实 验 报 告

学号：21160809
计算机学院计科八班
张 炎

实验三　步进电机原理及应用

1.实验原理

本实验使用简单的双四拍工作模式即可，这也是 FAN8200 比较方便的工作方式。只要将 CE1 和 CE2 分别置为高，然后 IN1 和 IN2 按照预定的脉冲输出，即 01->11->10->00->01 这个循环构成一个方向旋转的输出脉冲，将此序列翻转，就是相反方向的输出脉冲。

本开发平台有 3 个数码管，使用串行方式连接在一起，具体电路参见实验原理。要想输出一个字形码，就需要从高位到低位依次向移位寄存器输出 8 个比特。移位寄存器的数据线和时钟线分别接到单片机的 P4.5 和 P4.4 管脚，可以使用 MCS-51 里面的位操作指令进行输出。连续输出 3 个字形，24 个 bit 之后，欲显示的字形将稳定地显示在数码管上，程序可以转而执行其他工作。

2.实验内容

1）编制 MCS-51 程序使步进电机按照规定的转速和方向进行旋转，并将已转动的步数显示在数码管上。

2）步进电机的转速分为两档，当按下 S1 开关时，进行快速旋转，速度为 60 转/分。当松开开关时，进行慢速旋转，速度为 10 转/分。当按下 S2 开关时，按照顺时针旋转；当松开时，按照逆时针旋转。

3.实验步骤

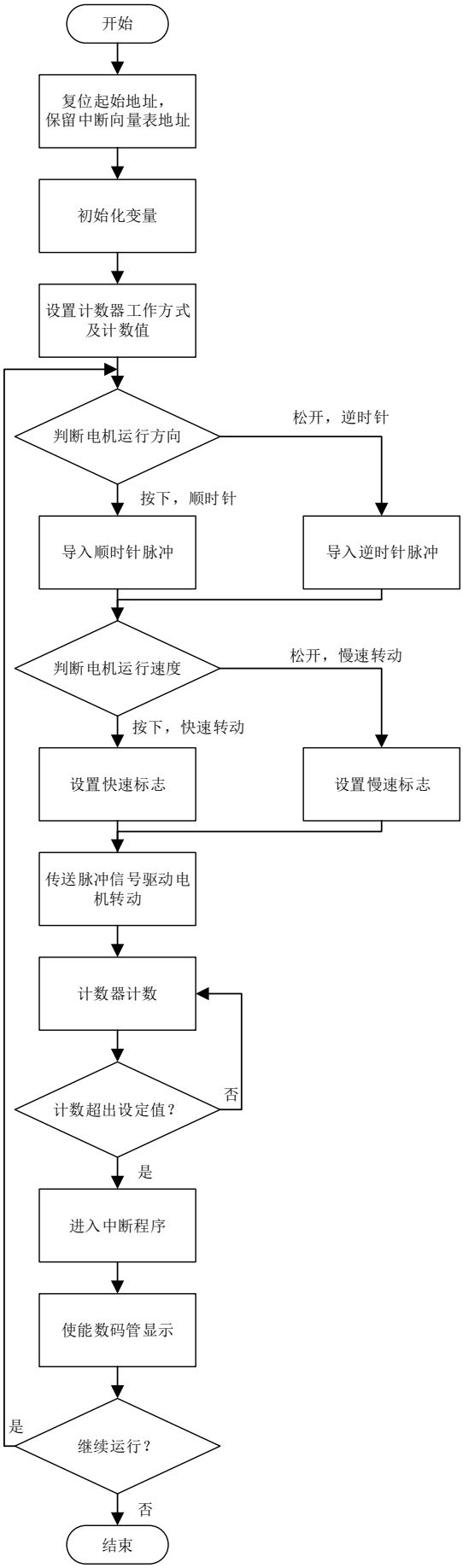
1）预习：参考附录二、附录三和 expr/资料/原理的辅助材料，学习 MCS-51 汇编语言使用和步进电机原理，阅读数码显示器的电路图，重点理解步进电机的工作方式和数码管显示方式。

2）简单程序录入和调试：MCS51 单片机汇编语言的基本格式比较简单，程序中可以使用通用寄存器或者内存单元进行计算。另外，单片机的程序没有退出到操作系统的概念，一般都是死循环程序。

