

实验三 步进电机原理及应用

53160803 张天悦

一、实验目的

- (1) 初步学习和掌握 MCS-51 的体系结构和汇编语言，了解 Keil 编程环境和程序下载工具的使用方法。
- (2) 了解步进电机的工作原理，学习用单片机的步进电机控制系统的硬件设计方法，掌握定时器和中断系统的应用，熟悉单片机应用系统的设计与调试方法。
- (3) 了解数码管输出的原理及编程方式。

二、实验原理

1、本实验采用定时器中断实现，使用定时器时，首先应由外部条件得到要定时的时间长度 t ，如本实验中，就是根据要求的速度计算出的每一步之间的间隔。然后选择适当的定时器工作方式，去计算想要设定的计数器初值 s ，使用如下方程： $(2^{\text{定时器最大位数}} - s) \times \text{定时周期} = t$

定时周期 = $12/\text{CPU 晶振频率}$

$(2^{\text{定时器最大位数}} - s) \times \text{定时周期} = t$

得到的 s 需要分成高 8 位和低 8 位，分别放入计数器 THx 和 TLx 中(x 为 0 或 1)。如果 s 为负数，说明需要的定时时间太长，即使定时器的最大时间也无法满足要求。这种情况下，需要加入软件循环才能实现。我们可以将需要的定时时间分成 n 份，利用定时器达到 t/n 的时间长度，然后在定时器处理程序中，累计某一变量，如果到达 n ，说明总的时间 t 已经达到。

要想使用定时器中断，除了上面的定时器初值设定外，还需要将其他相关的特殊功能寄存器也都设置好。如果使用方式 0 和方式 1，不要忘记在计数结束后重新恢复计数器初值。

2、我们使用的单片机系统的频率是 12M；步进电机转动一周需要 24 步。

本步进电机实验板，使用 FAN8200 作为驱动芯片。CPU 通过如下 4 个引脚与 FAN8200 相连，即：

CPU	FAN8200
P1.1	CE1
P1.4	CE2
P3.2	IN1
P1.0	IN2

3、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。

4、数码管显示：

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。

七段字形的编码方式需要通过实验获得。这些编码作为程序中的常数，使用 DB 命令存放。在程序中，需要将数值转换为相应的字形编码，可以使用 MOVC 指令来完成。

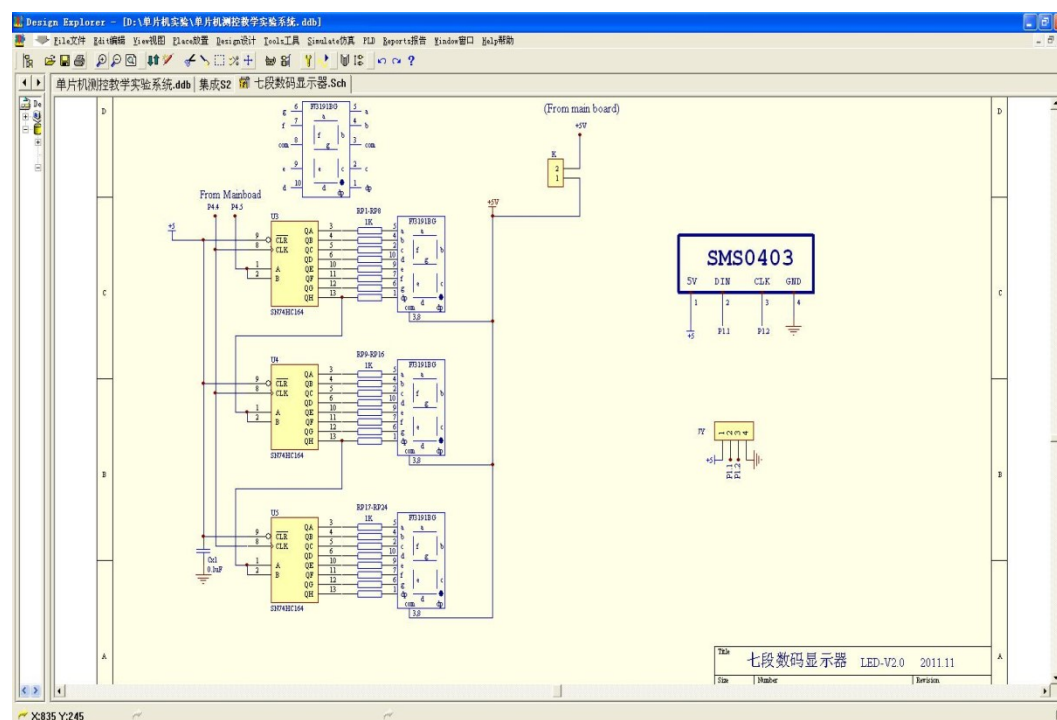
5、74HC164 是高速 CMOS 器件。74HC164 是 8 位边沿触发式移位寄存器，串行输入数据，然后并行输出。数据通过两个输入端（A 或 B）之一串行输入；任一输入端可以用作高电平使能端，控制另一输入端的数据输入。两个输入端或者连接在一起，或者把不用的输入端接高电平，一定不要悬空。

