

汇编语言程序设计期末复习和习题课

2021年10月

重要知识点和教材章节对应关系

第1章 1.1.2, 1.2节中的1.2.1, 1.2.2, 1.2.3

第2章 2.3节中的2.3.3, 2.4节中的2.4.1, 2.4.2

第3章 3.1节, 3.2节, 3.3节, 3.4节, 3.5节

(要求的汇编指令: MOV, LEA, XCHG, PUSH, POP, PUSH, POPA, ADD, ADC, SUB, SBB, INC, DEC, NEG, MUL, IMUL, DIV, IDIV, CMP, DAA, JMP, JC/JNC, JZ/JNZ, JS/JNS, JP/JNP, JO/JNO, JA/JNA, JC/JNC, JG/JGE, JL/JLE, LOOP, CALL, RET, INT, IRET, NOT, AND, OR, XOR, SAL, SAR, SHL, SHR, RCL, RCR, ROL, ROR, TEST, MOVSB(W/D), LOADS B(W/D), STOSB(W/D), CMPSB(W/D), SCASB(W/D), REP, REPE, REPNE, CLC, STC, CLD, STD) (要求的伪指令: SEG, OFFSET, \$, [], PTR, DB, DW, DD, EQU, =)

第4章 4.1节中4.1.1, 4.2节 4.2.1, 4.3节(DOS 功能调用要求 01H,02H,09H,0AH, 4CH 号功能调用; BIOS功能调用要求 int16H 00号功能调用, int10H 0E号功能调用), , 4.5节(对应课件上程序), 4.6节(对应课件上程序), 4.7节中4.7.1 (对应课件上程序), 4.8节中4.8.1(对应课件上程序)

第6章(对应课件掌握: 存储器分类, 地址线与存储器容量的关系, 结合课件理解读写存储器的过程,)

一、数制和码制

1、掌握二进制数，十进制数，十六进制数和BCD码数之间的转换方法，会比较数值大小。记住常用字符0-9，A~F，回车符和换行符的ASCII码。（参见复习 教材第1章课后作业 P.21 第4题）

$$(1) (11101.1011)_2 = (29.6875)_{10}$$

$$(2) (147)_{10} = (10010011)_2 = (93)_{16}$$

$$(3) (3AC)_{16} = (940)_{10}$$

$$(4) (10010110)_{BCD} = (1100000)_2$$

2、真值和补码数之间的相互转换方法（参见复习 教材第1章课后作业

P.21 第4题）

字长=8位, $[-1]_{\text{补}} = (\text{FF})_{16}$

$[X]_{\text{补}} = (\text{A5})_{16}$, 则 $X = (-5\text{B})_{16}$

字长=8位, $X = (8\text{E})_{16}$, 当X为原码时, X的真值为 $(-\text{E})_{16}$

字长=8位, $X = (8\text{E})_{16}$, 当X为反码时, X的真值为 $(-71)_{16}$

字长=8位, $X = (8\text{E})_{16}$, 当X为补码时, X的真值为 $(-72)_{16}$

字长=8位, $X = (8\text{E})_{16}$, 当X为无符号数时, X的真值为 $(8\text{E})_{16}$

3、n位字长的有符号数、无符号数的数值范围（如字长=8或16）

设机器数字长=n位, 则n位补码数, 其真值范围为 $-2^{n-1} \sim +2^{n-1}-1$

无符号数其数值范围为 $0 \sim 2^n-1$

容易出错：（1）真值数与原码的差别

二、补码加法计算，判定状态标志以及判断溢出。（参见复习 教材第1章课后作业P.21第4题(7)，第3章课后作业 P.135 第3题）

(7)设字长=8位，用补码形式完成下列十进制数运算。要求有运算过程并讨论结果是否有溢出？

(a) $(+75) + (-6)$

$$(+75)_{\text{补}} = (0100\ 1011)_2, \quad (-6)_{\text{补}} = (1111\ 1010)_2$$

$$\begin{array}{r} 0100\ 1011 \\ + 1111\ 1010 \\ \hline (1)\ 0100\ 0101 \end{array}$$

$(01000101)_2$ 的10进制真值 = +69

$$(+75) + (-6) = +69$$

$$\begin{aligned} OF &= X_S \cdot Y_S \cdot \overline{Z_S} + \overline{X_S} \cdot \overline{Y_S} \cdot Z_S = 0 \cdot 1 \cdot \overline{0} + \overline{0} \cdot \overline{1} \cdot 0 = 0 \cdot 1 \cdot 1 + 1 \cdot 0 \cdot 0 \\ &= 0 + 0 = 0 \end{aligned}$$

无溢出

容易出错：（1）CPU判断溢出（应使用公式计算）

$$OF = X_s \cdot Y_s \cdot \overline{Z_s} + \overline{X_s} \cdot \overline{Y_s} \cdot Z_s$$

