

# 单片机第三次实验报告

学号：53160819 姓名：李申瑞

## 一、实验题目：步进电机原理及应用

## 二、实验原理：

定时器计时，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： **$(2^{\text{定时器最大位数}} - s) \times \text{定时周期} = t$** ， **$\text{定时周期} = 12/\text{CPU 晶振频率}$** 。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）。我们将定时时间分成  $n$  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  $n$ ，说明总的的时间  $t$  已经达到，此时产生中断使相应寄存器置 1，并重新恢复计数器初值。

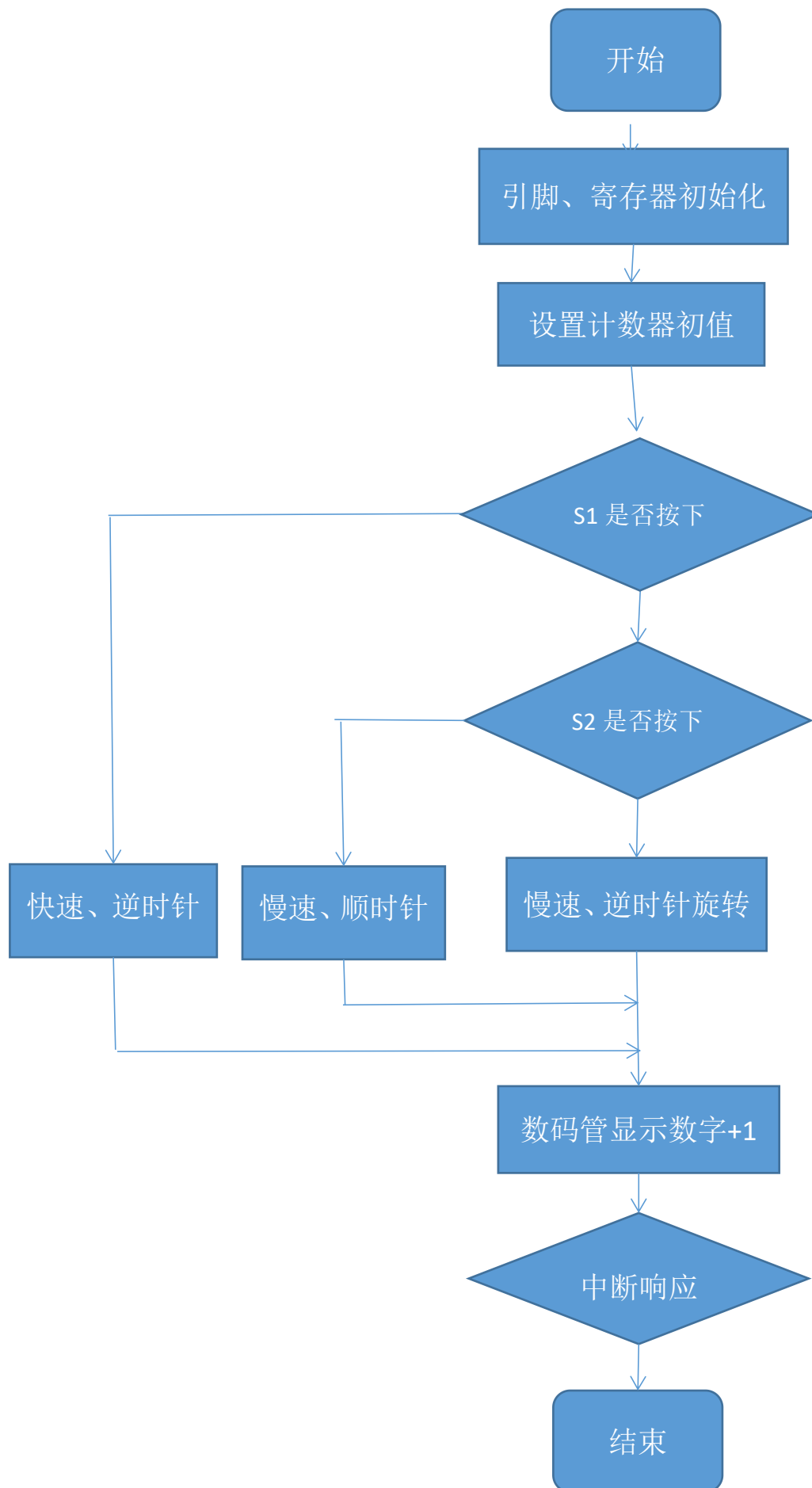
单片机系统的频率是 12M；步进电机转动一周需要 24 步。使用双四拍工作模式，将 CE1 和 CE2 分别置为高，然后 IN1 和 IN2 按照预定的脉冲输出，即 01->11->10->00->01 这个循环构成一个方向旋转的输出脉冲，将此序列翻转，就是相反方向的输出脉冲。