3）程序调试。

4）编写程序，完成功能。

I.程序流程图


开始
复位起始地址，
保留中断向量表地址
初始化变量
设置计数器工作方式
及计数值
判断电机运行方向
松开，逆时针
按下，顺时针
导入顺时针脉冲
导入逆时针脉冲
判断电机运行速度
松开，慢速转动
按下，快速转动
设置快速标志
设置慢速标志
传送脉冲信号驱动电
机转动
计数器计数
计数超出设定值？
否
是
进入中断程序
使能数码管显示
是
继续运行？
否
结束

II. 程序清单

```
ORG 0000H
        LJMP START
ORG 000BH
        LJMP EINT0
ORG 0040H
START:
        P4     EQU 0C0H
        P4SW EQU 0BBH
        CLK    EQU P4.4
        DAT    EQU P4.5
        SW     EQU P3.6

        MOV P4SW,#70H
        MOV DPTR,#TAB

LP:
        MOV R3,#0
        MOV R4,#0
        MOV R5,#0

I1:    MOV TMOD,#01H
        MOV IE,#82H
        MOV IP,#2H
        SETB P1.1
        SETB P1.4

NEXT:
        JB    P3.7,OPP
        MOV R0,#10110100B
        MOV 20H,R0
        LJMP SS1

OPP: MOV R0,#11100001B
        MOV 20H,R0

SS1:
        JB    P3.6,SPD
        MOV R2,#0H
        LJMP L0

SPD: MOV R2,#1H
```

```
L0:   MOV R1,#4
      MOV R0,20H

L1:
      MOV A,R0
      RLC A
      MOV P3.2,C
      RLC A
      MOV P1.0,C
      MOV R0,A
      LCALL NUM
     LCALL TIME
     DJNZ R1,L1

      LJMP NEXT

TIME:
      CJNE R2,#1,QUICK
      MOV R6,#6

TIM2:   MOV TH0,#5DH
        MOV TL0,#3EH
        SETB TR0
        MOV R7,#0H

TIM3: CJNE R7,#1H,TIM3
        DJNZ R6,TIM2
        LJMP OUT

QUICK:   MOV TH0,#5DH
          MOV TL0,#3EH
          SETB TR0
          MOV R7,#0H

TIM1:   CJNE R7,#1H,TIM1

OUT:
RET

EINT0:
        MOV R7,#1
RETI

NUM:
```

```
S0:     MOV A,R3
        CALL EXP
        MOV A,R4
        CALL EXP
        MOV A,R5
        CALL EXP

        CJNE R3,#9,S1
        MOV  R3,#0
        CJNE R4,#9,S2
        MOV  R4,#0
        CJNE R5,#9,S3
        MOV  R5,#0

S1:     INC R3
        LJMP STOP

S2:     INC R4
        LJMP STOP

S3:     INC R5
        LJMP STOP

STOP:
RET

EXP:
      MOV 21H,R0
      MOVC A,@A+DPTR
      MOV R0,#8

CLY:  CLR CLK
      RLC A
      MOV DAT,C
      SETB CLK

      DJNZ R0,CLY
       MOV    R0,21H
RET

TAB:
DB 0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H,80H,90H
END
```

4. 思考题

1）如采用单四拍工作模式，每次步进角度是多少，程序要如何修改？

答：**步进角度为 15°，程序应该改变相位值。**

2）如采用单双八拍工作模式，每次步进角度是多少，程序要如何修改？

答：**步进角度为 7.5°，定时器定时周期变了，故需修改定时初始值；相位值也需改变，还有循环次数。**

3）步进电机的转速取决于那些因素？有没有上、下限？

答：**取决于脉冲频率、转子齿数和拍数。有，上限根据电机不同而不同，下限为 0。**

4）如何改变步进电机的转向？

答：**改变脉冲信号的顺序。**

5）步进电机有那些规格参数，如何根据需要选择型号？

答：**功率，马力，电流，转速，效率，功率因数，额定转矩，额定电流，重量。判断需多大力矩，判断电机运转速度，选择电机的安装规格。**

6）MCS51 中有哪些可存取的单元，存取方式如何？它们之间的区别和联系有哪些？

答：**MCS-51 存储器片内 RAM、片外 RAM、ROM 三个空间。片内 RAM 必须使用 MOV 指令，片外 RAM 必须使用 MOVX 指令，片外 ROM 必须使用 MOVC 指令。**

