

第三章 单片机的汇编语言与程序设计习题

1. 设内部 RAM 中 59H 单元的内容为 50H, 写出当执行以下程序段后寄存器 A, R0 和内部 RAM 中 50H, 51H 单元的内容为何值?

```
MOV    A, 59H
MOV    R0, A
MOV    A, #00H
MOV    @R0, A
MOV    A, #25H
MOV    51H, A
MOV    52H, #70H
```

解: MOV A, 59H ; A=50H
MOV R0, A ; R0=50H
MOV A, #00H ; A=00H
MOV @R0, A ; 50H=00H
MOV A, #25H ; A=25H
MOV 51H, A ; 51H=25H
MOV 52H, #70H ; 52H=70H

所以: A=25H R0=50H ; 50H=00H 51H=25H

2. 请选用合适的指令对 P0 口内容做修改 ()。

解: MOV A, P0
ANL A, 0fh
Mov P0, A

3. 试问外部数据存储器和程序存储器可以用哪些指令来实现? 举例说明。

解：访问外部数据存储器指令有：

```
MOVX @DPTR, A    MOVX DPTR, #0100H    MOV @DPTR, A
MOVX A, @DPTR    MOVX DPTR, #0200H    MOV A, @DPTR
MOVX A, @Ri      MOVX A, @R0
MOVX @Ri, A      MOVX @R1, A
```

访问程序存储器指令有：

```
MOVX A, @A+PC
MOVX A, @A+DPTR
```

4. 设堆栈指针 SP 中的内容为 60H，内部 RAM 中 30H 和 31H 单元的内容分别为 24H 和 10H，执行以下程序段后，61H，62H，30H，31H，DPTR 及 SP 中的内容将有何变化？

```
PUSH    30H
PUSH    31H
POP     DPL
POP     DPH
MOV     30H, #00H
MOV     31H, #0FFH
```

```
解: PUSH    30H           ; 61H=24H
      PUSH    31H           ; 62=10H   SP=62H
      POP     DPL           ; DPL=10H
      POP     DPH           ; DPH=24H   SP=60H
      MOV     30H, #00H     ; 30H=00H
      MOV     31H, #0FFH    ; 31H=0FFH
```

5. 设(A)=40H (R1)=23H (40H)=05H 执行以下两条指令后，累加器 A 和 R1 以

及内部 RAM 中 40H 单元的内容各位何值？

```
XCH  A, R1
```

```
XCHD A, @R1
```

解： XCH A, R1 ; A=23H R1=40H

```
XCHD  A, @R1      ; A=25H  40H=03H
```

6. 两个四位 BCD 码数相加，被加数和加数分别存于 50H, 51H 和 52H, 53H 单元中（次序为千位、百位在低地址中），和数存在 54H, 55H 和 56H 中（56H 用来存放最高位的进位，试编写加法程序）

解： ORG 0000H

```
LJMP  START
```

```
ORG  0100H
```

```
START: MOV  A, 53H
```

```
ADD  A, 51H
```

```
DA  A
```

```
MOV  55H, A
```

```
MOV  A, 52H
```

```
ADDC A, 50H
```

```
DA  A
```

```
MOV  56H, C
```

```
SJMP $
```

```
END
```

{A} = 01010101B, {R5} = 10101010B, 分别写出执行 ANL A, R5; ORL A, R5; XRL A, R5 指令后结果。

解： ANL A, R5 ; 00000000B

ORL A,R5 ; 11111111B

XRL A,R5 ; 11111111

8. 设指令 SJMPrel=7EH, 并假设该指令存放在 2114H 和 2115H 单元中。当该指令执行后, 程序将跳转到何地址?