数码管显示，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。

### 三、实验内容

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。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。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、实验流程图



## 五、实验程序

```
ORG 0000H
LJMP START
ORG 000BH
LJMP T0IN;中断向量表
ORG 0040H
START;;初始化
    P4 EQU 0C0H
    MOV P4,#0FFH
    P4SW EQU 0BBH
    MOV P4SW,#70H
    DAT EQU P4.5
    CLK EQU P4.4
    SW1 EQU P3.6
    SW2 EQU P3.7
    CE1 EQU P1.1
    CE2 EQU P1.4
    IN1 EQU P3.2 ;01->11->10->00->01
    IN2 EQU P1.0 ;01->11->10->00->01
    SETB CE1
    SETB CE2
    CLR IN1
    CLR IN2
    MOV R6,#0
    MOV R5,#0
    MOV R4,#0
    MOV TMOD,#01H;和定时器/计数器相关的特殊功能寄存器，通过对它的设置
和读写就可以控制使用定时器/计数器
    MOV IE,#10000010B;中断允许寄存器
    MOV TMOD,#01H
    MOV TL0,#LOW(65536-41667) ;慢的情况 23869
    MOV TH0,#HIGH(65536-41667)
    SETB TR0;运行控制位，控制一个定时器是否允许计数
    SETB ET0;中断允许位，为 1 时允许中断
    SETB EA ;CPU 中断允许标志
MAIN:
    MOV A,R6
    MOV DPTR,#TAB
    MOVC A,@A+DPTR
    LCALL DISPLAY
    MOV A,R5
    MOV DPTR,#TAB
    MOVC A,@A+DPTR
```

```

    LCALL DISPLAY
    MOV A,R4
    MOV DPTR,#TAB
    MOVC A,@A+DPTR
    LCALL DISPLAY
    JB SW2,FAN
ZHENG:
    JNB IN1,Z1
    JNB IN2,Z2
    CPL IN1
    JMP BUJIA
Z1:
    JNB IN2,Z3
    CPL IN2
    JMP BUJIA
Z2:
    CPL IN2
    JMP BUJIA
Z3:
    CPL IN1
    JMP BUJIA
FAN:
    JNB IN1,F1
    JNB IN2,F2
    SETB IN1
    CLR IN2
    JMP BUJIA
F1:
    JNB IN2,F3
    SETB IN1
    SETB IN2
    JMP BUJIA
F2:
    CLR IN1
    CLR IN2
    JMP BUJIA
F3:
    CLR IN1
    SETB IN2
BUJIA:
    INC R6
    CJNE R6,#10,ZHONGDUAN
    INC R5
    MOV R6,#0

