

第6章 接口技术

6.1 LED显示器接口

6.2 键盘接口

6.3 SPI总线及应用

6.4 A/D、D/A转换器接口

6.5 功率驱动器件接口

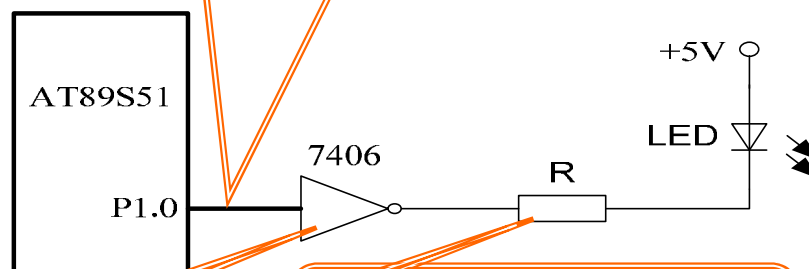
6.1 LED 显示接口技术

LED显示器用于显示工业控制参数、过程状

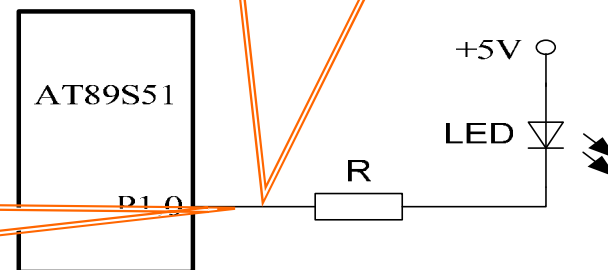
6.1.1 独立LED与单片机的接口

独立LED显示器多用于信号指示，它实际是一个压降为1.5~2.5V，电流为5~10mA的发光二极管，通过LED的电流决定它的发光强度。

输出高电平点亮
LED



输出低电平点亮LED



增大驱动能力

限流电阻怎么计算？

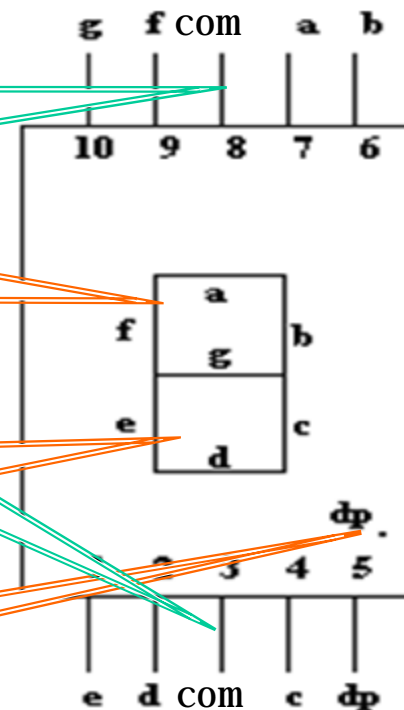
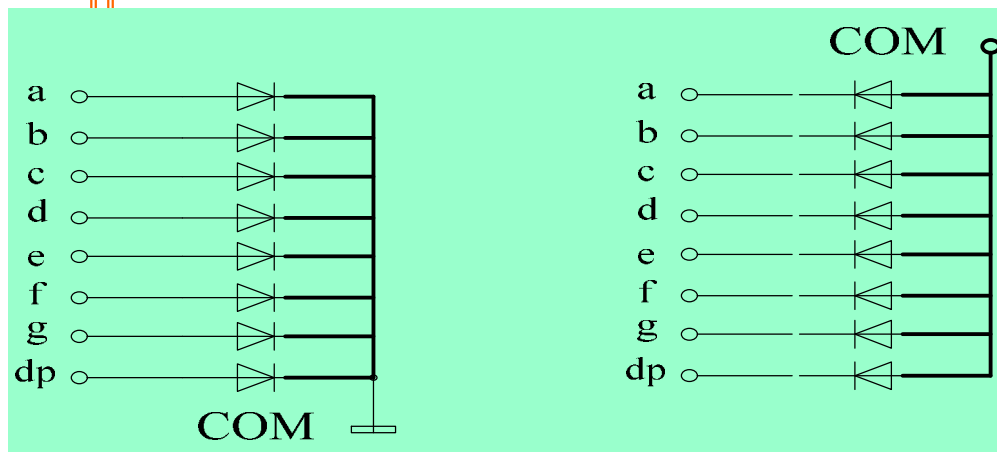
直接驱动
20mA，整个芯片100mA。

6.1.2 LED与单片机接口

1、LED数码管接口技术

八段LED的公共端。公共端为LED阴极叫共阴数码管；公共端为阳极叫共阳数码管。

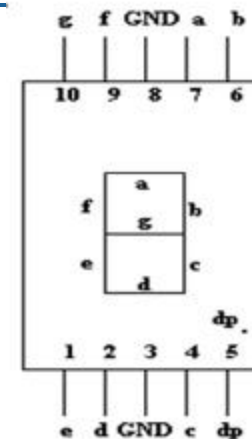
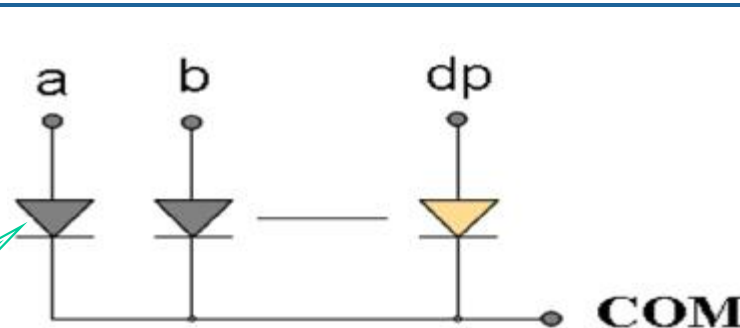
种数字、字母以及其他符





共阴与共阳LED段码

D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0
 dp g f e d c b a



共阴数码管对应字段
输出高电平点亮。共

共阳极接法的段选码与
共阴极接法的段选码是
逻辑“非”关系。

LED 的字形码 (字段码)

显示字符	共 阴 极	共 阳 极
0	3FH	C0H
1	06H	F9H
8.	FFH	00H
‘灭’	00	FFH
F	71H	84H
U	3EH	C1H
P.	F3H	02H

显示程序任务：

1) 设置显示缓冲区，存放待显示数据和字符（位置码）。

2) 显示译码：程序存储器中建立字形码常数表，查表得出对应数据和字符的字形码。

3) 输出显示：输出字形码到显示端口。

例：

```

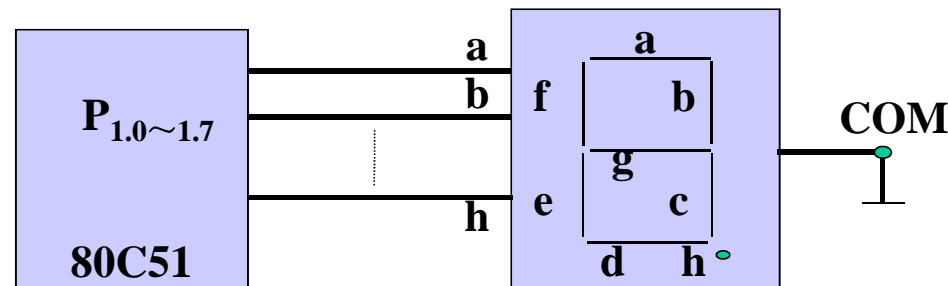
MOV DPTR, #DTAB; 指向字形码表首地址
MOV A, @R0      ; 取显示缓冲区中数据
MOVC A, @A+DPTR; 查表显示译码
MOV P1, A       ; 输出显示
  
```

...

```

DTAB:  DB  3FH, 06H, 5BH ; 字形码表
  
```

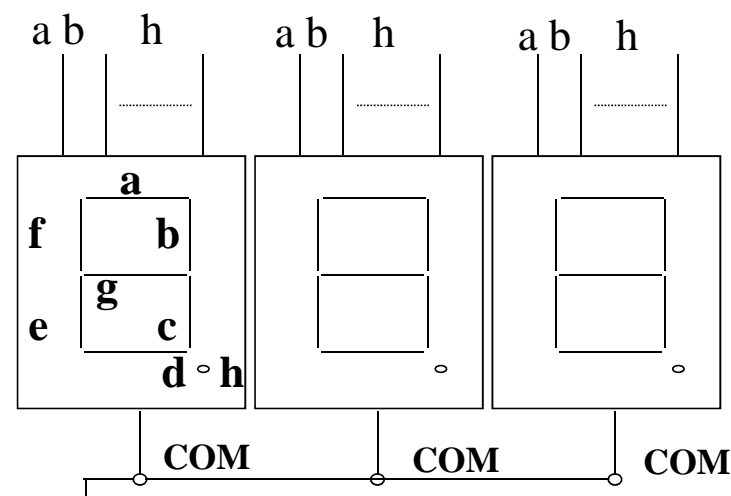
...