解: $2116H + 60H = 2176H$

9. 简述转移指令 AJMP addr11, SJMPrel, LJMPaddr16 及 JMP @A+DRTR 的应用场合。

解: AJMP addr11 为 2K 字节范围内的无条件转跳指令, 把程序的执行转移到指定的地址。

SJMP rel 是无条件转跳指令, 执行时在 PC 加 2 后, 把指令中补码形式的偏移量加到 PC 上, 并计算出转向目标地址。转向的目标地址可以在这条指令前 128 字节到后 127 字节之间

LJMP addr16 执行这条指令时把指令的第二字节和第三字节分别装入 PC 的高位和地位字节中无条件地转向指定地址。转移目标地址可以在 64K 程序存储器地址空间的任何地方, 不影响任何标志。

JMP @A+DRTR 指令的功能是把累加器中 8 位无符号数与数据指针 DRTR 中的 16 位数相加, 将结果作为下条指令地址送入 PC, 利用这条指令能实现程序的散转。

10. 试分析以下程序段, 当程序执行后, 位地址 00H, 01H 中的内容将为何值? P1 口的 8 条 I/O 线为何状态?

```
CLR    C
MOV    A,#66H
JC     LOOP1
CPL    C
SETB   01H
```

```

        JB      ACC.2,LOOP2

        CLR     00H

LOOP2:  MOV     P1,A

解:    CLR     C           ;CY=0

        MOV     A,#66H     ;A=66H

        JC      LOOP1

        CPL     C           ;CY=1

        SETB    01H       ;20H.1=1

LOOP:   ORL     C,ACC.0    ;CY=1

        ,LOOP2 ;

        CLR     00H

LOOP2:  MOV     P1,A       ;P1=01100110B

        SJMP    $

        20H.0=0  20H.1=1  P1=66H

```

11.的特查指令表，写出以下两条指令的机器码，并比较一下机器码中操作数排列次序点。

```
MOV 78H,80H
```

```
MOV 78H,#80H
```

解：直接寻址单元传送到直接寻址单元的机器码是第二个操作数在前，而立即数传送到直接地址单元是第一个操作数在前，次序正好相反。

12.手工汇编以下程序段

```
ORG 873BH
```

```

AAA      EQU      851AH
QQQ      MOV      A,35H
          CLR      C
          SUBB    A,#0AH
          JC      QQ16
          MOV      A,36H
          SUBB    A,#0AH
          JC      QQ15
          AJMP    AAA
QQ15:    MOV      35H,#00H
QQ16:    JNB     02H,QQ17
          MOV      R6,39H
          DEC     R6
          SJMP    QQ18
QQ17:    MOV      R6,39H
          INC     39H
QQ18:    CLR      05H
          LJMP    8500H
          END

```

13. 假设有两个无符号数 x, y 分别存放于内部存储器 50H, 51H 单元中, 试编写一个程序实现 $x*10+y$, 结果存入 52H, 53H 两个单元中。

```

解:      ORG     0000H
          SJMP    START
          ORG     0030H

```

```

START: MOV  A,50H

      MOV  B,#10

      MUL  AB      ;(50H)×10 积的高、低字节分别在 B、A 中
      ADD  A,51H

      MOV  53H,A  ;积的低字节加(51H) 其和存放在 53H 中
      MOV  A,#00H

      ADDC A,B

      MOV  52H,A  ;积的高字节加进位位存放在 52H 中。

      SJMP $

      END

```

14 从内部存储器 20H 单元开始，有 30 个数据。试编一个程序，把其中的正数，负数分别送 51H 和 71H 开始的存储单元，并分别记下正数负数的个数送 50H 和 70H 单元。

```

解：   ORG  0000H

      SJMP  START

      ORG  0030H

START: MOV  1EH,#51H      ;正数存放首地址 51H 存于 1EH 单元

      MOV  1FH,#71H      ;负数存放首地址 71H 存于 1EH 单元

      MOV  R0,#20H      ;建立取数（源操作数）的地址指针

      MOV  R2,#30      ;预置数据长度

      MOV  50H, #00H    ;正数个数统计单元清零

      MOV  70H,#00H    ;负数个数统计单元清零

LOOP: MOV  A,@R0      ;取数

      JB   ACC.7,NEG    ;是负数转 NEG 处理

POST: MOV  R1,1EH      ;是正数，将暂存的地址送 R1（间址寄存器）

      MOV  @R1,A      ;将正数入存

      INC  50H        ;正数个数加 1