7）说明 MOVC 指令的使用方法。

答：**MOVC 是累加器与程序存储区之间的数据传送指令。它比 MOV 指令多了一个字母“C”，这个“C”就是“Code”的意思，翻译过来就是“代码”的意思，就是代码区（程序存储区）与 A 之间的数据传送指令。它可以用于内部程序存储区（内部 ROM）与 A 之间的数据传送，也可以 用于外部程序存储区（外部 ROM）与 A 之间的数据传送。因为程序存储区内外统一编址，所以一条指令就可以了。**

8）MCS51 的指令时序是什么样的，哪类指令的执行时间较长？

答：**大多数 MCS－51 的指令执行时间为一个机器周期，小部分指令为 2 个机器周期，只有乘除法需要 4 个机器周期。**

5.问题及分析

在程序的编写过程中，出现了无法实现要求的现象，经过排查，发现对步进电机的驱动方式理解有误，后经过修改，逐步细化，实现了程序要求。但在老师的指点中，发现程序存在误差，可进一步通过修改程序来优化消除误差。

实验四　LED 点阵显示屏

1.实验原理

高亮度 LED 发光管构成点阵，通过编程控制可以显示中英文字符、图形及视频动态图形。

实验用的 LED 点阵显示屏为 16*16 点阵。行和列分别使用两个移位寄存器作为输出。当移位寄存器输出的第 i 行为 0，第 j 列为 1 时点亮点(i,j)。为了能够显示出一个点阵字型，需要进行循环扫描，也就是每一次只点亮一行，然后在列上输出该列对应的 16 个点阵值。输出一行后暂停一段时间，输出下一行。为了达到较好的显示效果，整屏总的扫描时间不高于 40ms。上述过程中行列可以互换。

在控制 74HC595 时,首先将数据放到串行输入的 SI 端,然后在串行时钟 SRCK 上产生一个脉冲，即可输出一个 bit，重复以上步骤 16 次，输出所有列值。然后给存储器时钟 RCK 一个脉冲,将串行数据锁存起来。将使能端　　输出低电平，驱动到 LED 点阵上。行的输出每次只移位一次，并重新锁存即可。

2.实验内容

1）了解 16*16 点阵电路的原理，编写汇编语言程序。

2）编写一行汉字字符（至少三个字）的显示程序。

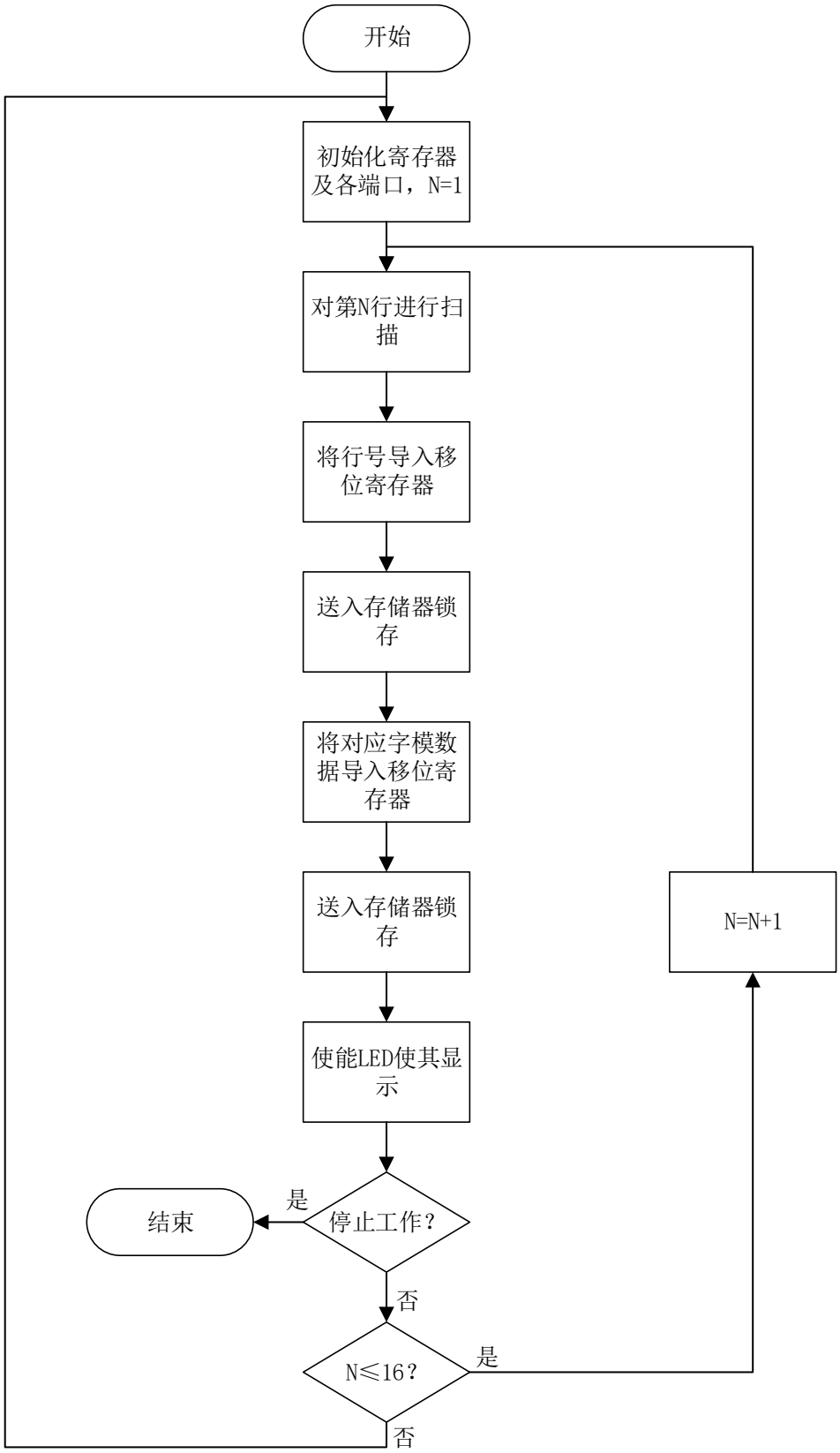
3）能够从左到右（或从右到左）循环显示（要求显示过程中字的大小与屏幕尺寸相适应）。