2、LED数码管与单片机的接口

1) 静态显示接口

原理简单，显示亮度强，无闪烁，每个LED需要一个8位并行口，占用I/O接口数量多。

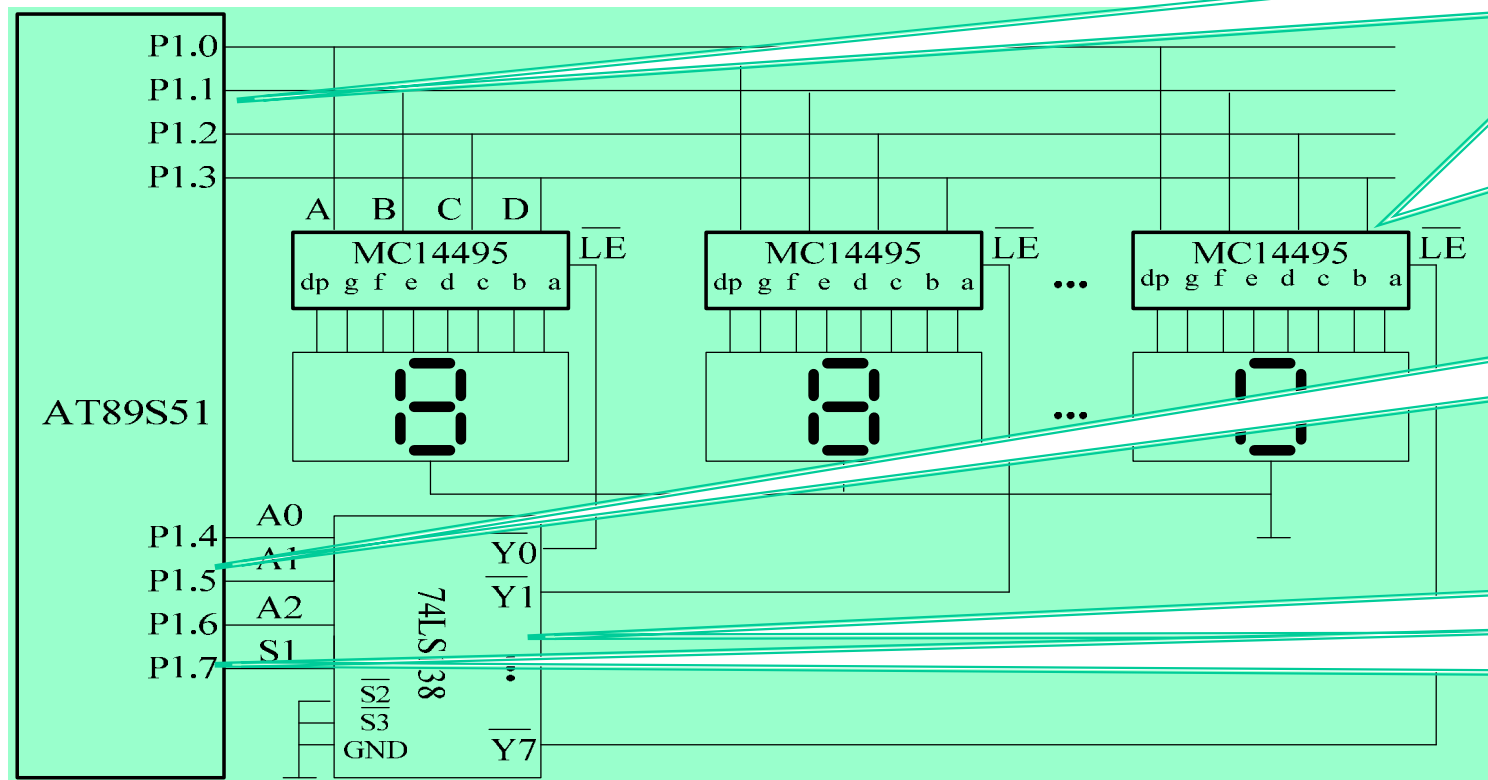


06 第六章 MCS-51单片机接口技术



静态显示接口电路：

输入BCD码 输出7段
P1口低4位输出BCD码。



程序例：
若要在最左端的LED
显示字符0，则P1口
输出10000000B
=80H。

```
MOV A, 80H
MOV P1, A
```

2) LED动态显示原理与接口

多位LED共用一个8位字段口，各位LED公共端用字位口控制，扫描输出显示不同字形。

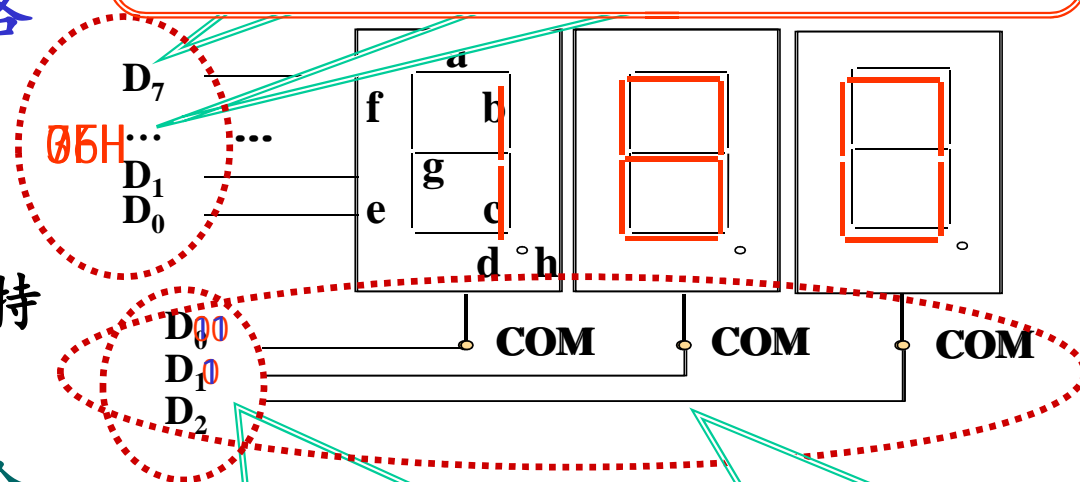
特点：与静态显示相比有什么特点？
程序复杂。

方法：扫描输出，每位数码管轮流导通显示。

原理：视觉暂留效应。

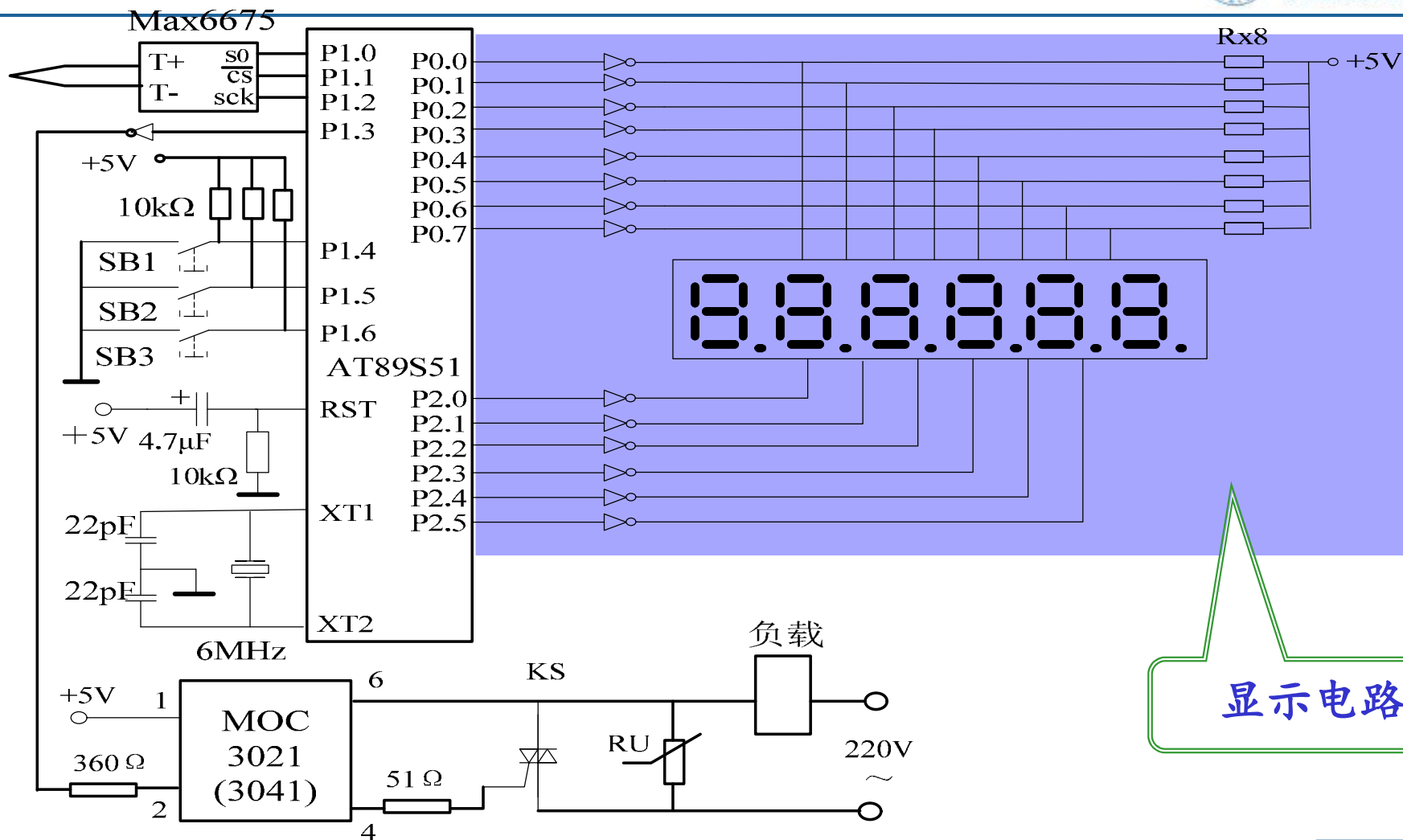
快速挥动火把的体验

怎样才能做到看上去是同时显示的？



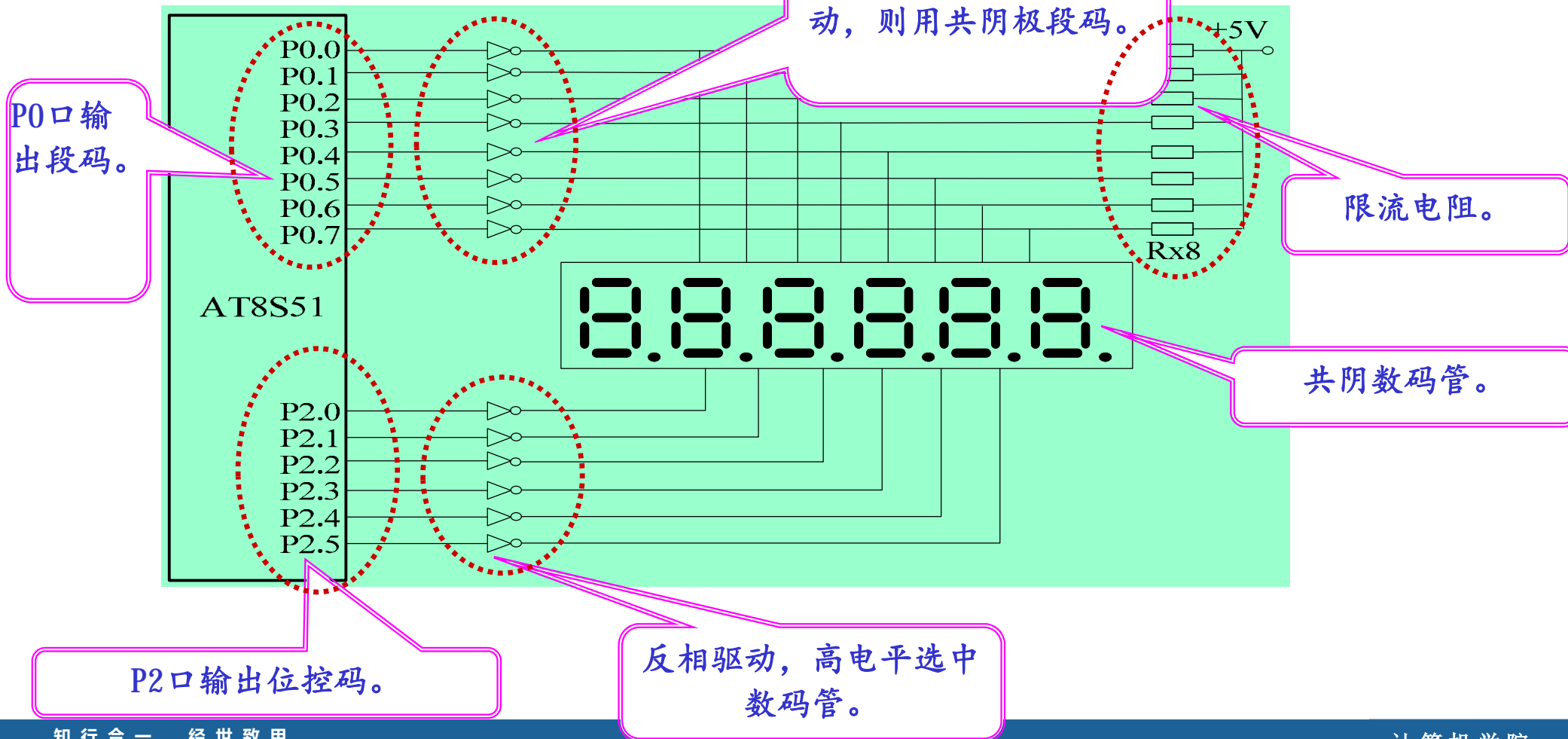
要求：轮流显示的速度要足够快，即每秒20次以上。

烘箱温度控制系统

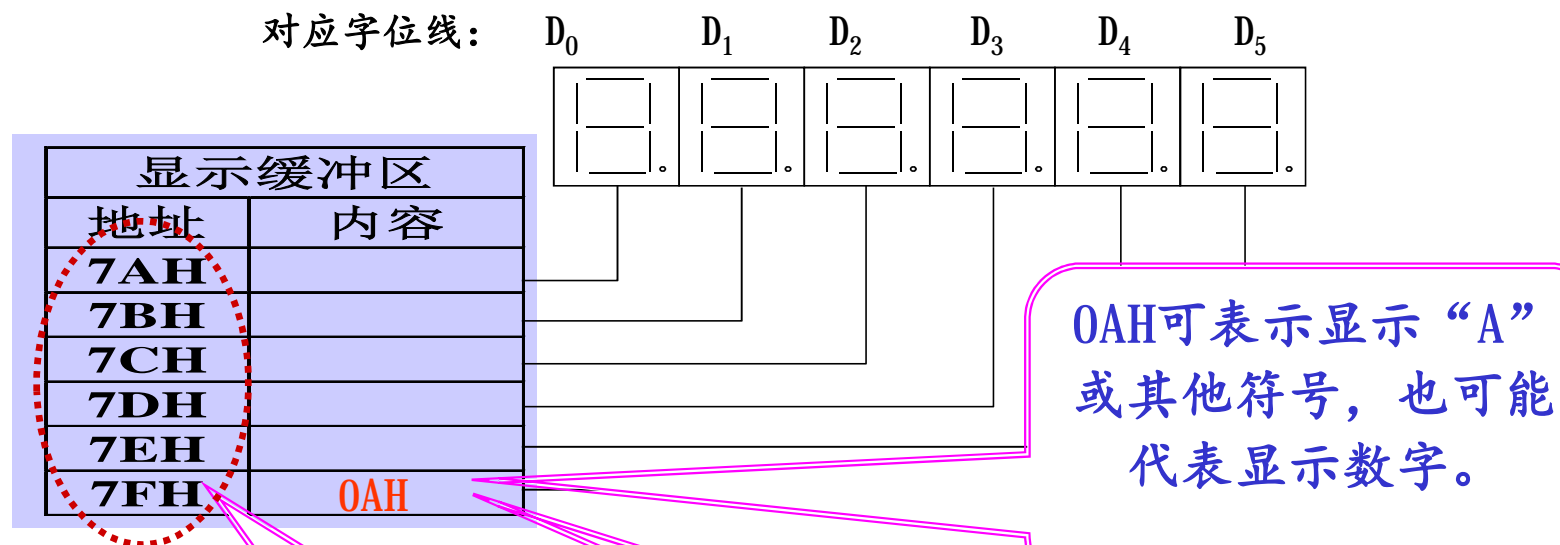


显示电路

烘箱控制系统的动态显示接口：



