

汇编语言习题课及期中课程小结

(2013/10/25)

第1次作业 (P. 15)

1.

- (1) $(29.6875)_{10}$
- (2) $(10010011)_2$, $(93)_{16}$
- (3) $(940)_{10}$
- (4) $(01100000)_2$ (对应十进制96, 将96变成二进制)
- (5) $(FF)_{16}$, $(-5B)_{16}$ (写成 $(DB)_{16}$ 是原码形式, 不是真值)
- (6) $(-0E)_{16}$ (写成 $(-E)_{16}$ 严格上讲是不正确的, 注意字长为8位), $(-72)_{16}$, $(-71H)_{16}$, $(8E)_{16}$
- (7) (a) 45H 或 69, 无溢出 (注意要有计算过程)
 - (b) -6EH 或 -110, 无溢出
 - (c) -46H 或 -70, 无溢出
 - (d) -76H 或 -118, 有溢出

容易出错:

- (1) 真值数与原码的差别
- (2) 溢出的判断方法 (应使用公式计算)

第2次作业（补充）

一个有16个字的数据区，它的起始地址为70A0H:DDF6H,请写出这个数据区首末字单元的物理地址。

首字单元 逻辑地址: 70A0H:DDF6H
 对应物理地址: $70A00H + DDF6H = 7E7F6H$

末字单元 逻辑地址:
70A0H:(DDF6H+30=DDF6H+1EH=DE14H)
对应物理地址: $70A00H + DE14H = 77E814H$

容易出错: 末字单元和末字节单元的含义

第2, 3次作业 (p.85)

1.

(1) 1234H:5678H ---> 179B8H

(2) 2F34H:2F6H 即 2F34H:02F6H ---> 2F636H

(3) 576AH:1024H ---> 586C4H

(4) 2FD0H:100H 即 2FD0H:0100H ---> 2FE00H

容易出错:

- (1) 16进制加法
- (2) 逻辑地址长度 (16位)

(p.85)

2. 寻址方式

(1) **MOV BX,50** 寄存器，立即

(2) **CMP [BX],100** → **CMP BYTE PTR[BX], 100** 间址访问数据段，立即

(3) **ADD [SI], 1000** → **ADD WORD PTR [SI],1000** 间址访问数据段，立即

(4) **MOV BP,SP** 寄存器，寄存器

(5) **MOV BX,[BP+4]** 寄存器，基址访问堆栈段

(6) **MOV AX,[BX+SI+5]** 寄存器，基+变 访问数据段

容易出错： (1) 段约定

(2) 寻址方式针对操作数中某一个具体的操作数

3.

(1) CF23H+C678H=959BH

1100111100100011
+ 1100011001111000

(1)1001010110011011

A=0,C=1,O=0,P=0,S=1,Z=0

容易出错: (1) CPU判溢 (应使用公式计算)

OF=F(Xs,Ys,Zs)

(2) **P, A**标志均只针对计算中最低位字节

(2) 6398H+C678H=2A10H

**0110001110011000
+ 1100011001111000
-----**

(1)0010101000010000

A=1,C=1,O=0,P=0,S=0,Z=0

(3) 94FBH+C678H=5B73H

**1001010011111011
+ 1100011001111000**

(1)0101101101110011

A=1,C=1,O=1,P=0,S=0,Z=0

(4) 65E2H+C678H=2C5AH

0110010111100010
+ 1100011001111000

(1)0010110001011010

A=0,C=1,O=0,P=1,S=0,Z=0

4.

(1) OR AH,0F0H

容易出错:

(1) AND与OR指令的具体作用.

(2) 0F0H写成F0H

4.

(2) XOR BX,0F00FH

(3) AND CX, 0FFF0H

5. (1)

将First/Second 按字节方式交换，需要进行2次交换

```
MOV AL,FIRST  
XCHG SECOND,AL  
MOV FIRST,AL  
MOV AL,FIRST+1  
XCHG SECOND+1,AL  
MOV FIRST+1,AL
```

将First/Second 按字方式交换，需要进行1次交换

```
MOV AX, WORD PTR FIRST  
XCHG WORD PTR SECOND,AX  
MOV WORD PTR FIRST,AX
```

容易出错：

- (1) 没有完成操作
- (2) 2个内存单元不通过寄存器直接交换
- (3) 字方式交换不使用PTR

5.(2)

```
PUSH WORD PTR FIRST  
PUSH WORD PTR SECOND  
POP WORD PTR FIRST  
POP WORD PTR SECOND
```

容易出错：

- (1) 不使用**PTR**
- (2) 将直接作为堆栈指令操作数 如**PUSH FIRST**

6.