```

```

    CJNE R5,#10,ZHONGDUAN
    INC R4
    MOV R5,#0
    CJNE R4,#10,ZHONGDUAN
    MOV R4,#0
ZHONGDUAN:
    MOV R3,#6H;慢的情况下，需要计数 6 次才能满足时间要求
TT1:
    MOV R0,#0
DELAY:
    CJNE R0,#1,DELAY
    JNB SW1,OUT
    DJNZ R3,TT1
OUT:
    JMP MAIN
DISPLAY:
    MOV R0,#8
D1:
    CLR CLK
    RLC A
    MOV DAT,C
    SETB CLK
    DJNZ R0,D1
    RET
TOIN:
    MOV R0,#1
    MOV TL0,#LOW(65536-41667)
    MOV TH0,#HIGH(65536-41667)
    RETI
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，每次步进角度是多少，程序要如何修改？

答：每次步进角度为 15 度。A=IN1，B=IN2，(!A)表示 IN1=0,(!B) 表示 IN2=0 。输出脉冲为修改为：  
A->B->(!A)->(!B)->A。

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：每次步进角度为 7.5 度。输出脉冲修改为：A->AB->B->B(!A)->(!A)->!A!B->!B->(!B)A->A。

3. 步进电机的转速取决于那些因素？有没有上、下限？

答：步进电机的转动速度的控制是通过改变时钟的周期，即改变换相的频率来实现的；它转过的角度由输入的脉冲个数决定。当然，由于各种物理因素（包括摩擦、机械惯性、响应时间等），步进电机的最高转速有限制。

4. 如何改变步进电机的转向？

答：改变控制系统的方向电平信号。

5. 步进电机有那些规格参数，如何根据需要进行选择型号？

答：步进电机的主要参数有电机的最大工作电压、最小启动电压、最大允许功耗及工作频率等。

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 有五个独立的寻址空间。

①64K 字节程序存储器空间（0—0FFFFH）

②64K 外部数据存储器空间（0—0FFFFH）

③256 字节内部 RAM 空间（0—0FFH）

④256 位寻址空间（0—0FFH）

⑤工作寄存器区

这些寻址空间中，工作寄存器区重合在内部 RAM 的前 128 字节空间中，后 128 字节是内部特殊功能寄存器（SFR）空间，位寻址区的前 128 个地址重合在内部 RAM 中，后 128 个地址重合在 SFR 中的一部分寄存器中。MCS—51 系列中不同型号的单片机，特殊功能寄存器的定义和使用不完全一致。

**7. 说明 MOV C 指令的使用方法。**

答：MOV C 为单片机内部的 A 累加器向 ROM（程序存储器）读取数据，并不能向其中写入数据。

**8. MCS51 的指令时序是什么样的，哪类指令的执行时间较长？**

答：MCS51 汇编语言按指令执行时间分，有 64 条单周期指令、45 条双周期指令和 2 条 4 周期指令。

**9. 在本实验环境下，能否控制显示数码管的亮度？如何实现？**

答：可以控制，增大频率。

## 第四次单片机实验



## 一、实验题目：LED 点阵显示屏

## 二、实验原理

实验用的 LED 点阵显示屏为  $16 \times 16$  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。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。

## 三、实验内容

了解  $16 \times 16$  点阵电路的原理，编写汇编语言程序。编写一行汉字字符（至少三个字）的显示程序。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）。

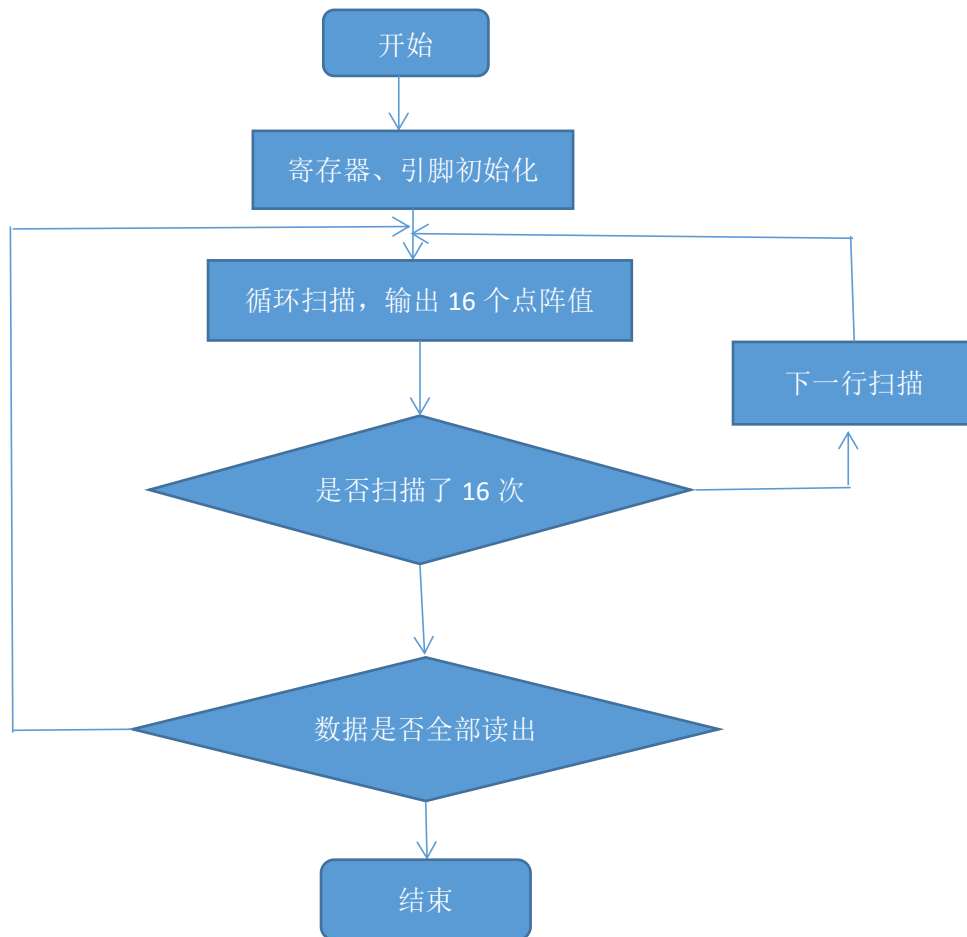
## 四、实验步骤

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；

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。

## 五、实验流程图



## 六、实验程序