（2）需要在运算前将参加运算的数写成补码形式。得出结果后按照要求写出对应形式，例如10进制真值数

二、补码加法计算，判定状态标志以及判断溢出。（参见复习 教材第1章课后作业P.21(7)，第3章课后作业 P.135 第3题）

3. 以 2^{16} 为模，将C678H分别和下列各数相加，列表写出十六进制和数，以及A、C、O、P、S、Z六种状态标志的值。

(1) CF23H (2) 6398H (3) 94FBH (4) 65E2H

C678H+ CF23H

```
  1100011001111000
+ 1100111100100011
-----
(1)1001 0101 1001 1011
```

计算结果：C678H+ CF23H =**959BH**

六个状态标志：A=0, C=1, O=0, P=0, S=1, Z=0

程序员判断溢出：

假设加数和被加数均为有符号数，判断O位, O=0,没有溢出。

假设加数和被加数均为无符号数，判断C位, C=1,有溢出。

二、补码加法计算，判定状态标志以及判断溢出。（参见复习 教材第1章课后作业P.21(7)，第3章课后作业 P.135 第3题）

容易出错：（1）CPU判断溢出（应使用公式计算）

$$OF = X_s \cdot Y_s \cdot \overline{Z_s} + \overline{X_s} \cdot \overline{Y_s} \cdot Z_s$$

（2）程序员判溢依据有符号数和无符号数的前提条件，判定方法不同。

（3）P, A标志均只针对计算中最低位字节

三、实地址模式下，一个逻辑段的体积是64KB，物理地址的形成
(要求会计算)

物理地址计算公式： 物理地址=段基址*16+偏移地址

(参见复习 教材第3章课后作业 P.135 第1题)

(1) 逻辑地址 1234H:5678H 对应的物理地址 (179B8H)

$$\begin{array}{r} 12340H \\ + 5678H \\ \hline 179B8H \end{array}$$

容易出错： (1) 16进制加法计算

四、80486的寻址方式和指令（常用的伪指令、常用的运算符和80486基本指令集）

486有3类7种寻址方式

1. 要求会判断操作数的寻址方式，对于内存操作数，会判断寻址的逻辑段

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（直接寻址，间接寻址，基址寻址，变址寻址，基址加变址寻址）：获得存储器操作数（内存操作数）

四、80486的寻址方式和指令（常用的伪指令、常用的运算符和80486基本指令集）

例：指出下列指令源、目操作数的寻址方式：

ADD AX, TABLE

MOV AX, [BX+SI+6]

MOV AL, [BX +6]

MOV AL, 0F0H

MOV DX, [BX]

MOV AX, BX

2. 要求会识别书写错误的指令（助记符写错，操作数不符合要求，寻址方式写错，PTR运算符需要使用时没有使用或用错等）（参见复习 教材第3章课后作业 P.135 第2题）

四、80486的寻址方式和指令（常用的伪指令、常用的运算符和80486基本指令集）

3. 掌握单操作数指令和双操作数指令中PTR运算符的使用规则
4. 掌握数据定义伪指令 DB , DW , DD 正确的书写格式， 定义的数据在内存中的存放方法（参见复习 教材第3章课后作业 P.135 第6题）
6. 掌握符号定义伪指令 EQU和 = 的含义
7. 掌握 \$ 运算符 在计算字符串长度中的应用
8. 掌握 SEG ， OFFSET 运算符在提取 段基址和有效地址 时的作用

教材第3章课后作业

6. 下面程序执行后，AX是多少？

设数据段中有如下定义：

TABLE DW 158,258,358,458

ENTRY DW 3

代码段中有如下指令：

MOV BX, OFFSET TABLE

MOV SI, ENTRY

MOV AX, [BX+SI]

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

五、能够书写指令完成处理任务；能读懂和书写简单的含有分支循环以及访问内存单元数据的程序；能掌握一个完整程序的各个组成部分的书写方法。（参见教材第3章 P.135 第4题，教材第3章课后作业 P.135 第5题，第8题；第4章课后作业 P.194 第1题 第2题，同时复习课件上对应章节课堂上讲解过的程序例子）

能够读懂以及编写利用功能调用完成字符/字符输入以及显示；能读懂以及编写简单地调用子程序的主程序以及对应子程序；能掌握宏调用和子程序调用的不同，能够读懂以及编写代码转换中以二进制形式显示内存单元或寄存器中数据；以十六进制形式显示内存单元或寄存器中的数据，以10进制显示显示内存单元或寄存器中的数据）

（参见复习 教材第4章4.8节 p.178 程序例4.15, p.181. 例4.18, 4.19，同时复习课件上对应章节课堂上讲解过的程序例子）

教材第3章课后作业

4.

(1) 用一条指令完成: AH 高4位取反, 低4位不变

XOR AH,0F0H

容易出错:

(1) AND, OR与XOR指令的具体作用.

(2) 0F0H写成F0H

4.