TABLE DW 158,258,358,458

内存单元内容依次为: (从左到右, 由低位地址到高位地址)

9EH	00H	02H	01H	66H	01H	CAH	01H
TABLE+0	TABLE+1	TABLE+2	TABLE+3	TABLE+4	TABLE+5	TABLE+6	TABLE+7

[BX + SI] 寻址方式指向的单元为**[TABLE+3]**, 取连续**2**个字节单元的内容 ---》**AX**

AX 中数据为 **6601H** (26113)

容易出错:

- (1)没有将**10**进制数值转换为**16**进制数值就开始往下做
- (2) **DB,DW,DD**定义的变量在内存单元中具体存放
- (3) 操作时源目的长度匹配

7.

(1) 用16位除法

1193182 等值16进制为 001234DE H

MOV DX,0012H

MOV AX,34DEH

MOV CX,433

DIV CX

MOV XX,AX

(2) 用32位除法， 无需进行16进制转换

MOV EAX,1193182

MOV EDX,0

MOV ECX,433

DIV ECX

MOV XX,AX

容易出错: (1) 用字节除法但没有进行16进制转换

(2) 被除数和除数单位未统一。

8.

(1) 最简单的方法(**SAL**可改用**SHL**)

SAL AL,4

SAL AX,4

(2) 其他方法(**SAL**可改用**SHL**)

SAL AH,4

SAL AL,4

SHR AL,4

ADD AH,AL (或OR AH,AL)

或

SAL AH,4

AND AL,0FH

ADD AH,AL (或OR AH,AL)

容易出错:

(1) 没有仔细考虑好逻辑计算的过程和结果

9.

利用C位作为排列中的存储单元

```
MOV CX,8  
MOV AL,3 ; 假设AL存放中数为 03H  
CIR: SHR AL,1 ;SHR也可用SAR,RCR,ROR  
RCL BL,1  
LOOP CIR  
MOV AL,BL
```

10. (2) 下面为完整程序，如果只要求写出程序片段，则写出下面兰色部分即可
.486

```
DATA SEGMENT USE16
    BUF  DB 50 DUP(?)
    NOTO DB 50 DUP(?)
DATA ENDS
CODE SEGMENT USE16
    ASSUME CS:CODE,DS:DATA
    BEG: MOV AX,DATA
        MOV DS,AX
        MOV BX,OFFSET BUF
        MOV SI,OFFSET NOTO
        MOV CX,50
    CIR: CMP BYTE PTR [BX],0
        JZ  NEXT
        MOV AL,[BX]
        MOV [SI],AL
        INC SI
    NEXT: INC BX
        LOOP CIR
    EXIT: MOV AH,4CH
        INT 21H
CODE ENDS
    END BEG
```

容易出错:

(1) 不会使用间接寻址方式访问
一个变量的连续单元

(2) PTR的使用问题

(3) 内存单元之间不能直接传送

(4) 判非零数的方法

还可以用

MOV AL, [BX]

OR AL, AL (或 AND AL,AL)

JZ NEXT

.....

(5) 标号不可以使用汇编语言中的
保留字，如寄存器名称等

10。(1) 完整程序

.486

DATA SEGMENT USE16

BUF DB 50 DUP(?)

PLUS DB 50 DUP(?)

MINUS DB 50 DUP(?)

DATA ENDS

CODE SEGMENT USE16

ASSUME CS:CODE,DS:DATA

BEG: MOV AX,DATA

MOV DS,AX

MOV BX,OFFSET BUF

MOV SI,OFFSET PLUS

MOV DI,OFFSET MINUS

MOV CX,50

CIR: MOV AL,[BX]

CMP BYTE PTR [BX],0

JZ NEXT

JG TOPLUS

MOV [DI],AL

INC DI

JMP NEXT

TOPLUS: MOV [SI],AL

INC SI

NEXT: INC BX

LOOP CIR

EXIT: MOV AH,4CH

INT 21H

CODE ENDS

END BEG

注意判断正数和负数的方法

10。(3) 完整程序

.486

DATA SEGMENT USE16

BUF DB 50 DUP(?)

DATA ENDS

CODE SEGMENT USE16

ASSUME CS:CODE,DS:DATA

BEG: MOV AX,DATA

MOV DS,AX

MOV BX,OFFSET BUF

MOV CX,50

CIR: CMP BYTE PTR [BX],0

JGE NEXT

NEG BYTE PTR [BX]

NEXT: INC BX

LOOP CIR

EXIT: MOV AH,4CH

INT 21H

CODE ENDS

END BEG

注意运用**NEG**指令求负数绝对值的方法

10。(4) 完整程序

.486

DATA SEGMENT USE16

BUF DB 50 DUP(?)