显示缓冲区与多位LED的对应关系：



如7AH、7BH、7CH可分别放置键盘输入给定温度值的百位、十位、个位；7DH、7EH、7FH同样输入实际温度值的百位、十位、个位。

BCD数、或显示字符在段码表的位置代码。

DIS: MOV R0, #0AH ;指向显示缓冲区起始单元

MOV R3, #0FEH ;字位码初值→R3

DLP: MOV P2, #0FFH ;熄灭所有LED

将显示缓冲区7AH—7FH的内容送至图

中数码管上刷新显示一遍。

ADD A, #16 ;查表偏移量

MOV P0, A ;输出字形码

MOV A, R3 ;取字位码

MOV P0.1, A ;输出字位码, 显示其中1位

LCALL DELAY ;延时1ms

INC R0 ;指向显示缓冲区下一单元

MOV A, R3 ;修改字位码

RL A ;显示下一位

MOV R3, A ;未显示完, 继续显示

JB ACC.6, DLP ;全部扫描一遍,

RET ;返回

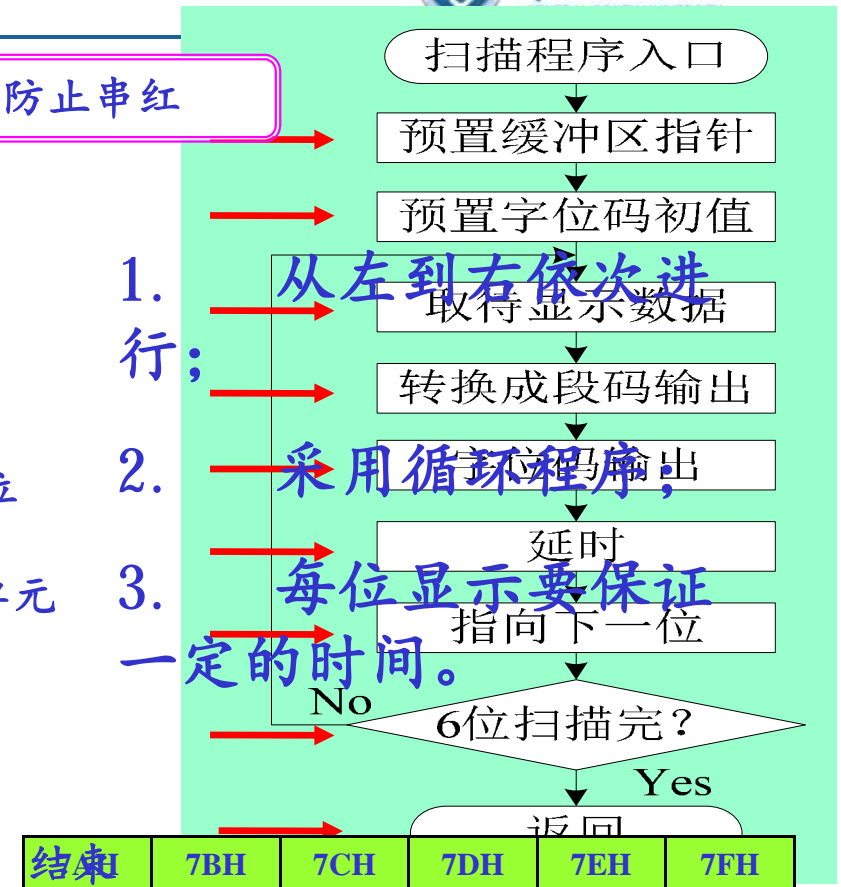
DTAB: DB 0C0H, 0F9H, 0A4H ;字形表

DB 0B0H, 99H, 92H, 82H, 0F8H

DB 80H, 90H, 0FFH, 0BFH

防止串红

1. 从左到右依次进行;
2. 采用循环程序;
3. 每位显示要保证一定的时间。

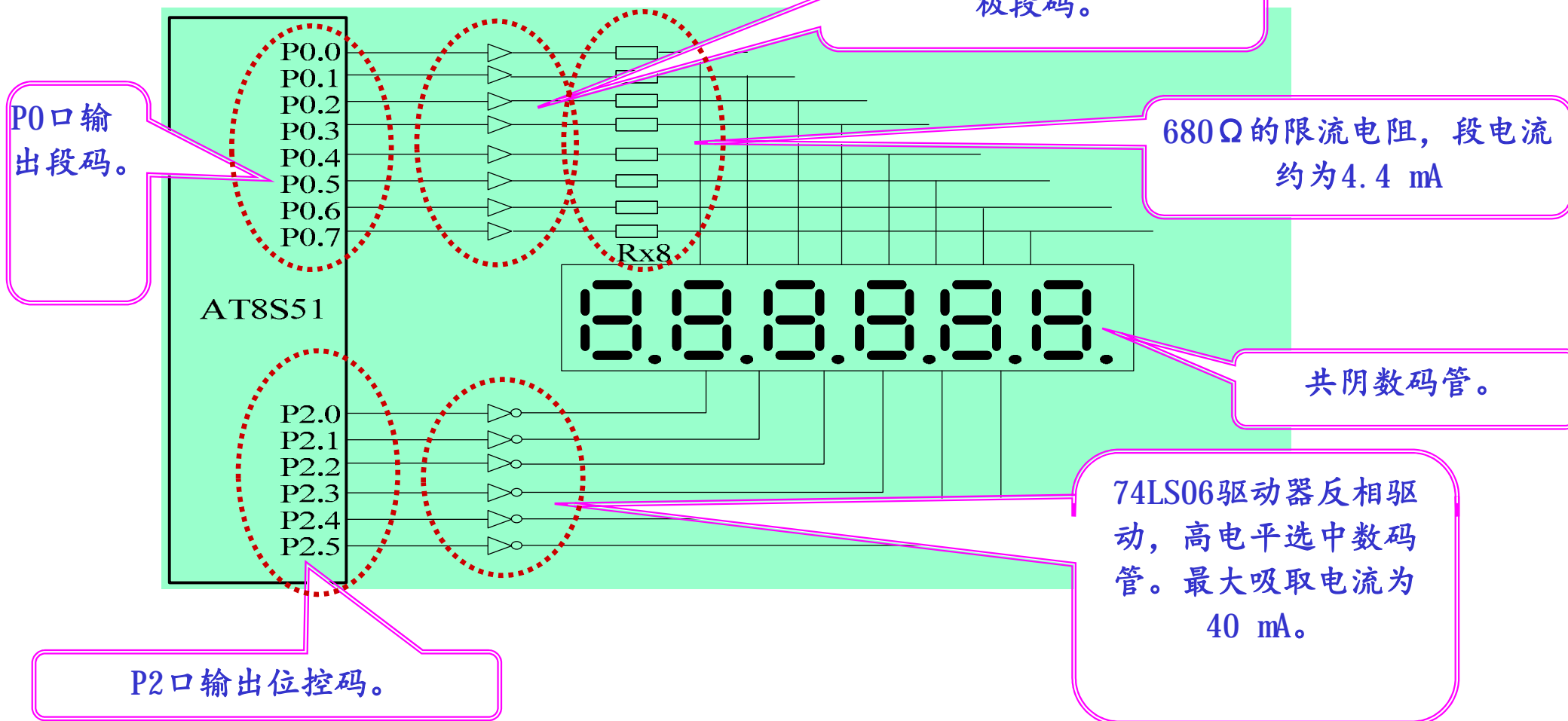