```

```

        INC    1EH          ;正数暂存地址加 1 修正
LOOP1: INC    R0           ;取数地址加 1 修正

        DJNZ   R2,LOOP    ;计数长度减 1, 不等于零, 继续循环统计

        SJMP   $           ;结束

NEG: MOV    R1,1FH        ;是负数, 将暂存的地址送 R1 (间址寄存器)

        MOV    @R1,A      ;将正数入存

        INC    70H        ;负数个数加 1

        INC    1FH        ;负数暂存地址加 1 修正

        SJMP   LOOP1     ;转取数地址修正

```

15 内部存储单元 40H 中有一个 ASCII 码字符, 试编一程序, 给该数的最高位加上奇检验。

```

解:     ORG    0000H

        SJMP   START

        ORG    0030H

START: MOV    A,40H      ;取数给 A

        CLR    ACC.7    ;A 最高位 (奇偶校验位) 清零

        JB     P, LOOP   ;40H 中的 ASCII 码原来就是奇数个 1 (最高位给零)

        SETB                   ;40H 中的 ASCII 码原来是偶数个 1, 最高位置 1

LOOP: MOV    40H,A      ;入存

        SJMP   $         ;结束

        END

```

16. 编写一段程序, 将存放在自 DATA 单元开始的一个四字节数 (高位在前) 取补后送回原单元。

解: 假设 DATA 在内部数据存储器中 (假设存放在 30H 起始的 4 个单元中); 采

用求反加 1 的算法；

```
ORG    000H

SJMP   START

ORG    0030H

START: SETB  C           ; 置进位位为 1

      MOV   R2, #04H     ; 预置计数长度

      MOV   R0, #DATA1+3 ; 取数指针指向低字节的地址

LOOP:  MOV   A, @R0      ; 取数

      CPL   A           ; 求反

      ADDC  A, #00H     ; 加进位

      MOV   @R0, A      ; 入存

      DEC   R0          ; 地址指针修正

      DJNZ  R2, LOOP    ; 4 字节未处理完, 继续循环处理

      SJMP  $           ; 结束

DATA1  EQU  30H;

END
```

17. 以 BUF1 为起始地址的外存储区中, 存放有 16 个单字节无符号二进制数, 是编一程序, 求其平均值并送 BUF2 单元。

解: 设 R2、R3 存放和, 将其除以 16 (R2、R3 联合左移 4 位) 即为平均置值 (单字节存放)

```
ORG    0000H

SJMP   START

ORG    0030H

START: MOV   R2, #00H    ;和高字节清零

      MOV   R3, #00H    ;和低字节清零
```

```

MOV DPTR #DATA1 ;建立外部数据存储器的地址指针

MOV R4, #10H ;预置计数长度

LOOP: MOVX A, @DPTR ;从外部数据存储器取数

ADD A, R3

MOV R3, A

MOV A, R2;

ADDC A, #00H;

MOV R2, A ;累加

INC DPTR ;地址指针修正

DJNZ R4, LOOP ;16字节未累加完, 继续循环累加

MOV R4, #04H ;R2、R3联合移位4次(除以16)

LOOP1: CLR C

MOV A, R2

RRC A

MOV R3, A

DJNZ R4, LOOP1 ;平均值存放于R3中

SJMP $ ;结束

DATA1 EQU 1000H;

END

```

18. 在 DATA1 单元中有一个带符号 8 位二进制数 x 。编一程序, 按以下关系计算 y 值, 送 DATA2 单元。

$$y = x + 5, \quad x > 0$$

$$y = x, \quad x = 0$$

$$y = x - 5, \quad x < 0$$

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/258076017110006046>