```
ORG 000H
LJMP START
ORG 0040H
START:
    D_X EQU P0.0
    D_Y EQU P0.3
    CKX EQU P0.1
    CKY EQU P0.5
    CK_XL EQU P0.2
```

```
CK_YL EQU P0.6
EN_X EQU P0.7
EN_Y EQU P0.4
```

```
MOV R7,#0
```

```
;;.....
```

```
SM5:
```

```
MOV R2,#5
```

```
SM:
```

```
MOV R0,#0
```

```
MOV R1,#1
```

```
MOV R4, #0
```

```
MOV R5, #1
```

```
;;.....
```

```
MOV R3,#16
```

```
SM16:
```

```
SETB EN_X
```

```
;;.....
```

```
CLR CK_XL
```

```
MOV DPTR,#TABLE1
```

```
MOV A,R0
```

```
MOVC A,@A+DPTR
```

```
MOV R6,#8
```

```
YW1:
```

```
CLR CKX
```

```
RLC A
```

```
MOV D_X,C
```

```
SETB CKX
```

```
DJNZ R6,YW1
```

```
MOV A,R1
```

```
MOVC A,@A+DPTR
```

```
MOV R6,#8
```

```
YW0:
```

```
CLR CKX
```

```
RLC A
MOV D_X,C
SETB CKX
DJNZ R6,YW0
```

```
SETB CK_XL
```

```
CLR EN_X
```

```
LCALL DELAY
```

```
;.....
```

```
SETB EN_Y
CLR CK_YL
```

```
MOV DPTR,#TABLE
```

```
MOV A,R4
ADD A,R7
MOVC A,@A+DPTR
MOV R6,#8
```

YW3:

```
CLR CKY
RLC A
CPL C;取反
MOV D_Y,C
SETB CKY
DJNZ R6,YW3
```

```
MOV A,R5
ADD A,R7
MOVC A,@A+DPTR
MOV R6,#8
```

YW2:

```
CLR CKY
RLC A
CPL C
MOV D_Y,C
SETB CKY
DJNZ R6,YW2
```

```
SETB CK_YL
```

CLR EN\_Y

LCALL DELAY

;.....

INC R0

INC R0

INC R1

INC R1

INC R4

INC R4

INC R5

INC R5

DJNZ R3,SM16 ;16 行

DJNZ R2,SM

INC R7

INC R7

MOV A,R7

SUBB A,#224

JZ START

LJMP SM5 ;死循环

;.....

DELAY:

PUSH 02H

PUSH 06H

MOV R6,#150

DEL1:

MOV R2,#20

DEL2:

DJNZ R2,DEL2

DJNZ R6,DEL1

POP 06H

POP 02H

RET

;

TABLE:

DB 00H,80H,01H,00H,06H,00H,1FH,FFH,E0H,00H,02H,08H,04H,30H,18H,C0H;  
DB F0H,02H,10H,01H,13H,FEH,10H,00H,10H,80H,14H,60H,18H,18H,00H,00H;"你",0

DB 08H,02H,08H,44H,0FH,A8H,F8H,10H,08H,68H,0FH,86H,00H,00H,01H,00H;  
DB 41H,02H,41H,01H,47H,FEH,49H,00H,51H,00H,61H,00H,01H,00H,00H,00H;"好",1

DB 04H,00H,04H,00H,04H,00H,7FH,FEH,04H,02H,04H,02H,FFH,E2H,04H,22H;  
DB 04H,22H,04H,22H,FFH,E2H,04H,02H,04H,02H,04H,02H,04H,00H,00H,00H;"世",2

DB 00H,10H,00H,10H,00H,20H,7FH,21H,49H,46H,49H,78H,49H,80H,7FH,00H;  
DB 49H,80H,49H,7FH,49H,40H,7FH,20H,00H,20H,00H,10H,00H,10H,00H,00H;"界",3

DB 00H,80H,01H,00H,06H,00H,1FH,FFH,E0H,00H,02H,08H,04H,30H,18H,C0H;  
DB F0H,02H,10H,01H,13H,FEH,10H,00H,10H,80H,14H,60H,18H,18H,00H,00H;"你",0

DB 08H,02H,08H,44H,0FH,A8H,F8H,10H,08H,68H,0FH,86H,00H,00H,01H,00H;  
DB 41H,02H,41H,01H,47H,FEH,49H,00H,51H,00H,61H,00H,01H,00H,00H,00H;"好",1

DB 04H,00H,04H,00H,04H,00H,7FH,FEH,04H,02H,04H,02H,FFH,E2H,04H,22H;  
DB 04H,22H,04H,22H,FFH,E2H,04H,02H,04H,02H,04H,02H,04H,00H,00H,00H;"世",2

DB 00H,10H,00H,10H,00H,20H,7FH,21H,49H,46H,49H,78H,49H,80H,7FH,00H;  
DB 49H,80H,49H,7FH,49H,40H,7FH,20H,00H,20H,00H,10H,00H,10H,00H,00H;"界",3

TABLE1:

DB 80H,00H  
DB 40H,00H  
DB 20H,00H  
DB 10H,00H  
DB 08H,00H  
DB 04H,00H  
DB 02H,00H  
DB 01H,00H

DB 00H,80H  
DB 00H,40H  
DB 00H,20H  
DB 00H,10H

```
DB 00H,08H
DB 00H,04H
DB 00H,02H
DB 00H,01H
```

END

## 七、思考题

**1、如何使用软件调整和控制 LED 点阵的亮度？**

答：增大系统的频率。

**2、如何尽量避免显示过程中的闪烁？**

答：减慢扫描速度，可以通过定时延时的方法。

**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：在行、列上分别设置 8、160 个 74HC595 来达到多行多列的效果。

## 八、两次实验的反思与领会

在第一次步进电机实验中，我基本了解了步进电机的工作原理、初步掌握了 MCS-51 的体系结构和汇编语言，熟悉 Keil 编程环境和程序下载工具的使用方法，还有对定时器和计数器的操作使用。

而在第二次 LED 点阵实验中，我了解 LED 点阵显示的基本原理和实现方法，初步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。

实验过程不是一帆风顺，很多时候会找不到方向，

但经过查找资料和与同学讨论基本上可以解决问题。  
两个实验受益匪浅，但今后仍需努力，争取学习更多的知识。