

实验三 步进电机原理及应用

53160821 2016 级计科 8 班 刘洋洋

一、实验目的和要求

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

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。

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。

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

$$\text{定时周期} = 12/\text{CPU 晶振频率}$$

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）。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。

三、实验器材

单片机测控实验系统
步进电机控制实验模块
Keil 开发环境
STC-ISP 程序下载工具

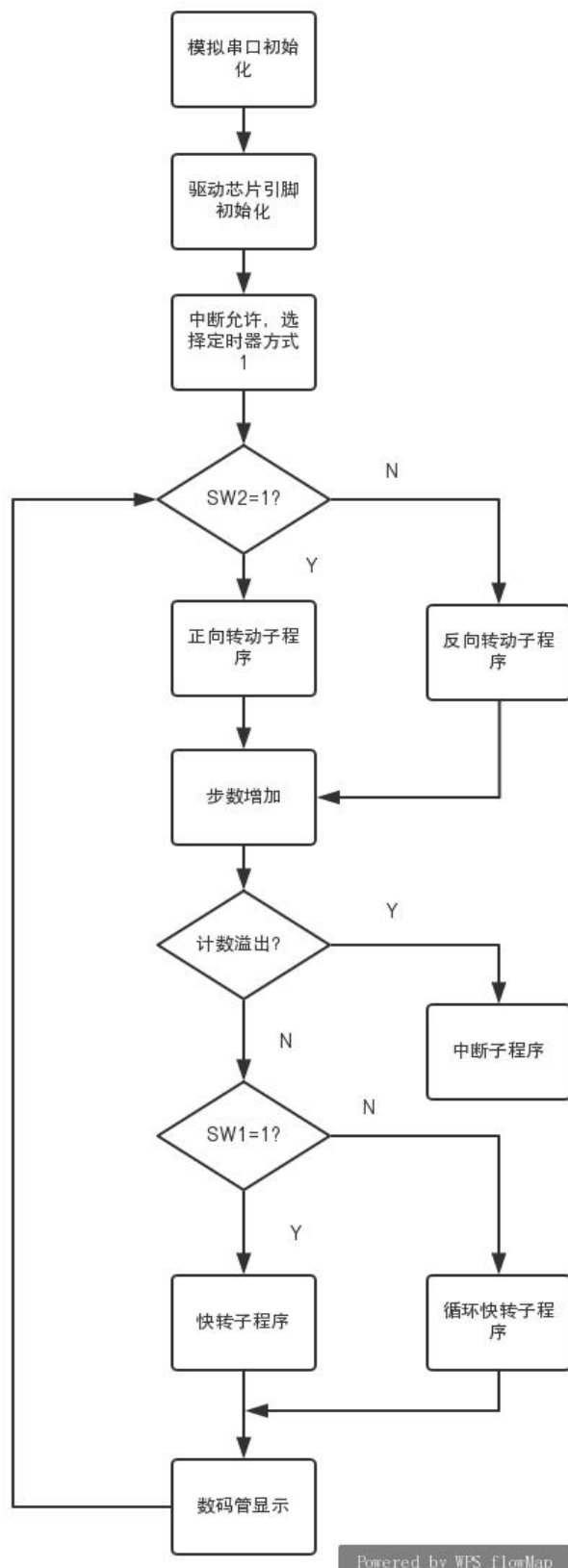
四、实验内容

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。

五、程序流程图



六、实验步骤

- 1 预习
- 2 简单程序录入和调试
- 3 程序调试

七、实验代码

```
ORG 0000H
    LJMP START
ORG 000BH
    LJMP T_INT
ORG 0040H
START:
    P4 EQU 0C0H
    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
    IN2 EQU P1.0

    SETB CE1
    SETB CE2
    CLR IN1
    CLR IN2

    MOV R6, #0
    MOV R5, #0
    MOV R4, #0

    MOV TMOD, #01H
    ;MOV IE, #10000010B
    MOV TLO, #3DH
    MOV TH0, #5DH
    SETB TRO
    SETB ETO
    SETB EA
```

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
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