教学案例：烘箱温度控制系统温度显示子系统设计

任务：烘箱温度控制系统的温度测量范围是 $0\sim 250^{\circ}\text{C}$ ，要求同时显示温度设定值和温度测量值，显示分辨率为 1°C 。

方案：温度设定值和温度测量值各需要3位数码显示，一共6位。采用6个LED数码管动态扫描方式来作为系统显示。

烘箱控制系统的动态显示接口

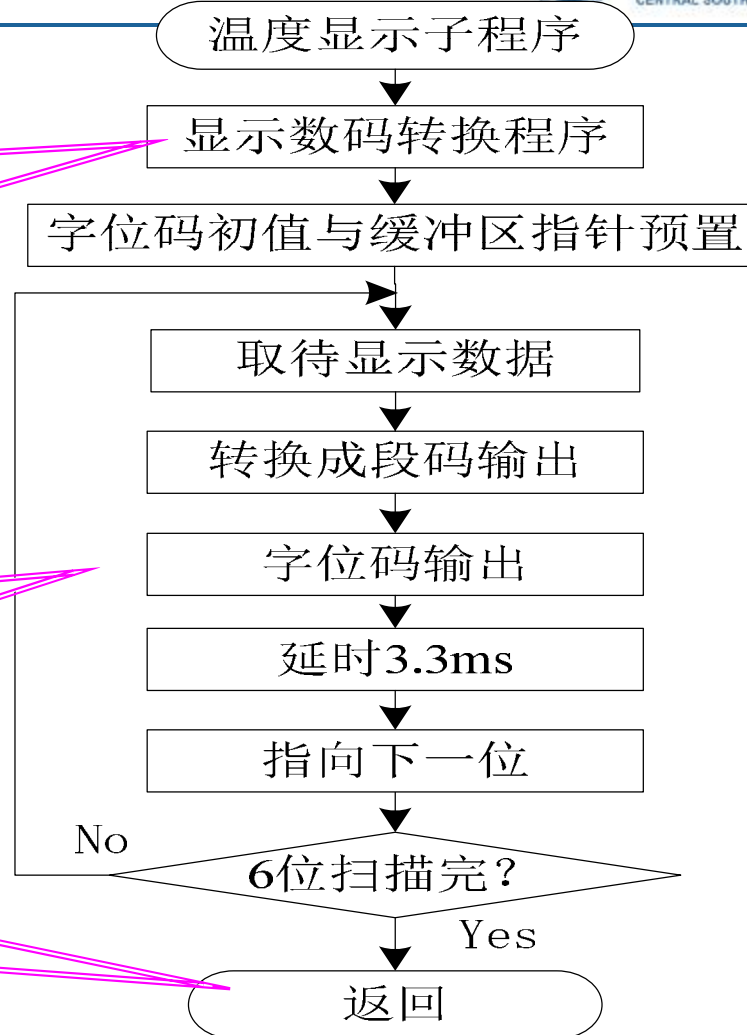


显示程序流程图

将待显示的内容（温度设定值和温度测量值）拆分转换成显示数码存放至显示缓冲区

每循环一次输出显示一位LED数据

刷新显示一遍需要20ms左右的时间



程序名: DISPLAY

功能: 将温度设定值与温度测量值送至数码管上刷新显示一遍, 程序运行时间若20mS。

占用资源: 累加器A, 状态寄存器PSW, 数据指针DPTR, 通用寄存器00组 (R0、R5、R6、R7)

显示缓冲区: DISBUF (3AH~3FH)

初始化部分

```
DISBUF      EQU      3AH
DISPLAY:    LCALL  CODECON      ; 调用拆分转换程序
            MOV     R0, #DISBUF  ; 设置显示缓冲区首地址
            MOV     R7, #20H     ; 设置初始位选码