**(2) 用一条指令完成: BH 高4位取反, 低4位不变;
BL高4位不变, 低4位取反**

XOR BX,0F00FH

(3) 用一条指令完成: CX低4位清零, 其他位不变

AND CX, 0FFF0H

8. 把AH低四位和AL低四位拼装成一个字节（AH低四位为拼装后的高四位）→ AH。

SAL AL, 4

SAL AX, 4

5. 已知数据段有：

FIRST DB 12H, 34H

SECOND DB 56H, 78H

(1) 要求采用传送指令编写一段程序，实现 **FIRST**和 **SECOND**单元的内容互换，**FIRST+1**单元和 **SECOND+1**单元的内容互换。

(2) 设(SS) = 2000H，(SP) = 3456H，用堆栈指令编写一段程序完成上述要求，并画出堆栈的数据变化示意图。

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**

初始状态：堆栈段 (SS)=2000H, 堆栈指针初值 (SP)=3456H

FIRS ~	12H
	34H
SECON ~	56H
	78H

FIRS ~	12H
	34H
SECON ~	56H
	78H

23456H	XXH	← SP
23457H	XXH	

(1) 执行 PUSH WORD PTR FIRST

23454H	12H	← SP
23455H	34H	
23456H	XXH	
23457H	XXH	

(2) 执行 PUSH WORD PTR FIRST

FIRST	12H
	34H
SECON	56H
	78H

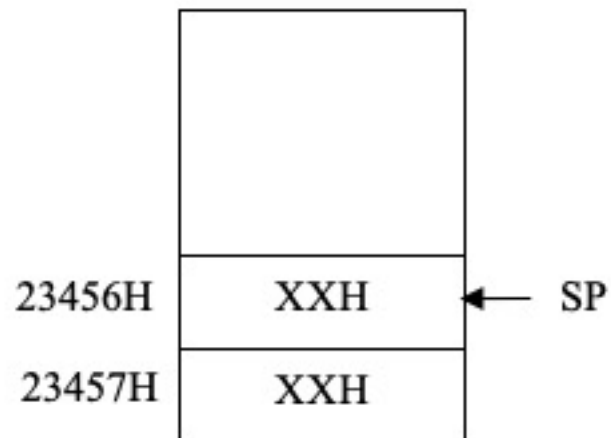
FIRS	56H
	78H
SECON	56H
	78H

23452H	56H	← SP
23453H	78H	
23454H	12H	
23455H	34H	
23456H	XXH	
23457H	XXH	

23454H	12H	← SP
23455H	34H	
23456H	XXH	
23457H	XXH	

(3) 执行 PUSH WORD PTR SECOND (4) 执行 POP WORD PTR FIRST

FIRST	56H
	78H
SECON	12H
	34H



(5) 执行 POP WORD PTR SECOND

10. 设数据段有：

BUF DB 50 DUP(?) ;50个有符号数

分别编写四个程序段：

（1）将BUF中的正数送数据段PLUS开始的若干单元，负数送数据段MINUS开始的若干单元。

（2）将BUF中的非零的数送数据段NOT0开始的若干单元。

（3）分别求出 BUF中各个数的绝对值。

（4）设BUF~BUF + 3 四个单元中存放的是一个双字型有符号数（BUF单元为最低位字节），求出其绝对值

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

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) 注意判断正数和负数的方法

(4) 正确使用转移指令

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) 判非零数的方法

还可以用

MOV AL, [BX]

OR AL, AL (或 AND AL,AL)

JZ NEXT

.....

(3) 用 NEG求绝对值的方法

MOV BX, OFFSET BUF

MOV CX, 50

CIR: CMP BYTE PTR [BX], 0

JGE NEXT

NEG BYTE PTR [BX]

NEXT: INC BX

LOOP CIR

教材第4章课后习题 (p.194)

1. (1) 1. 数据段NUMBER单元有一个数X,判断 $5 < X \leq 24$? 若是置FLAG单元为0,否则置FLAG单元为-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.194)

2. 计算从1开始连续50个偶数之和，并将结果存放在SUM字单元中。

$2+4+6\ldots+100 = 2550$ ，结果需要一个字单元存放

SUM DW ?

```
MOV CX,50
MOV AX,0
MOV BX,2
AGA: ADD AX, BX
      INC 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。

六。计算机系统构成及**CPU**

1. 冯诺依曼结构的计算机硬件组成的**5**个部分（运算器，控制器，存储器，输入设备，输出设备）
2. 三总线**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

七。存储器

- (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) **CPU**通过三总线读写存储器存储单元的过程（见课件）

八。实验

- (1) 掌握汇编语言程序开发的过程中3个步骤。每个步骤对应的文件名称
- (2) 掌握实验中编译错误的判断和解决方法（主要问题：指令书写格式，寻址方式格式，完整程序格式，子程序格式，PTR运算符等）
- (3) 了解实验中运行错误的调试方法