T_INT:
        MOV R0, #1
        MOV TLO, #3DH
        MOV TH0, #5DH
        RETI

```

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

八、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？
15 度。输出脉冲修改为：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？
7.5 度。输出脉冲修改为：A→AB→B→B(!A)→(!A)→!A!B→!B→(!B)A
3. 步进电机的转速取决于那些因素？有没有上、下限？
时钟的周期控制，通过改变输入脉冲的个数决定转过的角度；转速有上限，通过加大控制电压和降低线圈的时间常数可以提高上限；转速无下限。
4. 如何改变步进电机的转向？
通过反向 IN1 和 IN2 的输入
5. 步进电机有那些规格参数，如何根据需要进行选择型号？
最大工作电压、最小启动电压、最大允许功耗和工作频率等。
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？
 - (1) 工作寄存器组 (00H——1FH)
 - (2) 可位寻址 RAM 区 (20H——2FH)
 - (3) 通用的 RAM 区 (30H——7FH)
7. 说明 MOV C 指令的使用方法。
MOV C 用来读取程序存储器；以 16 位的程序计数器 PC 或数据指针 DPTR 作为基寄存器，以 8 位的累加器 A 作为变址寄存器，基址寄存器和变址寄存器的内容相加作为 16 位的地址访问程序存储器。如：
MOV C A, @A+PC
MOV C A, @A+DPTR
8. MCS51 的指令时序是什么样的，哪类指令的执行时间较长？
一个机器周期包含 6 个状态 (S1-S4)，每个状态分为两个节拍 P1 和 P2，通常，一个机器周期会出现两次高电平 S1P2 和 S4P2，每次持续一个状态 S。乘法及除法指令占 4 个周期，三字节指令均为双周期指令。