3.实验步骤

1）掌握点阵式 LED 显示屏的控制方法。

2）使用 MCS-51 汇编语言，使用 LED 点阵显示器显示出正确的汉字字符及动态效果。

I.程序流程图


开始
初始化寄存器及各端口，N=1
对第N行进行扫描
将行号导入移位寄存器
送入存储器锁存
将对应字模数据导入移位寄存器
送入存储器锁存
使能LED使其显示
停止工作？
是
结束
否
N≤16？
是
否
N=N+1

II. 程序清单

```
ORG 0000H                          ;复位起始地址
    LJMP START                     ;中间地址保留给中断向量表
ORG 0050H                          ;程序实际起始地址
 START:
XSDATA_595    EQU P0.0             ;串行移位行输入，DX
XSRCK_595     EQU P0.1             ;移位寄存器时钟输入
XRCK_595      EQU P0.2             ;存储器时钟输入
XEN_595       EQU P0.7             ;使能端
YSDATA_595    EQU P0.3             ;串行移位列输入，DY          ;
YSRCK_595     EQU P0.5             ;移位寄存器时钟输入
YRCK_595      EQU P0.6             ;存储器时钟输入
YEN_595       EQU P0.4             ;使能端
TT1: MOV DPTR,#TAB
     MOV R0,#00H                   ;行
     MOV R2,#01H
     MOV R5,#00H                   ;列
     CALL TT2

   MOV R0,#00H
   MOV R2,#02H
   CALL TT2

   MOV R0,#00H
   MOV R2,#04H
   CALL TT2

   MOV R0,#00H
   MOV R2,#08H
   CALL TT2

   MOV R0,#00H
   MOV R2,#10H
   CALL TT2

   MOV R0,#00H
   MOV R2,#20H
   CALL TT2

   MOV R0,#00H
   MOV R2,#40H
   CALL TT2

   MOV R0,#00H
```

```
    MOV R2,#80H
    CALL TT2

    MOV R0,#01H
    MOV R2,#00H
    CALL TT2

    MOV R0,#02H
    MOV R2,#00H
    CALL TT2

    MOV R0,#04H
    MOV R2,#00H
    CALL TT2

    MOV R0,#08H
    MOV R2,#00H
    CALL TT2

    MOV R0,#10H
    MOV R2,#00H
    CALL TT2

    MOV R0,#20H
    MOV R2,#00H
    CALL TT2

    MOV R0,#40H
    MOV R2,#00H
    CALL TT2

    MOV R0,#80H
    MOV R2,#00H
    CALL TT2
    JMP TT1

TT2:   MOV R1,#08H
       MOV A,R0                          ;将 R0 的内容送到第一个移位寄存器
WR_LOOP:
       RLC A
       CLR XSRCK_595
       MOV XSDATA_595,C
       SETB XSRCK_595
       DJNZ R1,WR_LOOP
```

```
        MOV A,R2
        MOV R1,#08H
WR_LOOP1:
        RLC A
        CLR XSRCK_595
        MOV XSDATA_595,C                ;将 R2 的内容送到第二个移位寄存器
        SETB XSRCK_595
        DJNZ R1,WR_LOOP1

        CLR XRCK_595