            MOV     DPTR, #TABDIS ; 指向字型表首址
```



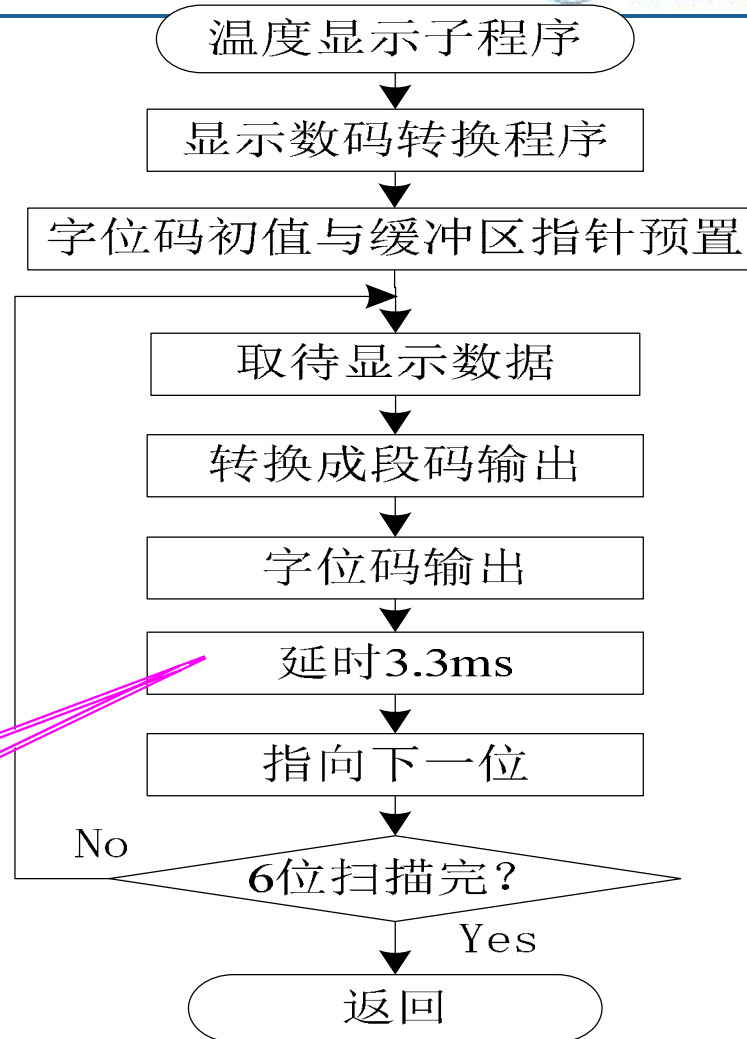
```

DIS1:    MOV     A, @R0 ; 取显示数据
          MOVC    A, @A+DPTR; 转换成相应的段选码
          MOV     P0, A      ; 段选码输出
          MOV     P1, R7     ; 位选码输出
DIS2:    MOV     R6, #6
DIS3:    MOV     R5, #250
          DJNZ    R5, $
          DJNZ    R6, DIS3   ; 延时3.3ms
          INC     R0
          MOV     A, R7
          RR      A          ; 指向下一位位置
          MOV     R7, A
          CJNE    A, #80H, DIS1 ; 未完继续
          RET
TABDIS:  DB      3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H
          DB      7FH, 6FH, 77H, 7CH, 39H, 5EH, 79H, 71H, 00H
  
