汇编语言各类指令复习资料

1. “jmp 段地址：偏移地址” ：
用指令给出的段地址修改 CS，偏移地址修改 IP。
2. “jmp 寄存器 ”:
用寄存器的值修改 IP。
例如 jmp ax 等同于 mov IP , ax
3. push ax:
(1) sp=sp-2 (2)将 ax 中的内容送入 SS:SP 指向的内存单元处
4. pop ax:
(1) 将 SS:SP 指向的内存单元处的数据送入 ax 中；
(2) SP=SP+2
5. mov ax,[bp] 含义：(ax)=((ss)*16+（bp）)
6. dup 的使用格式：
db 重复的次数 dup （重复的字节型数据）
dw 重复的次数 dup （重复的字型数据）
dd 重复的次数 dup （重复的双字型数据）
Db 3 dup(0,1,2)定义了 9 个字节，相当于 db 0,1,2,0,1,2,0,1,2
7. div 指令：被除数默认存放在 AX 或 DX、AX 中，如果除数为 8 位，则被除数为 16 位，默认存放在 AX 中，结果商存在 al，余数存在 ah；如果除数为 16 位，则被除数为 32 位，默认存放在 DX 和 AX 中，DX 存放高 19 位，AX 存放低 16 位，结果商存在 AX，余数存在 DX。
8. offset 标号名称：获取标号的偏移地址。
9. dd 伪指令：定义双字类型。
10. dw 伪指令：定义单字类型。
11. db 伪指令：定义字节类型。
12. jcxz 标号：如果(cx)=0，则转移到标号处执行。
13. loop 标号:先（cx）--,再 如果（cx）≠0，则转移到标号处执行。
14. ret 指令：pop IP;
15. retf 指令：pop IP； pop CS.
16. call 标号：push IP;jmp near ptr 标号。注意这个 IP 指的是 call 物理上相邻的下一条指令的偏移地址。
17. Call far ptr 标号： push CS ; push IP ; jmp far ptr 标号。
18. Call word ptr 内存单元地址：push IP; jmp word ptr 内存单元地址。
19. Call dword ptr 内存单元地址：push CS;push IP; jmp dword ptr 内存单元地址。
20. mul 指令：两个相乘的数位数必须相同，如果是 8 位，一个默认存放在 al 中，另一个存放在 8 位 reg 或内存单元中，结果默认存放在 AX 中；如果是 16 位，一个默认存放在 AX 中，另一个存放在 16 位 reg 或内存单元中，结果默认存放在 AX（存放低位）,DX（存放高位）。
21. mul byte ptr ds:[0]或 mul word ptr ds:[0]。
22. ZF 标志：判断结果是否为 0，若为 0，则 zf=1;
23. PF 标志：奇偶标志位，结果的所有 bit 种 1 的个数是否为偶数，若是则 pf=1;
24. SF 标志：符号标志位，判断结果是否为负，若是，则 sf=1;将数据当作有符号数来运算时，可以通过它的值判断结果的正负，当作无符号数时，结果没有任何意义。
25. CF 标志：进位标识符，在进行无符号数运算的时候，它记录运算结果的最高有效位向

更高位的进位值，或从更高位的借位值。

26. OF 标志：符号溢出位，在进行有符号数运算的时候，如结果超出了机器所能表示的范围，则称溢出。

27. CF 和 OF 所表示的进位与溢位，是分别对无符号数和有符号数运算而言的，他们之间没有任何关系。

28. adc 指令：adc ax,bx → (ax) = (ax)+(bx)+CF

29. sbb 指令：sub ax,bx → (ax) = (ax)-(bx) - CF

30. cmp 指令：相当于减法指令，只是不保存结果，对标志位产生影响。

cmp ax,bx 的逻辑含义是比较两者的值：

zf=1,说明 (ax)=(bx) zf=0,说明 (ax)≠(bx)

cf=1,说明 (ax)<(bx) cf=0,说明 (ax)≥(bx)

cf=0 且 zf=0,说明 (ax)>(bx)

cf=1 或 zf=1,说明 (ax)≤(bx)

（1）sf=1,of=0 → （ax）< （bx）

（2）sf=1,of=1 → （ax）> （bx） 因为溢出导致实际结果为负，逻辑上真正的结果为正

（3）sf=0,of=0 → （ax）≥（bx）

（4）sf=0,of=1→ （ax）< （bx） 因为溢出导致实际结果为正，逻辑上真正的结果为负

31. 指令 含义 检测标志位 通常和 cmp 指令同用

je:	等于则转移	zf=1	
jne	不等于则转移	zf=0	
jb	低于则转移	cf=1	
jnb	不低于则转移	cf=0	
ja	高于则转移	cf=0 且 zf=0	
jna	不高于则转移	cf=1 且 zf=1	

32. DF 指令：方向标志位，在串处理指令中，控制每次操作后 si 和 di 的增减。

df=0 每次操作后 si,di 递增

df=1 每次操作后 si,di 递减

33. movsb 指令：

相当于执行：((es)*16+(di))=((ds)*16+(si)),如果 df=0,则(si)=(si)+1,(di)=(di)+1，反之。

34. movsw 指令：

相当于执行：((es)*16+(di))=((ds)*16+(si)),如果 df=0,则(si)=(si)+2,(di)=(di)+2，反之。

35. 一般来说 movsb 和 movsw 和 rep 使用。

rep movsb 相当于

s:movsb

loop s

rep 的作用是根据 cx 的值，重复执行后面的串传送指令。

36. cld 指令：将 DF 的值设为 0

37. std 指令：将 DF 的值设为 1

38. pushf 指令：将标志寄存器的值压栈

39. popf 指令：从栈中弹出数据送入标志寄存器中。

40. iret 指令功能：

pop IP

pop CS

popf

41. shl 指令：逻辑左移指令
将一个寄存器或内存单元中的数据向左移位
将最后移除的一位写入 CF 中
最低为用 0 补充。
42. shr 指令：逻辑右移指令
将一个寄存器或内存单元中的数据向右移位
将最后移除的一位写入 CF 中
最高为用 0 补充。
shr al,1;　右移一位
mov cl,3
shr al,cl;　右移位数大于 1 时，必须将移动位数放在 cl 中。
43. sti,设置 IF=1；　接收可屏蔽中断
　　cli,设置 IF=0.　屏蔽可屏蔽中断
44. seg 操作符：取某一标号的段地址。
45. 数值+30h=对应字符的 ASCII 值。
46. a db 1,2,3　标号 a 不但表示内存单元的地址，还表示了内存单元的长度，即表示在此标号处的单元是一个字节单元还是字单元还是双字单元。
47. a :db 1，2，3 标号 a 仅代表内存单元的地址。