A、B、C 都是未组合 BCD 码。 解：

(1) MOV AL, B

MUL C

ADD AL, A

ADC AH, 0

MOV YY, AX

(2) MOV AX, AA

ADD AX, BB



MOV BL, C

SUB BL, D

IDIV BL

MOV X, AL

MOV Y, AH

(3)MOV AL, A

SUB AL, B

AAS

MUL C

AAM

MOV YY, AX

7、利用移位指令编写程序段，实现以下运算。

(1)计算  $AX \times AX \times 10$ ，其中 AX 为无符号数；

(2)32 位带符号数存放在寄存器 DX 和 AX 中，其中 DX 存放高 16 位，AX 存放低 16 位计算  $(DX, AX) \times (DX, AX) \times 2$ 。

(3)32 位带符号数存放在寄存器 DX 和 AX 中，其中 DX 存放高 16 位，AX 存放低 16 位计算  $(DX, AX) \times (DX, AX) / 2$ 。

解：

(1)SHL AX, 1

MOV BX, AX

SHL AX, 1

SHL AX, 1

ADD AX, BX

(2)SAL AX, 1

RCL DX, 1

(3) SAR DX, 1

RCR AX, 1

8、指出以下指令执行后，标志寄存器 OF、SF、ZF、AF、PF、CF 的状态。

(1) (AL)=0FFH, (BL)=20H, 执行指令:ADD (2) (AL)=01H, (BL)=02H, 执行指令:CMF (3) (AL)=0FFH, 执行指令:INC AL (4) (AL)=0, 执行指令:DEC AL

(5) (AL)=0F0H, (BL)=04H, 执行指令:IMUL (6) (AX)=1F0H, (BL)=08H, 执行指令:DIV (7) (AL)=12H, (BL)=34H, 执行指令:TEST (8) (AL)=98H, 执行指令:SAL AL, 1 解:

BL AL, BL BL BL AL, BL AL,

注:0 表示结果为 0, 1 表示结果为 1, ×表示结果不变, —表示结果不确定。

9、在 8086 系统下，编写实现如下功能的程序段：(1)从地址为 80H 的端口中读入一个字节；

(2)如果该字节最高位为“1”，则将字节 0FFH 送到地址为 81H 的端口中，并退出程序；

(3)如果该字节最高位为“0”，则转向第(1)步，继续循环扫描。

解：参考程序段：

AGAIN: IN AL, 80H

TEST AL, 80H

JZ AGAIN

MOV AL, 0FFH

OUT 81H, AL

10、我们知道，MOV 指令并不能直接修改 CS、IP 的值，但事实上，

我们还可以通过其他方法来达到修改 CS、IP 的目的。试编写一个程序段，使该程序段运行后，(CS)=0100H，(IP)=0000H。 解：

方法一、利用段间返回指令 RETF：

```
MOV AX, 0100H
```

```
PUSH AX ;段基址入栈
```

```
MOV AX, 0
```

```
PUSH AX ;偏移量入栈
```

```
RETF
```

方法二、利用 JMP 指令实现段间跳转，假设 ADDR 为双字变量：MOV WORD PTR ADDR, 0 ;WORD PTR ADDR?偏移量

```
MOV WORD PTR ADDR+2, 0100H ;WORD PTR ADDR+2?段基址
```

```
JMP DWORD PTR ADDR
```

## 第 4 章 汇编语言程序设计

### 习题与参考答案

1、指出以下数据定义伪指令所分配的字节数(8086 系统)。

(1)DATA1 DB 10, , , 'A'

(2)DATA2 DW 10 DUP(2, 3 DUP(?), 1)

(3)DATA3 DB 'HELLO, WORLD~' , '\$'

(4)DATA4 DW DATA4

解：(1)3 字节；(2)100 字节；(3)13 字节；(4)2 字节。

2、指出以下数据定义伪指令所分配的字节数(80386 系统)。

(1)DATA1 DF 12, 34, 56

(2)DATA2 DF DATA2

(3)DATA3 DQ 0, 10 DUP(?)

(4) DATA4 DT 0, 1, 2

解: (1) 18 字节; (3) 6 字节; (3) 88 字节; (4) 30 字节。

3、指出以下指令中哪些是无效的, 并说明原因。

(1) ADDR DB \$

(2) DATA DB FOH, 12H

(3) 1\_DATA DW 1234H

(4) @VAR DW VAR1 ; VAR1 为一个字节变量

(5) MOV AX, [10-VAR1] ; VAR1 为一个字变量

(6) MOV BX, [VAR2\*2+1] ; VAR2 为一个字变量

解:

(1) 非法, 地址计数器\$是 32 位数;

(2) 非法, 以字母开头的 16 进制数应在前面加“0”, 即应为“0FOH”;

(3) 非法, 变量名非法, 不能以数字开头;

(4) 合法;

(5) 非法, 地址表达式不能是“常数-地址”;

(6) 非法, 地址表达式不能出现乘法。

4、假设已定义数据段如下:

DATA SEGMENT