```

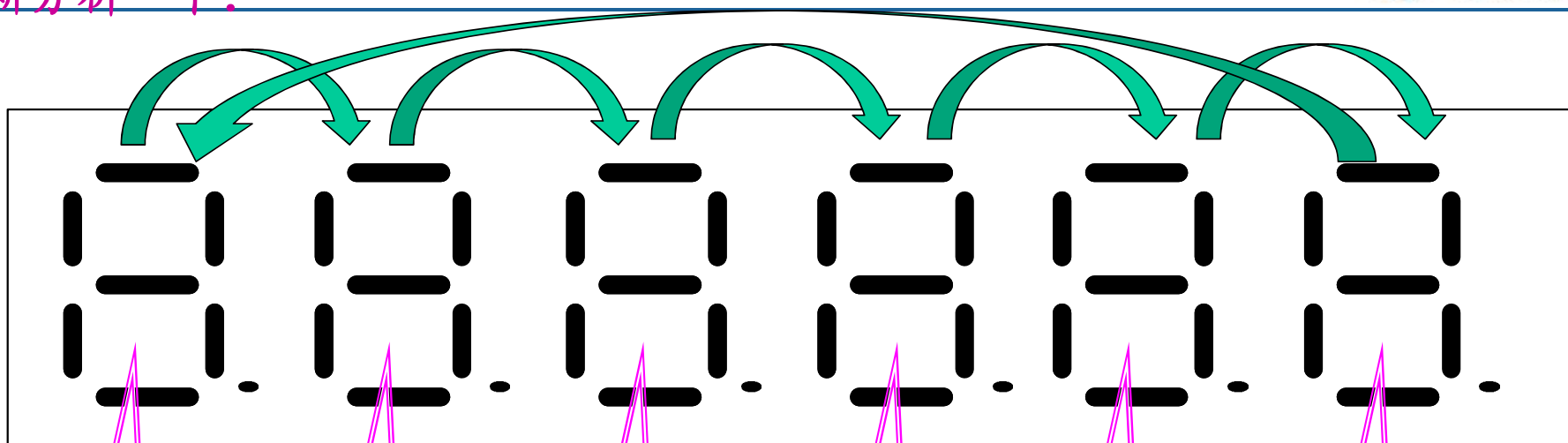
功能：将温度设定值与温度测量值送至数码管上刷新显示一遍，程序运行时间若20mS。

如何避免这么长的时间？

时间白白浪费了！



重新分析一下:



结论: 6个数码管轮流显示, 每次点亮一个数码管, 延时3.3ms点亮下一个数码管, 周而复始。

送出显示,
3.3ms

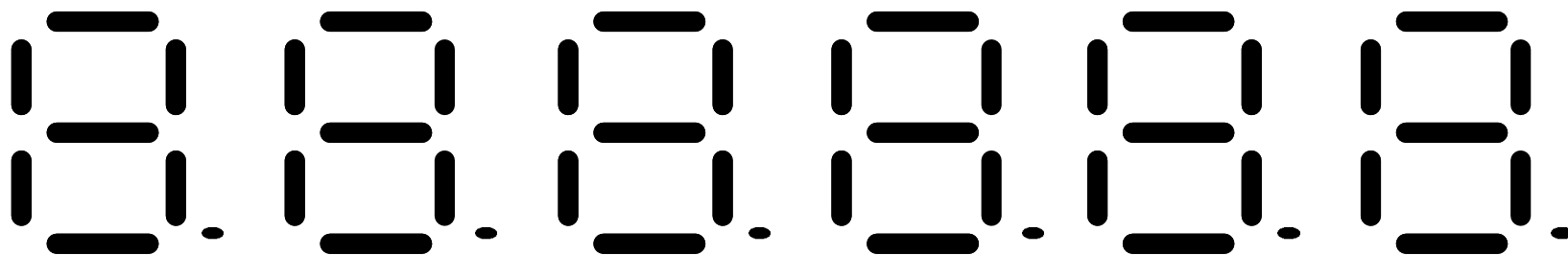
送出显示,
3.3ms

送出显示,
3.3ms

送出显示,
3.3ms

送出显示,
3.3ms

送出显示, 延时
3.3ms



显示程序功能:

将当前数码管的内容送出显示,
并且指向下一个数码管。

程序执行要求:

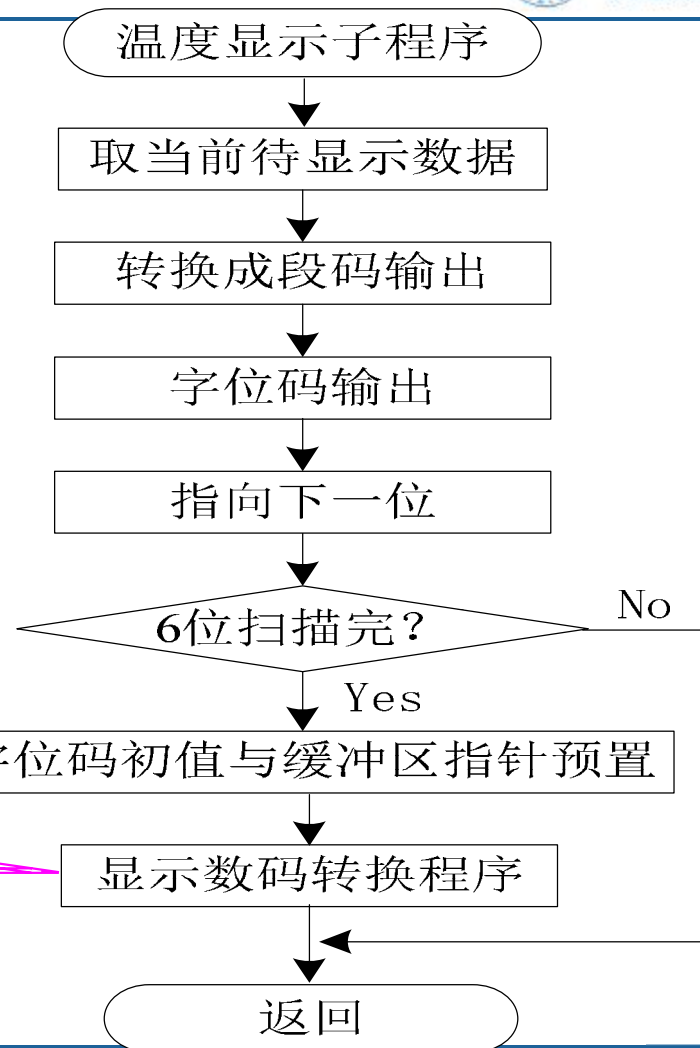
每隔3ms左右调用执行一次。

显示程序功能：

将显示缓冲区当前数码管的内容送出显示，并且指向下一个数码管。每隔3ms左右调用执行一次。显示一遍后刷新显示缓冲区。

该流程为标准的显示模块程序

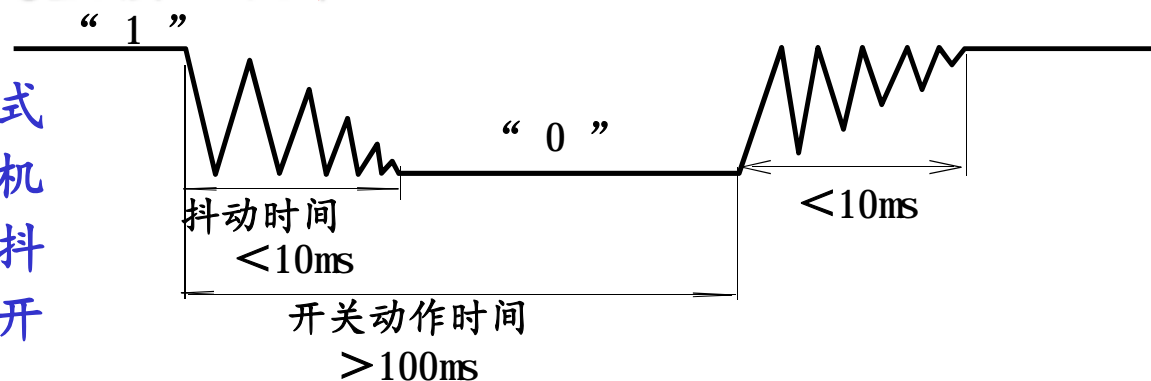
必须在主程序初始化部分调用执行一次



6.2 键盘接口技术

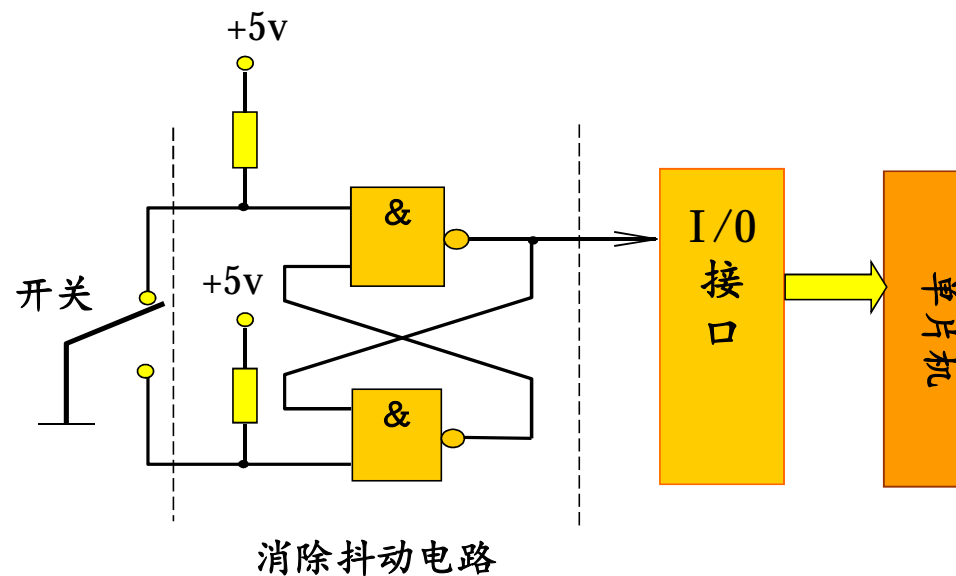
按键结构与特点

由于弹性作用的影响，机械式按键按下或释放时通常有一定时间的机械抖动，然后其触点才稳定下来。其抖动过程如图所示，抖动时间的长短与开关的机械特性有关，一般为5~10 ms。



有硬件电路消除抖动和
软件消除抖动两种方法。

软件方案——延时10ms~20ms
后再次判断



键处理程序任务

键输入：检查键盘是否有键被按下，消除按键抖动。

键译码：确定被按键的键号，获取键号。键号为键盘位置码，根据键号查表得出被按键的键值。键值：数字键0~9、字符键0AH~0FH、功能键10H~ 。

键处理：根据键值转移到不同程序段。若键值属于数字、字符键，则调用显示数字和字符的子程序。若键值属于功能键，则进行多分支转移，执行各个功能程序段。

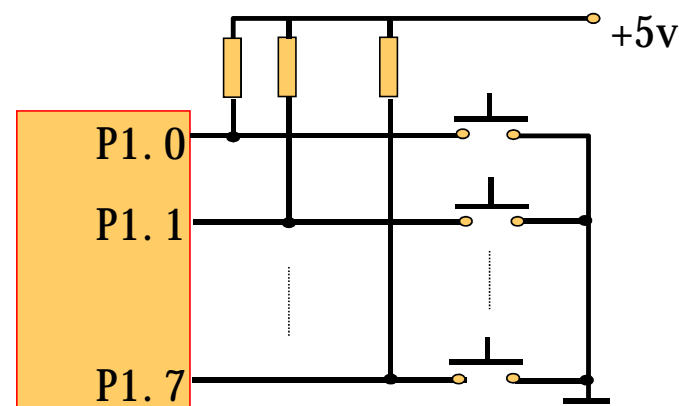
6.2.1 独立式键盘电路

特点:

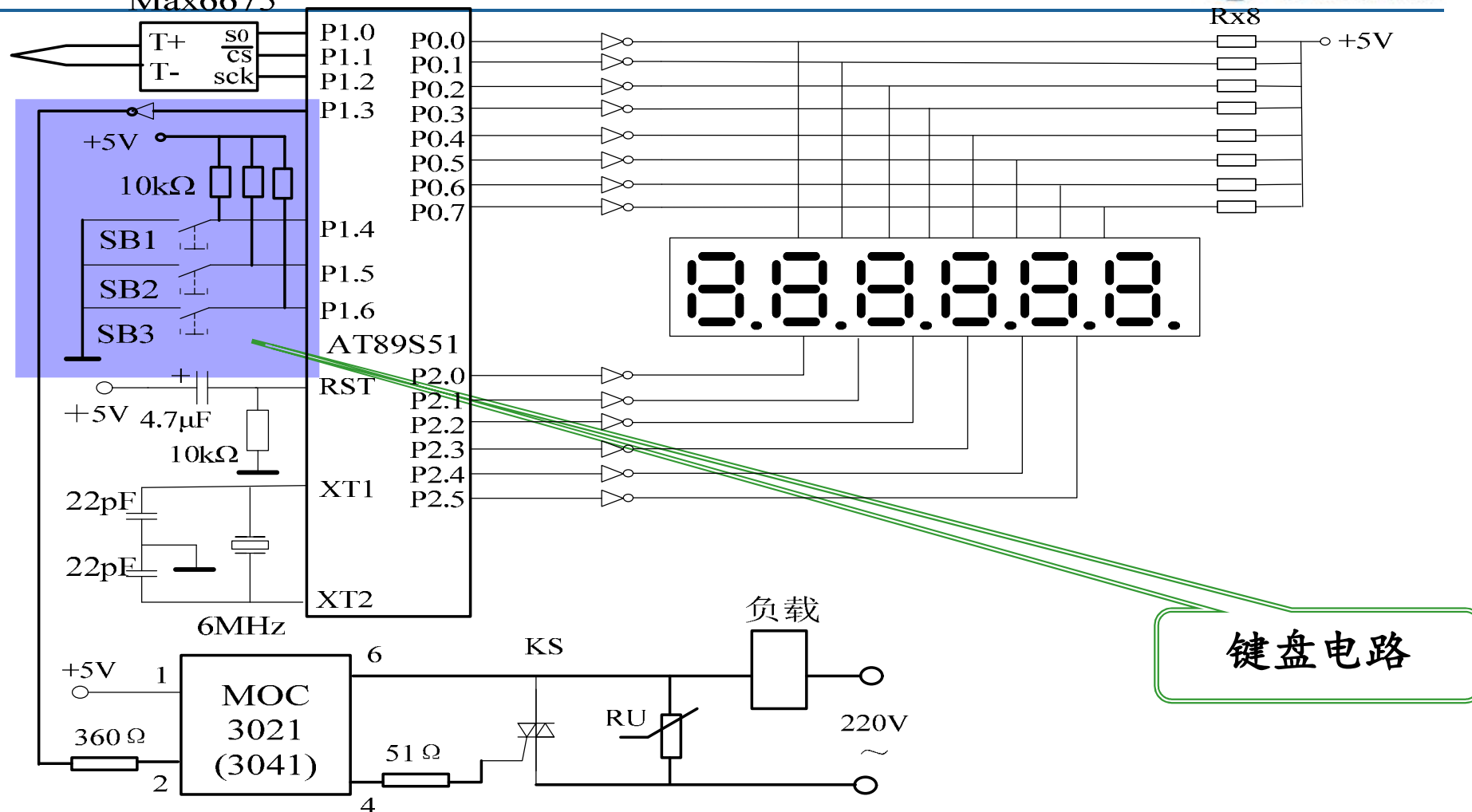
每个按键单独占有一根I/O接口引线,
I/O口线浪费大。

配置灵活, 结构简单, 编程容易。

独立式按键的软件常采用查询式结构。



烘箱控制系统



独立按键编程举例：（设30H单元是温度给定值）

P3.2所接按键每按下一次，则将30H单元内容加1。 P3.3所接按键每按下一次，则将30H单元内容减1。

```
JCL:   JB     P3.2 , JCL2   ; 键未按下则转移
        LCALL  D20MS        ; 延时去抖
        INC    30H
JCL1:   JNB    P3.2 , JCL1   ; 等待按键释放
        LCALL  D20MS
JCL2:   JB     P3.3 , JCLR   ; 键未按下则退出返回
        LCALL  D20MS        ; 延时去抖
        DEC    30H
JCL1:   JNB    P3.3 , JCL2   ; 等待按键释放
        LCALL  D20MS
JCLR:   RET                  ; 返回主程序
```