9. 在本实验环境下，能否控制显示数码的亮度？如何实现？
能，通过修改刷新频率

实验四 LED 点阵显示屏

一、实验目的和要求

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

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。

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。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。

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。

先送出第 1 行控制发光管亮灭的数据并锁存，然后选通第 1 行使其点亮一段时间；然后送出第 2 行控制发光管亮灭的数据并锁存，然后选通第 2 行使其点亮相同时间……第 16 行之后又重头点亮第 1 行，反复轮回，速度够快，由于人眼的暂留效果，就能看到屏上稳定的字。

每次循环显示完 16 行即一屏之后，再从第 1 行开始显示时，直接从上次开始的列数的下一列开始取值，比如上次从第 1 列开始显示，则循环完 16 行后，再从第 2 列开始显示……。这样就能形成字符滚动显示的现象。

三、实验设备

- 1、单片机测控实验系统
- 2、LED 点阵显示器实验模块
- 3、Keil 开发环境
- 4、STC-ISP 程序下载工具

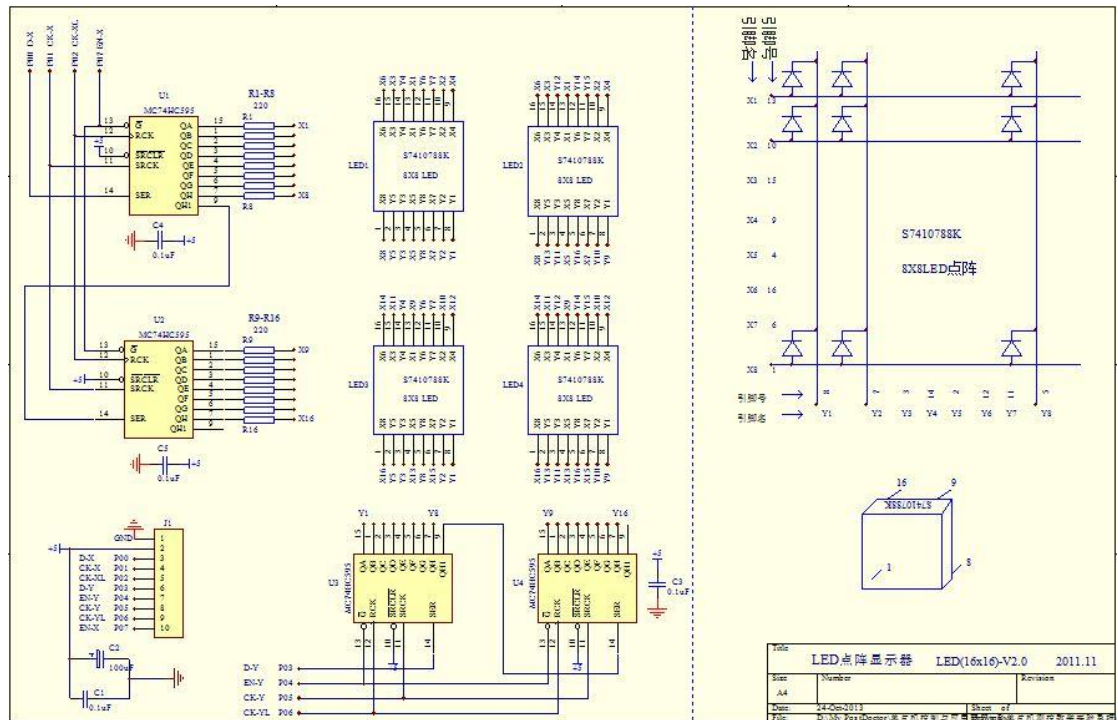
四、实验内容

了解 16*16 点阵电路的原理，编写汇编语言程序。

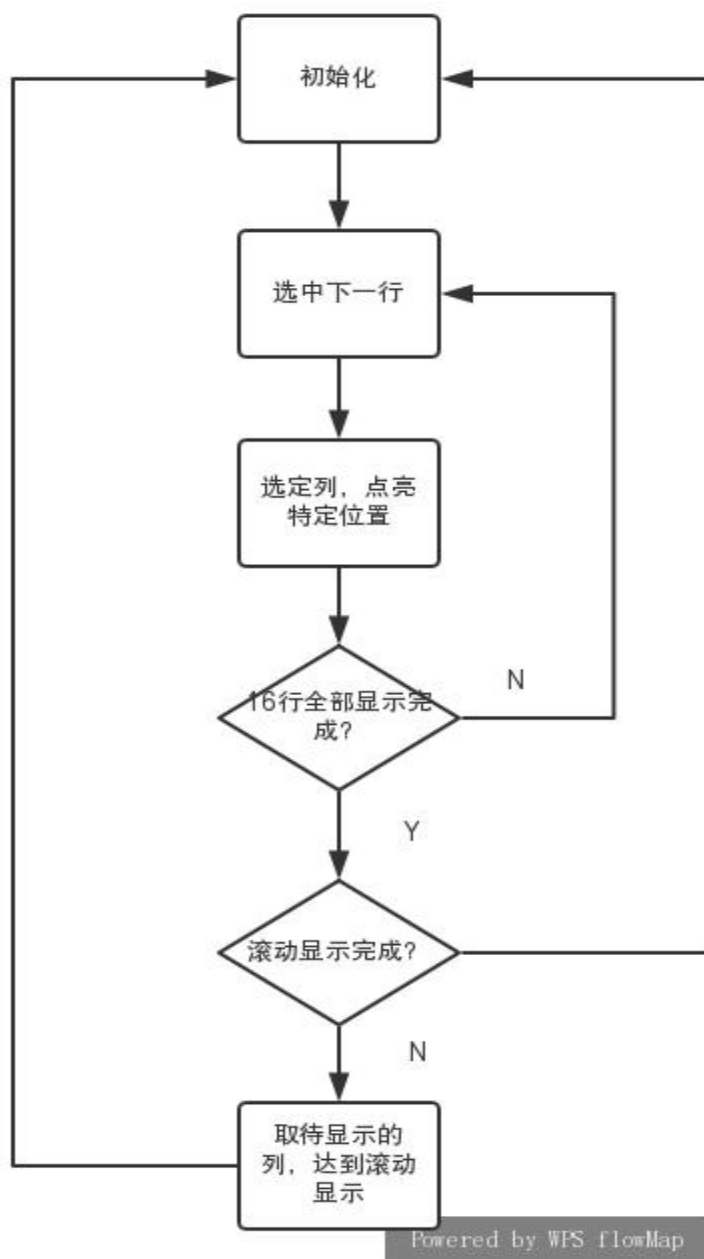
编写一行汉字字符（至少三个字）的显示程序。

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)。
本实验原理图如下：



五、程序流程图



六、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。

七、实验代码