6、时钟（CLK）每次由低变高时，数据右移一位，输入到 Q0，Q0 是两个数据输入端（A 和 B）的逻辑与，它将上升时钟沿之前保持一个建立时间的长度。

7、主复位（CLR）输入端上的一个低电平将使其所有输入端都无效，同时非同步地清除寄存器，强制所有的输出为低电平。

8、采用 3 个 74HC164 级联控制三个数码管的显示，具体实验原理如下图所示。其中使用单片机 P4.5 作为模拟串口数据，使用 P4.4 模拟串口时钟，CLR 端接高电平。使用上一个 74HC164 的 Q7 作为下一个 74HC164 的输入端。

原理图：



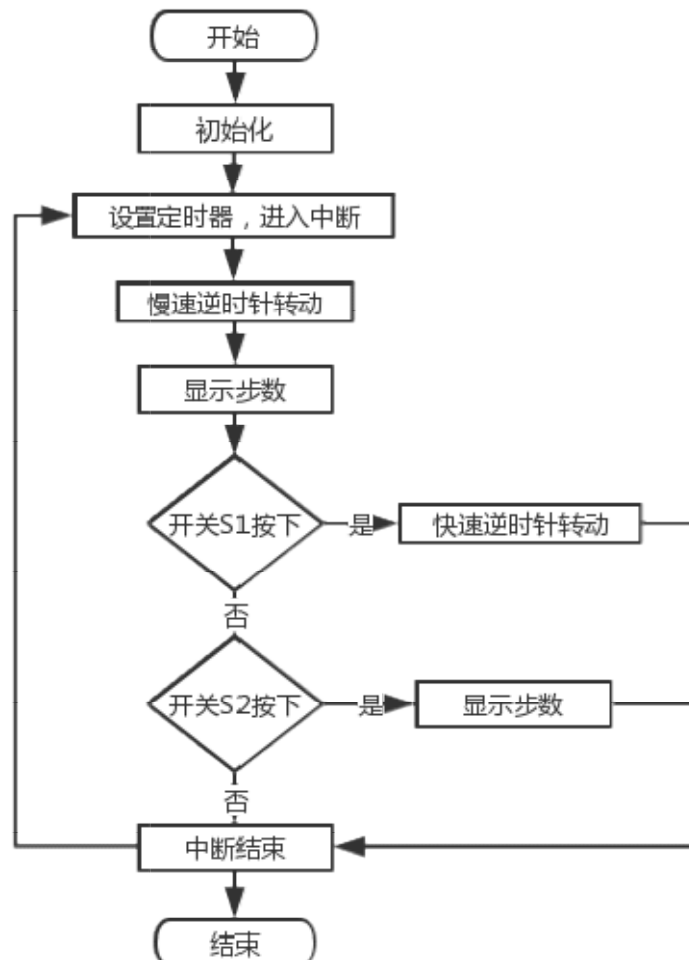
三、实验内容

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. 本程序要求使用定时器中断来实现，不准使用程序延时的方式。

四、流程图



五、实验代码

```
ORG 0000H                ;复位起始地址
    LJMP START
ORG 000BH                ;中间地址保留给中断向量表
    LJMP EINT0            ;定时器 0 中断程序入口地址
ORG 0040H                ;程序实际起始地址
START:
    P4      EQU 0C0H
    P4SW EQU 0BBH
```

```

CLK EQU P4.4 ;时钟线
DAT EQU P4.5 ;数据线
SW EQU P3.6

MOV P4SW,#70H
MOV DPTR,#TAB ;s = 23869 --5D3E

LP:
MOV R3,#0 ;计数
MOV R4,#0
MOV R5,#0
l1: MOV TMOD,#01H ;选择工作方式，即对 TMOD 赋初值。T0 工作在方式 1，16
位的计数器 GATE(GATE 一门控位，控制定时器的两种启动方式，)等于 0，不受外部控制

MOV IE,#82H ;全局中断，T0 中断允许 中断控制字;直接对中断允许寄存器
IE 和 优先级寄存器 IP 设置
;ORL IP,#2H ;逻辑或，T0 中断优先级高

SETB P1.1 ;CE1 置高
SETB P1.4 ;CE2 置高

NEXT:
JB P3.7,OPP ;如果 P3.7 等于 1 则转移
MOV R0,#00101101B ;按下，顺时针
MOV 20H,R0

LJMP SS1
OPP: MOV R0,#01111000B ;松开，逆时针
MOV 20H,R0
SS1:
JB P3.6,SPD
MOV R2,#0H ;按下，快速 ;23870 5D3E
LJMP L0
SPD: MOV R2,#1H ;松开，慢速

L0: MOV R1,#4
MOV R0,20H
L1: MOV A,R0
RLC A ;循环左移操作
MOV P3.2,C ;IN1
RLC A