DATA ENDS

CODE SEGMENT USE16

ASSUME CS:CODE,DS:DATA

BEG: MOV AX,DATA

MOV DS,AX

MOV BX,OFFSET BUF

BT DWORD PTR [BX],31

JNC EXIT

NEG DWORD PTR [BX]

EXIT: MOV AH,4CH

INT 21H

CODE ENDS

END BEG

注意运用**NEG**指令求负数绝对值的方法

(p.101)

1. (1)

汇编后数据段内容(由左到右，从低地址到高地址)

32H, FFH, FAH, 57H, 9CH, 78H, 34H, 12H, 41H, 00H

注意：负数变成补码，字符变为对应 ASCII码

2.

(1) 指令改正

MOV AX,BNUM -> MOV AX,WORD PTR BNUM

执行后 **AX = FF32H**

MOV BX,BNUM +5 -> MOV BX, WORD PTR BNUM +5

执行后 **BX = 4478H** (其中**44H**高位字节是相邻来自**DNUM**变量定义的数据, 特别注意**LLL**常量不占用内存空间)

MOV CL, DNUM+2 -> MOV CL, BYTE PTR DNUM +2

执行后 **CL = 22H**

(2) **MOV AX, WORD PTR DNUM + 1 (或 DNUM[1])**

(3) **MOV WORD PTR DNUM, BX**

(4) **INC BYTE PTR WNUM , WNUM为 00H,00H**

INC WNUM , WNUM 为 00H,01H (体会由PTR运算符体现出的指令的不同操作意图)

第4次作业 (p.167)

1. (1) 判断x是否在 (5, 24] 区间

假设 X为有符号数, 该程序中X为无符号数时候相应指令
更换JLE -> JNA , JG -> JA

```
MOV FLAG, -1
CMP NUMBER, 5
JLE NEXT
CMP NUMBER, 24
JG NEXT
MOV FLAG, 0
NEXT: ...
```

注意: (1) 复合分支程序设计
(2) 程序运行流程

(p.67)

2.

$1 + \dots + 200 = 20100$ ，结果需要一个字单元存放

SUM DW ?

```
MOV CX,200
MOV AX,0
MOV BX,1
AGA: ADD AX, BX
      INC BX
      LOOP AGA
      MOV SUM, AX
```

注意： 考虑结果存放需要的容量决定寄存器的选用；注意使用正确的指令来完成需要的操作。

二。期中课程小结（主要知识点）

一。数制/码制转换

1、二进制数，十进制数，十六进制数和BCD码数之间的转换方法等。

例： $(210.875)_{10} = (\quad)_2 = (D2.E)_{16}$

$(10010111)_{BCD} = (97)_{10} = (01100001)_2$

例： $(129)_{10} = (10000001)_2 = (81)_{16}$

2、真值数和补码数之间的转换方法等

例：若 $[X]_{补} = 0AAH$ ，则 X 的真值为 $(-86)_{10}$

例：字长=8位，则 $[-6]_{补} = (FA)_{16}$ ，

若 $[X]_{补} = E8H$ ，则 X 的真值为 $(-1E)_{16}$

3、 n 位字长的有符号数、无符号数的数值范围

设机器数字长= n 位，

则 n 位补码数，其真值范围为 $-2^{n-1} \sim +2^{n-1}-1$ 位

无符号数其数值范围为 $0 \sim 2^n-1$

例：16位二进制数所能表示的无符号数范围是 0 ~ 65535。

@复习作业 P.15 - 1(1)-(6), 以及课件范例

二。补码运算和判断溢出以及6个状态标志的变化

用补码形式完成运算。要求有运算过程，写出运算结果、**C, O, A, P, S, Z** 标志的具体值，并判断结果是有溢出（**CPU**角度和程序员角度）

例：字长8位 **(+66) + 9**

42H+09H=4BH O=0,C=0, A=0,P=1, S=0,Z=0

@复习作业 **P.15 - 1(7), P85 - 3** 以及课件范例

三。存储空间中三种地址空间的概念以及IO空间的概念，实模式下由逻辑地址得到物理地址

物理地址计算公式：

物理地址=段基址*16+偏移地址

例：一个有16个字节的数据区，它的起始地址为**3000H: 4567H**，则这个数据区首字节的物理地址是(**34567H**)，最后一个字节的物理地址为(**34576H**)。

@复习作业**P.85 - 1**

四。变量，常量定义以及**PTR**运算符，**SEG,OFFSET**等伪指令的正确使用

- (1) 变量在内存中的存放细节(**DB,DW,DD**)
- (2) 必须使用**PTR**运算符的情形
(牢记上课讲义上的双操作数**3**+单操作数**2** 共**5**种情形)
- (3) 使用**PTR**运算符后在存取内存操作数时的特点

