

MCS-51 单片机 实验教程

刘胜利 毛建国

南京航空航天大学
能源与动力学院

2006年2月

目 录

实验一 顺序与循环程序设计.....	1
1. 1 无符号十进制数加法实验.....	1
1. 2 数据传送实验.....	3
实验二 分支和与循环程序.....	4
2. 1 统计正整数个数实验.....	4
2. 2 统计学生成绩实验.....	5
实验三 堆栈和子程序.....	7
3. 1 十六制数转换成ASCII码.....	7
3. 2 二进制数转换成压缩BCD码.....	9
实验四 80C51 综合软件设计.....	10
实验五 P1 口亮灯实验.....	13
实验六 可编程并行口 8255A 实验.....	15
6. 1 多路开关控制实验.....	15
6. 2 舞台彩灯控制器.....	18
实验七 定时器与计数器实验.....	20
7. 1 航标实验.....	20
7. 2 定时器中断实验.....	23
7. 3 电子时钟.....	24
实验八 A/D 转换实验.....	29
实验九 D/A 转换实验.....	34
实验十 80C51 的串行口实验.....	37

实验十一	键盘与显示实验	39
实验十二	数据采集 (设计性实验)	42
附录一	DES—51 开发系统常用菜单命令	46
附录二	DES-51常用芯片	49
附录三	80C51指令表	52

实验一 顺序与循环程序设计

1.1 无符号十进制数加法实验

一、实验目的：

通过此加法程序实验，熟悉用于实验的仿真器开发系统装置，掌握仿真器开发系统的使用方法，并掌握 MCS-51 汇编语言程序设计方法和调试方法。

二、实验内容：

无符号多字节十进制加法。加法的首地址由 R0 指出，被加数和结果的存储单元首地址由 R1 指出，字节数由 R2 指出。

三、步骤：

- 1) 向仿真器开发系统输入程序
- 2) 试求 $362418+382615=?$

加数地址	30H	31H	32H	R0 ← 加数内容
内容	18H	24H	36H	
被加数地址	40H	41H	42H	R1 ← 被加数内容
内容	15H	26H	38H	
结果地址	40H	41H	42H	通过 A 计算结果
内容	33H	50H	74H	

四、参考程序如下：

```
RESET: AJMP MAIN      ; 转移到主程序
        ORG    0050H    ; 主程序开始地址
MAIN:   MOV    R2, #3   ; R2 为计数器，表示有几个字节相加
        MOV    R0, #30H ; 被加数首地址
        MOV    R1, #40H ; 加数首地址
        CLR    C        ; 进位清 0
NEXT1: MOV    A, @R0    ; 被加数的内容放入 A 寄存器中
        ADDC   A, @R1   ; 两个字节求和即 A ← @R0+@R1+C
        DA    A        ; 进行十进制调整
        MOV    @R1,A    ; 将其结果存入 R1
        INC    R0       ; R0+1 被加数地址+1
        INC    R1       ; R1+1 加数地址+1
```

```
DJNZ R2,NEXT1 ; 计数器-1, 若不等于 0 , 继续循环相加
SJMP $ ; 计算完在此等待
END ; 程序结束
```

五、实验要求:

- 1) 充分理解程序, 观察 DA A 指令执行前后 A 的变化, 从而掌握 DA A 指令作用。
- 2) 改变其被加数与加数的内容, 再观察运行结果。

1.2 数据传送实验

一、实验目的：

进一步熟悉 MCS-51 汇编语言程序设计，掌握 MCS-51 循环设计方法。
掌握 8031 内部 RAM 和外部 RAM 的数据操作，掌握这两部分 RAM 存储的特点与应用。