        NOP
        NOP                             ;将数据送到存储器
        SETB XRCK_595
        NOP
        NOP
        CLR XEN_595                     ;使能端为低电平，驱动到 LED 点阵上
        CALL L1
RET

L1:
        MOV R6,#08H
        MOV A,R5
        MOVC A,@A+DPTR
WR_LOOP2:
        RLC A
        CLR YSRCK_595
        CPL C                           ;取反
        MOV YSDATA_595,C        ;将 TAB 中第 R5 个单元的内容送到第一个移位寄存器
        SETB YSRCK_595
        DJNZ R6,WR_LOOP2

        MOV R6,#08H
        INC R5
        MOV A,R5
        MOVC A,@A+DPTR
WR_LOOP3:
        RLC A
        CLR YSRCK_595
        CPL C
        MOV YSDATA_595,C      ;将 TAB 中第 R5+1 个单元的内容送到第二个移位寄存器
        SETB YSRCK_595
        DJNZ R6,WR_LOOP3
```

```
        INC R5
        CJNE R5,#5FH,MM1
        MOV R5,#0
 MM1:
        CLR YRCK_595
        NOP
        NOP
        SETB YRCK_595                    ;将数据送到存储器
        NOP
        NOP

        CLR YEN_595                      ;低电平使能
        CALL DELAY
RET
DELAY:
        MOV   R7,#10
LOOPS:
        MOV   B, #25
LOOPR:
        DJNZ B, LOOPR
        DJNZ R7,LOOPS
RET

TAB:
DB 12H,22H,0AH,22H,47H,0A4H,28H,24H,00H,28H,04H,0B0H,59H,20H,42H,0FFH;
DB 7CH,20H,41H,30H,40H,0A8H,7FH,24H,00H,24H,10H,22H,0CH,22H,00H,00H;"梁",0
DB 20H,10H,2FH,0D0H,2AH,90H,2AH,90H,0FAH,0FFH,2AH,90H,2AH,90H,2FH,0D1H;
DB   20H,12H,00H,0CH,7FH,0F0H,44H,40H,44H,42H,44H,41H,7FH,0FEH,00H,00H;" 朝
",1
DB 00H,80H,01H,00H,06H,00H,1FH,0FFH,0E0H,00H,10H,40H,12H,40H,12H,40H;
DB   12H,40H,0FFH,0FFH,12H,40H,12H,40H,12H,48H,12H,44H,10H,78H,00H,00H;" 伟
",2
END
```

3）将编译后的程序下载到 51 单片机，观察 LED 显示屏的显示结果。

4.思考题

1）如何使用软件调整和控制 LED 点阵的亮度

答：可利用 PWM 即脉宽调制的方法来对 LED 点阵的亮度进行调整和控制。

2）如何尽量避免显示过程中的闪烁

答：显示过程中闪烁有可能是因为刷新率过低，可通过提高刷新率的方法来解决。

3）如何将本实验的软硬件推广到多行多列的 LED 显示屏（如 64*1280）

答：将多行多列 LED 显示屏划分为多个子块，分别设计驱动程序，并交由一个总控程序来统一驱动，由此来解决闪烁等问题。

5.问题及分析

在实验过程中，先是出现了对硬件理解不清导致的无法将字模导出到 LED 点阵屏上，后充分理解移位寄存器及锁存器的工作过程及注意事项，解决了问题；之后在调试过程中，又出现了字体显示不全、闪烁等问题，后通过修改程序中循环次数和加快扫描时间等方法解决了问题。