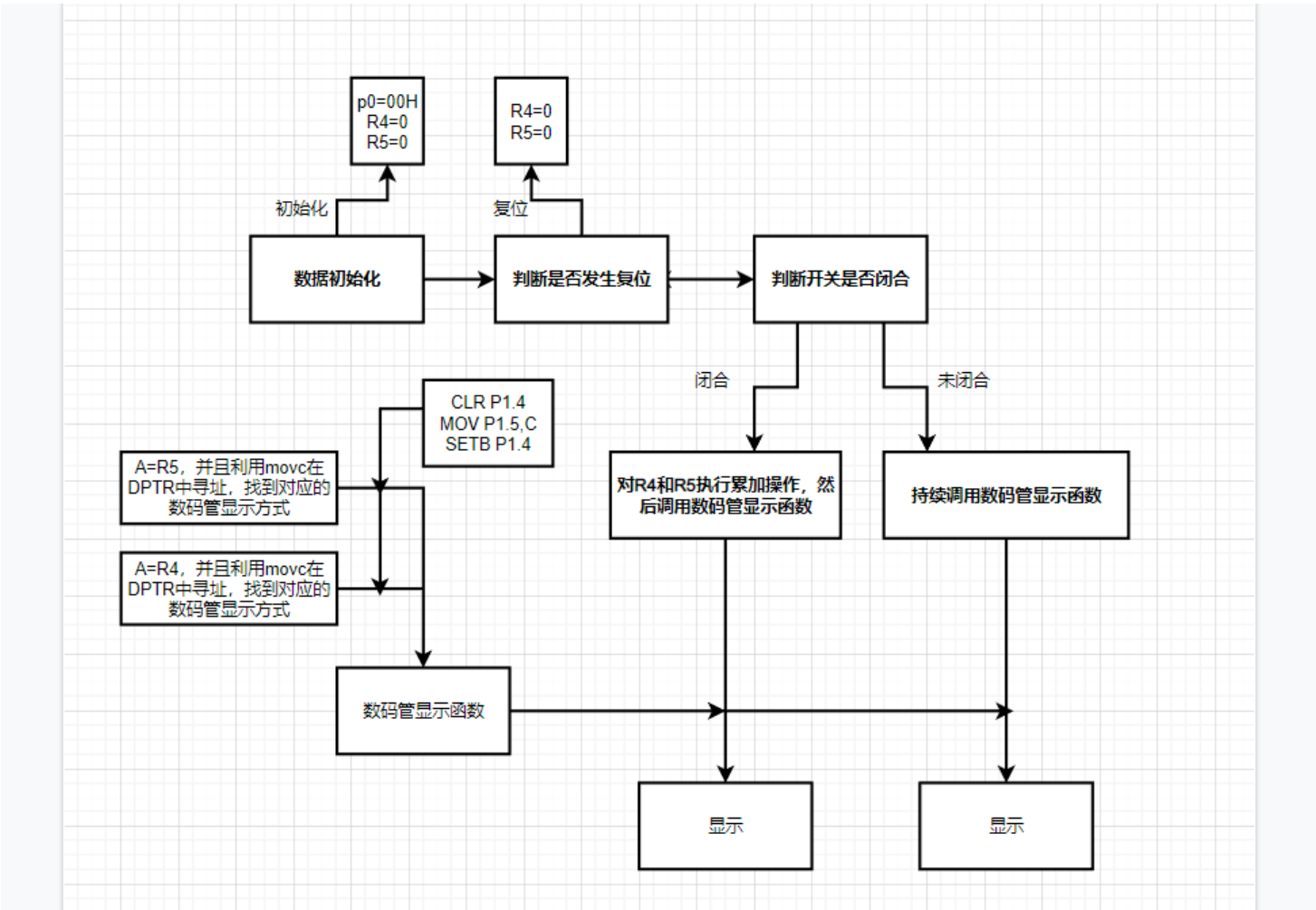


# 一、原理总结

利用两个寄存器 **R4** 和 **R5** 来存储两个数码管的显示数字，**R4** 用来存储前一个数码管显示数字，而 **R5** 用来存储后一个数码管显示数字，利用左移操作 **RLC** 取 **A** 中首位放入 **C** 中，然后将 **C** 输入到数码管中，使用 **R0** 来控制循环，对于 **R4** 和 **R5** 重复上述过程各自重复8次，使得当数码管中每位都有数据时，数码管便会显示，之后使用停顿函数使数码管上数字停留一段时间。