二、实验内容：

将内部 RAM 40H~4FH 中的 16 个单字节数据传送到外部 RAM 1000H~100FH 单元中。

三、参考程序如下：

```
ORG 0000H
RESET: AJMP MAIN
ORG 0050H
MAIN: MOV R0, #40H      ; RAM 40H 为首地址
      MOV R2, #10H     ; 计数器 R2 为 16 个单字节

      MOV A, #0A0H    ; 把立即数 A0H 送给 40H 地址中
A1:   MOV @R0, A      ; 地址+1
      INC R0          ; 地址+1
      INC A           ; A 寄存器内容+1
      DJNZ R2, A1     ; 16 个字节没满，继续送数
      MOV R0, #40H    ; 恢复 40H 为首地址

      MOV R2, #10H
      MOV DPTR, #1000H ; 外部 RAM 的首地址送
                          ; 寄存器 DPTR
A2:   MOV A, @R0      ; 将内部 RAM 内容 (A0) 送 A
      MOVX @DPTR, A   ; 通过 A 转送外部 RAM 中
      INC R0          ; 内部 RAM 地址+1
      INC DPTR        ; 外部 RAM 地址+1
      DJNZ R2, A2     ; (R2) - 1 > 0 跳到语句标号 A2
      SJMP $
      END
```

40H~4FH
单元地址
里分别置
初值：
A0H~AF
H

四、实验要求：

请在此程序上加一段把外存 1000H~100FH 的内容，再送回内存 60H~6FH 中。

实验二 分支和与循环程序

2.1 统计正整数个数实验

一、实验目的：

进一步学习分支与循环的基本技能

二、实验内容：

在一个有正、负数的数据块中，统计正整数的个数。假如每个数据都是 8 位二进制数，数据长度放在 41H 单元，数据从 42H 单元开始存放，正整的个数存放在 40H 单元中。

假如在 41~47H 单元中存放着 06H, 73H, F5H, 85H, 57H, 98H 和 2BH。

三、主程序如下：

```
ORG 2000H           ; 注意程序开始地址为 2000H
START: MOV R0, #41H   ; 取数据长度送 R2
      MOV A, @R0
      MOV R2,A
      MOV R3,#00H
LOOP1: INC R0
      MOV A, @R0      ; 取从 42H 单元开始的数据
      JB ACC.7, LOOP2 ; 判符号位为 1 跳到 LOOP2
      INC R3
LOOP2: DJNZ R2,LOOP1
      MOV 40H,R3
      SJMP $
      END
```

四、步骤：

- 1) 将源程序输入并存盘；
- 2) 了解在调用和返回时，堆栈的变化情况。
- 3) 从 41H 单元起，置入所给的 7 个数据（数据也可自选）；
- 4) 执行程序，查看结果。

2.2 统计学生成绩实验

一、实验目的：

进步学习分支与循环的基本技能。

二、实验内容：

统计60分以下，60~69分、70~79分、80~89分、90~99分及100分的人数，放在50H~55H单元中。

三、参考程序：

```
ORG    0000H
JMP    MAIN
ORG    600H
main:  MOV    50H, #00
      MOV    51H, #00
      MOV    52H, #00
      MOV    53H, #00
      MOV    54H, #00
      MOV    55H, #00
      MOV    R2, #0AH
      mov    b, #00h
      mov    a, #00h
      mov    dptr, #table
COMPA: mov    a, b
      MovC   a, @A+dptr
      clr    c
      cjne  a, #60, L5
L5:   JC    FIVE
      CJNE  A, #70, L6
L6:   JC    SIX
      CJNE  A, #80, L7
L7:   JC    SEVEN
      CJNE  A, #90, L8
L8:   JC    EIGHT
      CJNE  A, #100, L9
L9:   JC    NINE
      INC   55H
      AJMP  NEXT
NINE: INC    54H
```

： 初始化
： 50H~55H准备存放计算结果
： 计数器
： 存放中间结果
： 跳转表首地址送数据指针
： 恢复查表地址指针
： 取表中值
： 小于60跳FIVE


```

                AJMP  NEXT
EIGHT:         INC   53H
                LJMP  NEXT
SEVEN:        INC   52H
                LJMP  NEXT
        SIX:    INC   51H
                LJMP  NEXT
FIVE:         INC   50H
NEXT:         INC   B                ; 修改地址指针
                DJNZ  R2, COMPA      ; 依次重复比较, 直至R2=0
                SJMP  $

                ORG   500H
TABLE:        DB    56, 69, 84, 82, 73, 88, 99, 63, 100, 80
                END