```

D_Y EQU P0.0;列
CK_Y EQU P0.1

```

```
CK_YL    EQU P0.2
D_X      EQU P0.3
EN_X     EQU P0.4
CK_X     EQU P0.5
CK_XL    EQU P0.6
EN_Y     EQU P0.7
```

```
        ORG 00H
        LJMP  START
        ORG 40H
START:
        CLR CK_X
        CLR CK_XL
        CLR CK_Y
        CLR CK_YL
        SETB EN_X
        SETB EN_Y
        MOV DPTR, #TA
        MOV R7, #0
LOOP:
        MOV A, #0
        MOV R0, #0
        MOV R1, #0
        MOV R5, #255

LOOP_0:
        MOV A, R0
        ADD A, R7
        CJNE A, #224, L0
L0:
        JC L1
        SUBB A, #224
L1:
        MOVC A, @A+DPTR
        MOV R2, A
        INC R0
        MOV A, R0
        ADD A, R7
        CJNE A, #224, L2
L2:
        JC L3
        SUBB A, #224
L3:
        MOVC A, @A+DPTR
```

```

MOV R3, A
INC R0

MOV A, R3
MOV R4, #8
Y1:
    RRC A
    MOV D_Y, C
    SETB    CK_Y
    NOP
    CLR CK_Y
    DJNZ    R4, Y1

MOV A, R2
MOV R4, #8
Y2:
    RRC A
    MOV D_Y, C
    SETB    CK_Y
    NOP
    CLR CK_Y
    DJNZ    R4, Y2

SETB    CK_YL

CJNE    R1, #0, LOOP1
ACALL   OUTDX
MOV R1, #1
LJMP    LOOP2
LOOP1:
    SETB    D_X
    SETB    CK_X
    NOP
    CLR CK_X

    SETB    CK_XL

LOOP2:
    CLR CK_XL
    CLR CK_YL
    CLR EN_X
    CLR EN_Y
    ACALL   DELAY
    SETB    EN_X

```

```

        SETB     EN_Y

        MOV  A, #0
        MOV  R4, #8
C1:
        RRC  A
        MOV  D_Y, C
        SETB     CK_Y
        NOP
        CLR  CK_Y
        DJNZ   R4, C1

        MOV  A, #0
        MOV  R4, #8
C2:
        RRC  A
        MOV  D_Y, C
        SETB     CK_Y
        NOP
        CLR  CK_Y
        DJNZ   R4, C2

        SETB     CK_YL
        NOP
        CLR  CK_YL
        CLR  CK_XL
        CLR  EN_X
        CLR  EN_Y
        ACALL  DELAY
        SETB     EN_X
        SETB     EN_Y

        CJNE   R0, #32, LOOP3
        MOV  R0, #0
        MOV  R1, #0
LOOP3:
        DJNZ   R5, LOOP5
        INC  R7
        INC  R7
        CJNE   R7, #224, LOOP4
        MOV  R7, #0
LOOP4:
        LJMP   LOOP
LOOP5:

```

LJMP LOOP_0

OUTDX:

X0:

MOV A, #255

MOV R4, #8

X1:

RLC A

MOV D_X, C

SETB CK_X

NOP

CLR CK_X

DJNZ R4, X1

MOV A, #254

MOV R4, #8

X2:

RLC A

MOV D_X, C

SETB CK_X

NOP

CLR CK_X

DJNZ R4, X2

SETB CK_XL

RET

DELAY:

MOV R6, #255

DE1: INC R6

DEC R6

DJNZ R6, DE1

RET

TA:

DB 00H, 00H, 07H, F0H, 08H, 08H, 10H, 04H, 10H, 04H, 08H, 08H, 07H, F0H, 00H, 00H; "0", 0

DB 00H, 00H, 0EH, 38H, 11H, 44H, 10H, 84H, 10H, 84H, 11H, 44H, 0EH, 38H, 00H, 00H; "8", 1

DB 00H, 00H, 0EH, 0CH, 10H, 14H, 10H, 24H, 10H, 44H, 10H, 84H, 0FH, 0CH, 00H, 00H; "2", 2

DB 00H, 00H, 00H, 00H, 08H, 04H, 08H, 04H, 1FH, FCH, 00H, 04H, 00H, 04H, 00H, 00H; "1", 3

DB 10H, 02H, 14H, 04H, 12H, 08H, 91H, 90H, 70H, 60H, 11H, 98H, 1EH, 06H, 10H, 00H;

DB 10H, 00H, 00H, 00H, 1FH, F0H, 00H, 02H, 00H, 01H, FFH, FEH, 00H, 00H, 00H, 00H; "刘", 4

DB 08H, 20H, 06H, 20H, 40H, 7EH, 31H, 80H, 00H, 00H, 08H, 20H, 89H, 20H, 69H, 20H;
DB 09H, 20H, 0FH, FFH, 09H, 20H, 29H, 20H, C9H, 20H, 08H, 20H, 00H, 20H, 00H, 00H; "洋", 5

DB 08H, 20H, 06H, 20H, 40H, 7EH, 31H, 80H, 00H, 00H, 08H, 20H, 89H, 20H, 69H, 20H;
DB 09H, 20H, 0FH, FFH, 09H, 20H, 29H, 20H, C9H, 20H, 08H, 20H, 00H, 20H, 00H, 00H; "洋", 6

END

八、思考题

1. 如何使用软件调整和控制 LED 点阵的亮度

答：可以通过控制行显示延时调整亮度。延时越短，扫描频率越快，LED 点阵越亮。

2. 如何尽量避免显示过程中的闪烁

答：增加每一屏显示次数，一般刷新频率提高到 24hz 以上。

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：硬件方面可以通过添加新的 led 以及 74hc59 来实现，软件方面将控制行扫描的 16 位数字 0ffffH 改为 64 位的 0fffffffffffffeH 将读入列值的 2 字节改为 160 字节，及重复输出 1280bit, 结束后令行的输出移位一次。