# 二、程序分析

图示部分：



代码部分：

```

; Reset Vector
    org    0000h
    jmp    Start

;=====
; CODE SEGMENT
;=====

    org    0100h
TAB : DB  3FH,06H,5BH,4FH,66H,6DH,7DH,07H,7FH,6FH,77H,7CH,39H,5EH,79H,71H    ;是用来存储0-15这16个数据，之后使用

PAUSE: ;停顿函数      ;R1是第一层嵌套循环,R2是第二层嵌套循环,当R2不为0时跳转，为0时不跳转，执行DJNZ R1, D1使得R1- -,当
        MOV R1,#255 ;R1是第一层嵌套循环
D1:      MOV R2,#255      ;R2是第二层嵌套循环
D2:      DJNZ R2,D2      ;当R2不为0时跳转，为0时不跳转，执行DJNZ R1, D1使得R1- -
        DJNZ R1,D1      ;当R1不为0时跳转，为0时不跳转
RET

LED:     ;数码管亮灯函数
        MOV R0,#8      ;将8赋值给R0，8位左移
        MOV A,R5        ;R5代表右边的LED的数
        MOVC A,@A+DPTR  ;A是偏移地址，是从DPTR中寻址

LOOP1:
        RLC A           ;A左移,将A中高位放入C中
        CLR P1.4        ;传输C到数码管中
        MOV P1.5,C
        SETB P1.4
        DJNZ R0,LOOP1    ;若R0为0，也即是执行完8次循环，不跳转，否则跳转

        MOV R0,#8      ;和上面同理
        MOV A,R4
        MOVC A,@A+DPTR

LOOP2:
        RLC A
        CLR P1.4
        MOV P1.5,C
        SETB P1.4
        DJNZ R0,LOOP2

        MOV R6,#0      ;使用停顿函数，构成3层循环，使停顿时间延迟
S1:      LCALL PAUSE
        INC R6
        CJNE R6,#10,S1  ;循环次数为10次

RET

START:
        MOV p0,A        ;将p0置为0，使上面的数码管不显示

```

```

MOV DPTR,#TAB    ;将TAB首地址赋给DPTR
MOV R4,#0        ;赋初值为0
MOV R5,#0        ;赋初值为0
JMP XUNHUAN

A1:      MOV C,P1.0    ;当开关不闭合时，持续显示一个数字
        JC XUNHUAN
        MOV C,9        ;当9号接口传输数据时，复位
        JC FUWEI
        INC R5          ;R5++
        CJNE R5,#16,XUNHUAN ;当R5为16时要对R4++，而R5<16时跳转
        MOV R5,#0
        INC R4
        CJNE R4,#16,XUNHUAN ;当R4为16时重新执行START，而R5<16时跳转
        JMP FUWEI

XUNHUAN:
        LCALL LED
        JMP A1

FUWEI:
        MOV R4,#0
        MOV R5,#0
        JMP A1

JMP START
;=====
END

```

### 三、思考题

1. 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？

MCS51可存取单元分为三种：

- 1、片内外统编址的64K的程序存储器地址空间
- 2、256B的片内数据存储器的地址空间
- 3、64K 片外数据存储器的地址空间

在访问三个不同的逻辑空间时，应采用不同形式的指令以访问不同的存储器空间的选通信号。程序代码存放于程序存储器中，使用MOVC指令访问。数据存储器是随机存取方式，访问片内RAM用MOV指令，访问片外RAM用MOVX指令，用于存放程序执行的中间结果和过程数

2. 说明MOVC指令的使用方法。

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。

3. MCS51的指令时序是什么样的，哪类指令的执行时间较长？

MCS -51的时序单位有四个，分别是节拍、状态、机器周期和指令周期。

MCS51指令系统中，按指令的长度和执行所需的机器周期分为六种时序：

单字节指令单机器周期。单字节指令双机器周期

双字节指令单机器周期。双字节指令双机器周期

三字节指令双机器周期。单字节指令四机器周期

机器周期越长，执行时间越长。所以其中需要四个机器周期的指令执行时间最长。

1. 在本实验中，能否控制显示数码的亮度？如何实现

可通过更改延时子程序内容或增加调用延时子程序次数，来控制通电时间的长短进而控制显示数码管的亮度。若通电时间缩短，则亮度降低。即扫描速度越快，亮度越低。#### 四、问题分析

## 一、遇到问题：

1. 不理解为什么使用 **RLC** 而非使用 **RRC** ；
2. 在不进行初始化p0的话，会导致上面的数码管显示一个 “8” ；

## 二、解决方法：

1. 是由于本题图中的数码管是反接的，而又由于高位要对应高位，所以需要使用 **RLC** 来使之高位对高位，低位对低位
2. 是由于器件本身的性质决定，51单片机本身对p0是输出高电平的，所以会导致上面的数码管显示一个 “8”，所以在初始化时要赋值为0