```

```

    MOV P1.0,C          ;IN2
    MOV R0,A
    LCALL NUM
    LCALL TIME
    DJNZ R1,L1

    LJMP NEXT

TIME:
    CJNE R2,#1,QUICK
    MOV R6,#6            ;慢速            ;分 6 次来计时
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      ;定时器 0 启动;;快速,60 转/分
    MOV TL0,#3EH
    SETB TR0
    MOV R7,#0H
TIM1:  CJNE R7,#1H,TIM1
OUT:
RET
EINT0:
    MOV R7,#1
RETI
NUM:   ;显示已转动的步数，每转动一次显示一个数
S0:    MOV A,R3
    CALL EXP
    MOV A,R4
    CALL EXP
    ;mov r5,0ffh
    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
        LJMPL STOP
S2:     INC R4
        LJMPL STOP
S3:     INC R5
        LJMPL STOP
STOP:
RET

EXP:
        MOV    21H,R0
        MOVC  A,@A+DPTR
        MOV    R0,#8
CLY:    CLR CLK    ;P4.4    ;时钟线低电平
        RLC A      ;累加器 A 的逻辑操作指令
        MOV    DAT,C
        SETB CLK   ;P4.4    ;时钟线高电平

        DJNZ   R0,CLY
        MOV    R0,21H
RET

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

```

六、思考题

1. 如采用单四拍工作模式，每次步进角度是 15° ，程序要如何修改？

单四拍：A-B-~A-~B-A

01-10-00-00

修改：改变相位值

2. 如采用单双八拍工作模式，每次步进角度是 7.5° ，程序要如何修改？

八拍：A-AB-B-BC-C-CD-D-DA-A

01 11 10 10 00 00 00 01

修改：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？

脉冲频率。

步进机的转速由上下限：会受物理因素（包括摩擦、机械惯性、响应时间）影响，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？

通过改变转动脉冲。

七、实验心得

中断中执行的程序太多，但由于使用的是定时器中断，不能保证在程序在定时周期内执行完毕。因此在定时器中断设计过程中应避免在中断程序中执行了过多程序。

实验四 LED 点阵显示屏

三、 实验目的

了解 LED 点阵显示的基本原理和实现方法。掌握点阵汉字库的编码和从标准字库中提取汉字编码的方法。

二、实验原理

1. 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。所显示字符的点阵数据可以自行编写（即直接点阵画图），也可从标准字库（如 ASC16、HZ16）中提取。后者需要正确掌握字库的编码方法和字符定位的计算。

2、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。

3、实验中使用的移位寄存器是 74HC595，它是一个同时具有串行移位和输出锁存驱动功能的器件。74HC595 是具有 8 位移位寄存器和一个存储器，三态输出功能。移位寄存器和存储器是分别的时钟。