```

四、实验要求：

- 1) 能否编一个比这个程序简练的程序。
- 2) 学会查表语句的应用。
- 3) 请看懂此并画出程序的框图。

计 算 结 果						
十个分值	56, 69, 84, 82, 73, 88, 99, 63, 100, 80					
分数范围	50~59	60~69	70~79	80~89	90~99	100
内存地址	50H	51H	52H	53H	54H	55H
结果	1	2	1	4	1	1

实验三 堆栈和子程序

3.1 十六进制数转换成ASCII码

一、实验目的：

- 1) 了解主程序和子程序的关系
- 2) 了解在实验调用和返回时，堆栈的变化情况
- 3) 进步熟悉“单步”执行方式，跟踪程序的运行情况

二、实验内容：

十六进制数转换成 ASCII 码，十六进制数的 ASCII 代码如下：

16 进制	0	1	2	3	4	5	6	7	8	9
ASCII	30H	31H	32H	33H	34H	35H	36H	37H	38H	39H

16 进制	A	B	C	D	E	F
ASCII	41H	42H	43H	44H	45H	46H

可见，转换时分两种情况：对于 9 以下的数字，加上 30H 就得到对应的 ASCII 码，对于 A 以上的数加上的数，则要加 37H (41H-0AH) = 37H。

设有 5 个 16 进制数，数组长度存在 30H 中 (05H)，以 31H 单元开始存放 5 个数为 0FH, 03H, 0AH, 0BH, 0CH，将它换成对应的 ASCII 码并存放在 51H 开始的连续单元中。

三、主程序如下：

```
ORG 2000H
STRAT: MOV R0, #30H
MOV @R0, #5H
INC R0
MOV @R0, #0FH
INC R0
MOV @R0, #03H
INC R0
MOV @R0, #0AH
INC R0
MOV @R0, #0BH
INC R0
MOV @R0, #0CH
```

```
MOV R1,#51H
MOV R0,#31H
MOV R2,30H : 数组长度
LOOP: MOV A,@R0 : 取数组值
      CLR C
      ACALL NEXT : 转换 ASCII 码
      MOV @R1,A
      INC R0
      INC R1
      DJNZ R2,LOOP
      SJMP $
```

实验 3.2 二进制数转换成压缩 BCD 码

一、实验目的:

目的与上题相同

二、实验内容:

将片内 RAM60H 的二进制数转换成压缩 BCD 码的形式存放在 62H, 63H 单元中。

三、程序清单如下:

```
ORG    3000H
MOV    R0, #60H
MOV    @R0, #6FH
START: MOV    A, 60H
        ACALL BTD
        MOV    62H, R2; 高位
        MOV    63H, R3; 低位
        SJMP   $
BTB:   MOV    B, #64H
        DIV    AB
        MOV    R2, A ; 商放在 R2
        MOV    A, #0AH
        Xch   A, B ; 余数放在 B 寄存器, AB 内容对调
        DIV    AB ; 余数除 10
        SWAP  A ; 商放在高 4 位
        ADD   A, B ; 余数放在底 4 位
        MOV   R3, A ; 组成的低 8 位送给 R3
        RET
        END`
```

四、步骤:

- 1) 仔细阅读程序并加以注解, 说明程序的功能, 指出出口及入口参数;
- 2) 输入程序至内存, 置二进制数于 60H 单元;
- 3) 单步执行程序, 检查 ACALL 指令和 RET 指令执行前后堆栈的变化;
- 4) 执行程序查看结果。

五、实验报告的要求:

- 1) 写出程序的框图并加注释;
- 2) 说明堆栈的用途并记录下堆栈在调用子程序时的情况。

实验四 80C51 综合软件设计

一、实验目的：

进一步掌握80C51的指令系统和编写程序的方法

二、实验内容：

将RAM50H~59H中的数据降序排列。根据参考程序由学生自己编一个程序，借助51单片机开发系统求出10个正整数的最大、最小值、求和或平均值。

```
ORG 0000H
REAE:AJMP MAIN
ORG 0100H
MAIN:MOV SP, #60H
MOV R0, #50H
MOV @R0, #6FH
INC R0
MOV @R0, #6EH
INC R0
MOV @R0, #6AH
INC R0
MOV @R0, #6CH
INC R0
MOV @R0, #6DH
INC R0
MOV @R0, #6BH
INC R0
MOV @R0, #67H
INC R0
MOV @R0, #69H
INC R0
MOV @R0, #68H
INC R0
MOV @R0, #64H
ACALL QUE
ACALL MAX
ACALL MIN
LCALL MEAM1
HERE: AJMP HERE
QUE: CLR 00H ; 排序
```

```
mov 5ah,r3; 求平均值
mov 5bh,r2

mov a,r2
swap a
mov b,a
mov a,r3
anl a,#0f0h
swap a
add a,b
mov b,#0ah
div ab
mov r2,a
mov a,b
swap a
mov b,A

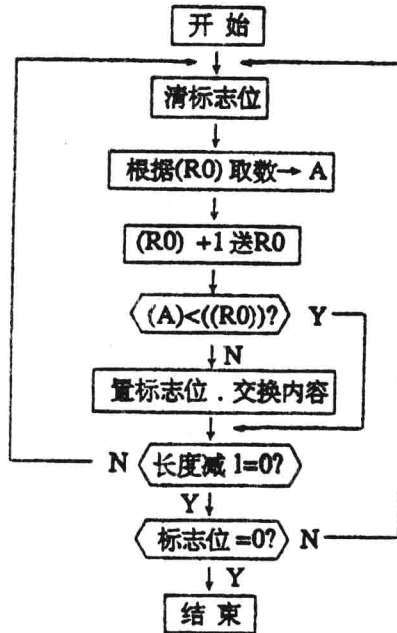
mov a,r3
anl a,#0fh
add a,b
mov b,#0ah
div ab
mov 23h,a
mov a,r2
swap a
add a,23h
mov 5ch,a
mov 5dh,b
sjmp $
```

```

MOV     R7, #09H
MOV     R0, #50H
MOV     A, @R0
L2:    INC     R0
MOV     R2, A
SUBB   A, @R0
MOV     A, R2
JNC    L1
SETB   00H
XCH    A, @R0
DEC    R0
XCH    A, @R0
INC    R0
L1:    MOV     A, @R0
DJNZ   R7, L2
JB     00H, QUE
RET

```

数据实验排队程序框图如下：



MAX:	MOV	R1, #30H ;最大	MIN:	MOV	R1, #20H ;最小
	MOV	R0, #50H		MOV	R0, #50H
	MOV	R2, #09H		MOV	R2, #09H
	MOV	A, @R0		MOV	A, @R0
	INC	R0		INC	R0
LOOP:	CLR	C	LOOP1:	CLR	C
	MOV	R7, A		MOV	R7, A
	SUBB	A, @R0		SUBB	A, @R0
	MOV	A, R7		MOV	A, R7
	JNC	NOCH		JC	NOCH1
	MOV	A, @R0		MOV	A, @R0
NOCH:	INC	R0	NOCH1:	INC	R0
	DJNZ	R2, LOOP		DJNZ	R2, LOOP1
	MOV	@R1, A		MOV	@R1, A
	RET			RET	

MEAM1: CLR	C	; 求和	clr c	; 求和
MOV	R0, #50H		mov	r2, #00h
MOV	R1, #10		mov	r3, #9h
MOV	R5, #00H		mov	r0, #50h
MOV	A, R1		mov	a, @r0
MOV	R7, A		L4: inc	r0
MOV	R2, #00H		add	a, @r0
MOV	R3, #00H		jnc	L3 ; 无进位跳
LPO: MOV	A, R2		inc	r2
ADD	A, @R0		L3: djnz	r3, L4
MOV	R2, A		mov	r3, a
MOV	A, R3		mov	5eh, r3
ADDC	A, #00H		mov	5fh, r2
MOV	R3, A			
INC	R0		MOV	5AH, R2
DJNZ	R7, LPO		MOV	5BH, R3
MOV	5AH, R2			
MOV	5BH, R3		INC	R3
RET			MOV	R4, #00H
			MOV	A, R2
			ZH: CLR	C
			SUBB	A, #0AH
			INC	R4
			JNC	ZH
			DJNZ	R3, ZH
			DEC	R4
			MOV	5CH, R4
			RET	