@ 复习课件范例以及作业 **P85.- 6, P101.-1,2**

五。80486的寻址方式

486有3类7种寻址方式

立即寻址方式：获得立即数

寄存器寻址方式：获得寄存器操作数

存储器寻址方式（5种）：获得存储器操作数（内存操作数）

内存寻址方式 16位寻址规定可使用的寄存器

(1) 直接寻址

(2) 寄存器间接寻址 **BX, BP, SI, DI**

段寄存器：[间址寄存器]

某单元的物理地址=段寄存器内容 \times 16+间址寄存器

(3) 基址寻址 **BX, BP**

段寄存器：[基址寄存器+位移量]

物理地址=段寄存器内容 \times 16+基址寄存器+位移量

(4) 变址寻址 **SI, DI**

(5) 基址+变址寻址

例：16位间接寻址寄存器及约定段DS：SI，DI，BX； SS：BP

例：指出下列指令源操作数的寻址方式：

ADD AX,[SI+6]	变址
MOV AX,[BX+SI+6]	基加变
MOV DX,[BX]	间址
MOV DX,BX	寄存器
MOV AL,-1	立即
SUB AX,WORD PTR BUF	直接
ADD DX,[BX+4]	基址
ADD AX,TABLE	注意TABLE是变量还是常量

@复习课件范例以及作业 P.85.-2

六。指令系统

指令格式，功能，指令对F寄存器的影响，D标，I标的作用等

例：**RET**，**IRET**指令，分别从堆栈中弹出多少个字装入什么寄存器。

例：**MOV** 指令的目标寄存器有哪些？

@ 复习课件范例以及 涉及程序编写的作业

七。计算机系统构成及CPU

1. 冯氏结构以及现代计算机硬件组成 的5个部分
2. 三总线结构AB,DB,CB 以及 控制信号 D/C W/R M/IO
3. CPU 内部的 基本结构寄存器，名称以及对应容量，以及功能

八。存储系统

- (1) 存储器分类以及存储器容量的基本单位,
- (2) 存储器容量与引脚（数据线，地址线）的关系,
- (3) 存储器容量的扩展方式（字扩展和位扩展，掌握概念 **P.206**）

@复习课件范例

九。程序设计和程序阅读

- (1) 伪指令在程序中的作用和结果 (**DB,DW,DD**变量定义; 常量定义; 宏指令 (使用以及与子程序的区分))
- (2) 完整的源程序的结构 (包含伪指令以及语句的使用)
- (3) 常见程序片段的设计方法 (分支, 循环, 代码转换等)
例: 二进制数, 十进制数的显示, 分支, 循环程序设计等。
- (4) 在程序设计中使子程序
- (5) 功能调用(**BIOS/DOS**)在设计程序输入输出时的应用

举例程序阅读

.486

DATA SEGMENT USE16

FIRST DB 12H,34H

SECOND DB 56H,78H

DATA ENDS

CODE SEGMENT USE16

ASSUME CS:CODE,DS:DATA

BEG:MOV AX,DATA

MOV DS,AX

PUSH WORD PTR FIRST

PUSH WORD PTR SECOND

POP WORD PTR FIRST

POP WORD PTR SECOND

MOV AH,4CH

INT 21H

CODE ENDS

END BEG

该程序完成的功能是_**FIRST**和**SECOND**两个变量内容交换_

例：设BX=1200H，SI=0002H，DS=3000H，(31200H)=50H，(31201H)=02H，(31202H)=0F7H，(31203H)=90H。请写出下列各条指令单独执行后，有关寄存器及存储单元的内容。

1. DEC BYTE PTR[SI+1200H] 指令执行后
(31202H)=__F6__H，操作数的寻址方式为__变址__。

2. MOV AX, WORD PTR [BX+SI] 指令执行后，AX=__90F6__H，源操作数的寻址方式为__基加变__。

设在数据段有如下定义：

BUF1 DB 55H, 66H, 'ABCD'

BUF2 DW 11H

BUF3 DB 2 DUP(?)

写出CPU执行代码段的下列指令后的结果：

MOV AX, DATA

MOV DS, AX

MOV SI, OFFSET BUF1

MOV BL, BUF1

； **BL= 55 H**，源操作数为 直接 寻址方式

ADD BL, BYTE PTR[SI+1]

； **BL= BB H**，目标操作数为 寄存器 寻址方式

MOV BYTE PTR BUF2+1, BL

； **BUF2= 11 H**，**BUF2+1= BB H**。

@复习课件范例以及作业中涉及编程的部分

十。汇编程序的完整开发过程

实验操作步骤： 编辑、编译、链接；每个步骤输出的文件是什么。