4、数据在 SRCK（移位寄存器时钟输入）的上升沿输入到移位寄存器中，在 RCK（存储器时钟输入）的上升沿输入到存储寄存器中去。

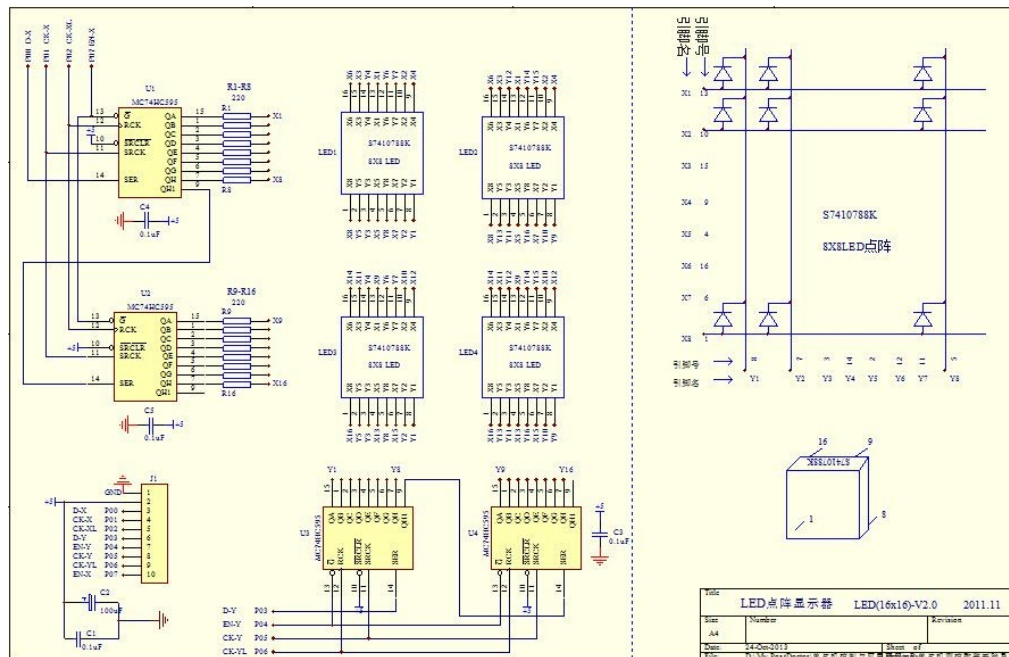
移位寄存器有一个串行移位输入（行 D_x (P00)、列 D_y (P03)），和一个串行输出（QH），和一个异步的低电平复位，存储寄存器有一个并行 8 位的，具备三态的总线输出，当使能（P02 和 P07 为低电平）时，存储寄存器的数据输出到总线。

5、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。

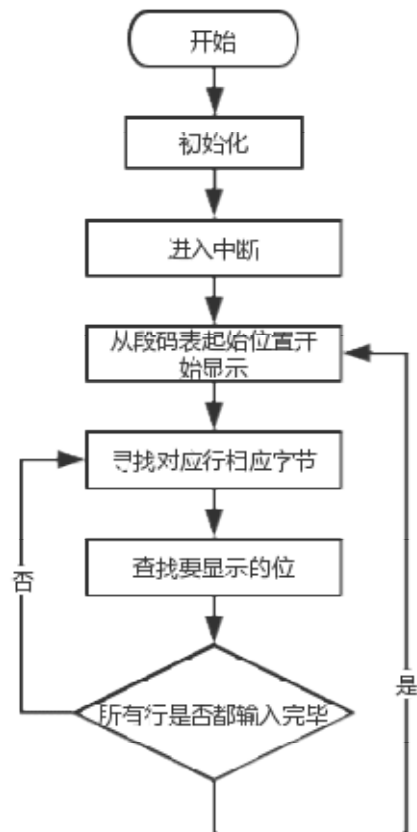
原理图：



三、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）。

四、流程图



五、实验代码

ORG 0000H

LJMP START

ORG 000BH

LJMP TOIN

ORG 0040H

CHINA:

DB 00H,00H,00H,00H,00H,00H ;显示汉字，16*48

DB 00H,00H,00H,00H,00H,00H

DB 40H,80H,20H,80H,10H,80H

DB 20H,80H,0fbH,0e0H,7fH,0e0H

DB 0fH,0f0H,70H,0c0H,10H,80H

DB 0e2H,00H,0aaH,0a0H,00H,00H

DB 24H,80H,20H,80H,10H,00H

DB 2fH,0f0H,08H,00H,0fdH,0e0H

DB 20H,80H,1fH,00H,29H,20H

DB 2fH,0f0H,31H,00H,49H,20H

DB 20H,80H,0aH,00H,29H,20H

```
DB 50H,80H,04H,00H,11H,20H
DB 8fH,0f8H,18H,00H,029H,0e0H
DB 00H,00H,00H,00H,00H,00H
DB 00H,00H,00H,00H,00H,00H
DB 00H,00H,00H,00H,00H,00H
```

START:

```
ROWBLN EQU 48 ;数组宽度
ROWLEN EQU 6
DISLEN EQU 16 ;视窗宽度
TIMLEN EQU 2
X_DAT EQU P0.0 ;串行移位输入
Y_DAT EQU P0.3 ;串行移位输入
X_DCLK EQU P0.1 ;移位寄存器时钟输入
Y_DCLK EQU P0.5
X_BCLK EQU P0.2 ;存储器时钟输入
Y_BCLK EQU P0.6
X_EBLE EQU P0.7 ;使能，低电平时输出存储寄存器数据到总线
Y_EBLE EQU P0.4
```

```
MOV SP,#30H ;栈指针
```