实验五 P1 口亮灯实验

一、实验目的：

- 1) 学习 P1 口的使用方法；
- 2) 学习延时子程序的编写。

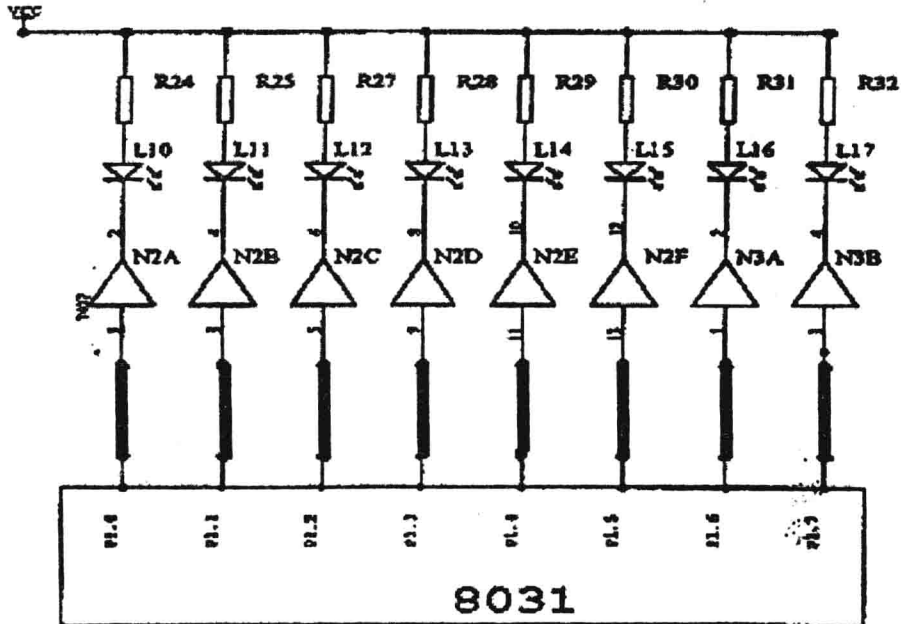
二、实验内容：

P1 口做输出口，接八只发光二极管，编程序，使发光二极管循环点亮。

三、实验预备知识：

- 1) P1 口为一组双向口，每位都可独立地定义为输出或输入，在作输入线使用前，必须向锁存器相应位写入“1”，该位才能作为输入（即：`MOV P1, 0FFH`）。
- 2) 本实验中延时子程序采用循环来实现，定时长度=机器周期（ $12M/6$ ）×机器所需机器周期数×循环次数，在系统时间允许的情况下可以采用此方法。
- 3) `RL A`、`RR A` 指令为不带进位循环位移，`CPL A`； $A \rightarrow A$

三、实验接线图：



五、实验步骤：

用 P1.0~P1.7 与 L1~L8 分别相连程序后，观察发光二极管闪亮移位情况。

六、参考程序：

```
                ORG    0000H
                JMP    START
                ORG    0200H
START:  MOV    P1, #0FFH
        MOV    A,  #0FEH
LOOP1:  MOV    P1, A
        RL     A           ; 左移
        SJMP  LOOP1

        DELAY: MOV    R6, #0FFH
        DEL2:  MOV    R7, #0FFH
        DEL3:  DJNZ  R7, $
                DJNZ  R6, DEL2
                RET
                END
```

七、实验要求：

- 1) 改变延时时间常数，使发光二极管闪亮时间改变。
- 2) 修改程序，使发光二极管闪亮移位方向改变。