MOV TMOD,#01H ;T0 计数器，方式 1，高 4 位和低 4 位分别控制 T1 和 T0 GATE 为 0 时定时器不受外部控制 M1M0 为 01 时为方式 1，方式 1 与方式 0 的区别仅在于它是 16 的定时/计数器，由 TH 和 TL 联合构成。

```
MOV IE,#82H ;允许 CPU 和 T0 响应中断
```

```
MOV IP,#02H ;T0 中断优先级为高
```

```
MOV TH0,#3CH
```

```
MOV TL0,#0B0H ;计数初值，TH0，TL0 为 T0 的 16 位计数的高 8 位和低 8 位
```

MOV TCON,#10H ;TR0=1 允许计数，TF0 为 1 时标志已达到最大计数值，发生溢出，产生定时器中断，IT0=0 为电平触发方式

```
MOV R0,#00H//SHIFT
```

```
MOV R1,#0FH//DISOFFSET
```

```
MOV R2,#00H//ROW
```

```
MOV R4,#TIMLEN//TIMER
```

```
MOV DPTR,#CHINA
```

FLUSH:

```
CLR X_BCLK ;清除
```

```
CLR Y_BCLK
```

```
CLR X_EBLE
```

```
CLR Y_EBLE
```

OUTPUT:

CLR X_DCLK

CLR Y_DCLK

MOV A,R2

MOV B,#ROWLEN

MUL AB ;A 和 B 中无符号 8 位数相乘，16 位积低位放在 A 中，高位放在 B 中

MOV R3,A ;切换显示行

MOV A,R0

ADD A,R1 ;要显示的内容

MOV B,#ROWBLN ;48

DIV AB ;A 除以 B，整数放在 A 中，余数放在 B 中

MOV A,B

MOV B,#8

DIV AB ;A 存放显示查询到的字节，B 存放字节中要显示的位

ADD A,R3 ;寻找对应行相应字节

MOVC A,@A+DPTR ;查表，8 位累加器 A 作为变址寄存器，DPTR 作为基址寄存器，相加作为 16 位地址访问程序存储器

MOV R3,B

CJNE R3,#00H,FINDBIT ;查找要显示的位

JMP COLSET

FINDBIT:

RL A ;左循环一位，位 7 入 0，不影响标志

DJNZ R3,FINDBIT ;源操作数减 1 回送，结果不为 0 则转移

COLSET:

RLC A ;带进位标志左循环一位，位 7 入 CY，CY 入 0，不影响标志

MOV X_DAT,C ;CY (PSW.7) 作为累加器 C，存放要显示的位

CLR C

MOV A,#15

SUBB A,R2 ;判断是否所有行已输入结束

CJNE A,01H,ROW_OFF ;不相等则转移，若第一操作数小于第二操作数则置位 CY，否则 CY 清零

CLR Y_DAT

JMP UPJMP

ROW_OFF:

SETB Y_DAT

UPJMP:

SETB X_DCLK

SETB Y_DCLK

DEC R1

CJNE R1,#0FFH,OUTPUT

SETB X_BCLK

SETB Y_BCLK

SETB X_EBLE

SETB Y_EBLE

MOV R1,#0FH

INC R2

CJNE R2,#10H,FLUSH

MOV R2,#00H

JMP FLUSH

TOIN:

MOV TCON,#00H

DJNZ R4,TOEND ;源操作数减 1 回送，结果不为 0 则转移

MOV R4,#TIMLEN

INC R0 ;数组向右移动一位

CJNE R0,#ROWBLN,TOEND

MOV R0,#00H ;若数组显示完毕则从数组最左端重新开始

TOEND:

MOV TH0,#3CH

MOV TL0,#0B0H

MOV TCON,#10H

RETI

END

六、思考题

1. 如何使用软件调整和控制 LED 点阵的亮度？

答：对于 LED 显示屏，点的亮度值对应于有效的显示时间，由列的打开时间来实现的。列的打开时间越长，亮度越高。

2. 如何尽量避免显示过程中的闪烁？

答：对于用户来说，显示频率越高，屏的视觉稳定性越强；显示频率越低，屏的视觉稳定性越差，即显示屏看起来会有闪烁和抖动的现象。既应提高显示频率。

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：可以增加新的 LED 和 74hc59 实现。

七、实验心得

在段码表方面要注意阴极阳极，取模方式，定义格式等。要调整延时程序避免字符显示过快。