6.2.2 行列式键盘

1、特点：

按键设置在行线与列线交叉点上，按键数量多时占用I/O口线少。

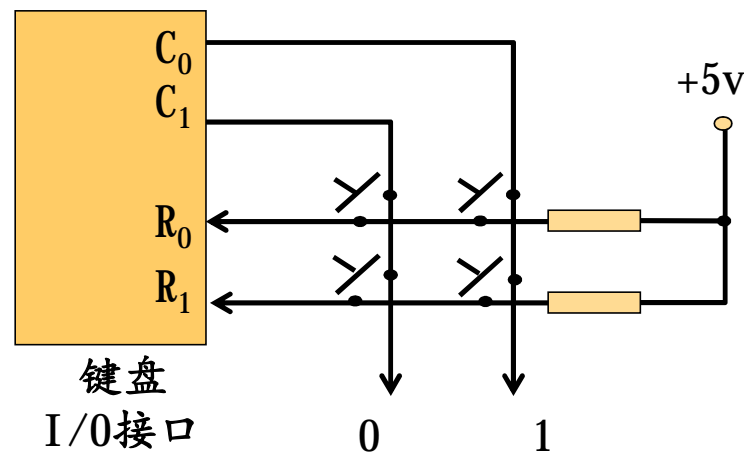
编程复杂。

2、键盘工作方式：

1) 扫描法

列线输出，行线输入。

列线逐行输出0，某行有按键，行线输入有0，若无按键，行线输入全部为1。



扫描法编程:

原理:

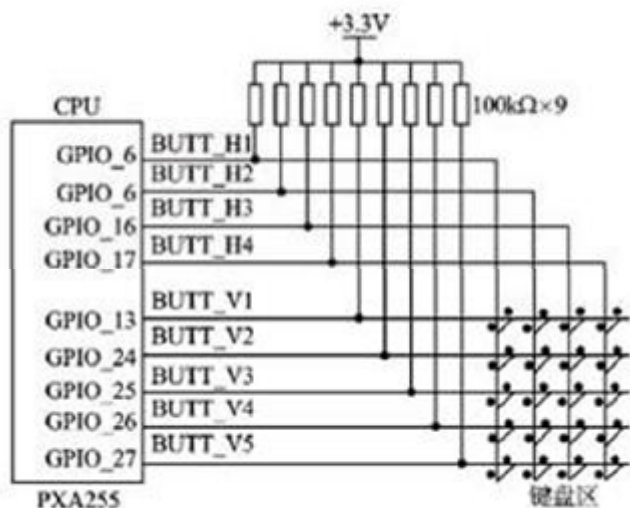
在某一时刻只让一条列线处于低电平，其余列线均处于高电平，则当这一列有键按下时，该键所在的行电平将会由高电平变为低电平，可判定该列相应的行有键按下。

流程:

当第0列处于低电平时，逐行查找是否有行线变低，若有，则第0列与该行的交叉点按键按下；若无，则表示第0列无键按下，再让下一列处在低电平，依此循环，这种方式称为键盘扫描。

2) 反转法

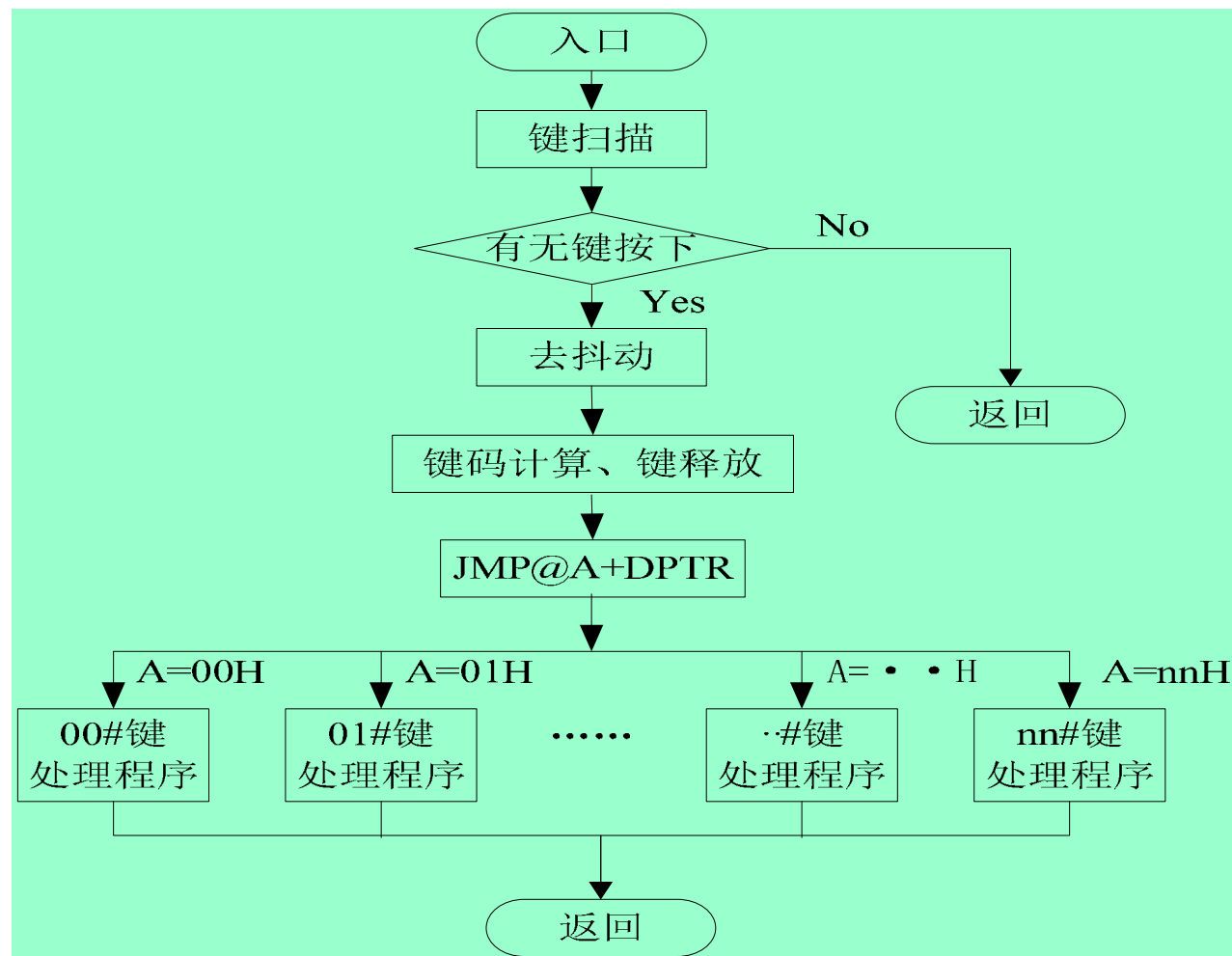
- ① 行列线交换输入、输出，两步获取按键键码值。
- ② 查表，根据按键键码值得到相应的键号。



Step 1: 将列线作为输出线，行线作为输入线。置输出线全部为0，此时行线中呈低电平0的为按键所在行，如果全部都不是0，则没有按键按下。

Step 2: 将第一步反过来，即将行线作为输出线，列线作为输入线。置输出线全部为0，此时列线呈低电平的为按键所在的列。这样，就可以确定了按键的位置 (X, Y)。

3、按键处理程序：



4、行列式键盘接口与编程

1) 接口电路

0至15号键的键值码

E7H, D7H, B7H, 77H

EBH, DBH, BBH, 7BH

EDH, DDH, BDH, 7DH

EEH, DEH, BEH, 7EH

0号键键码值11100111, 或

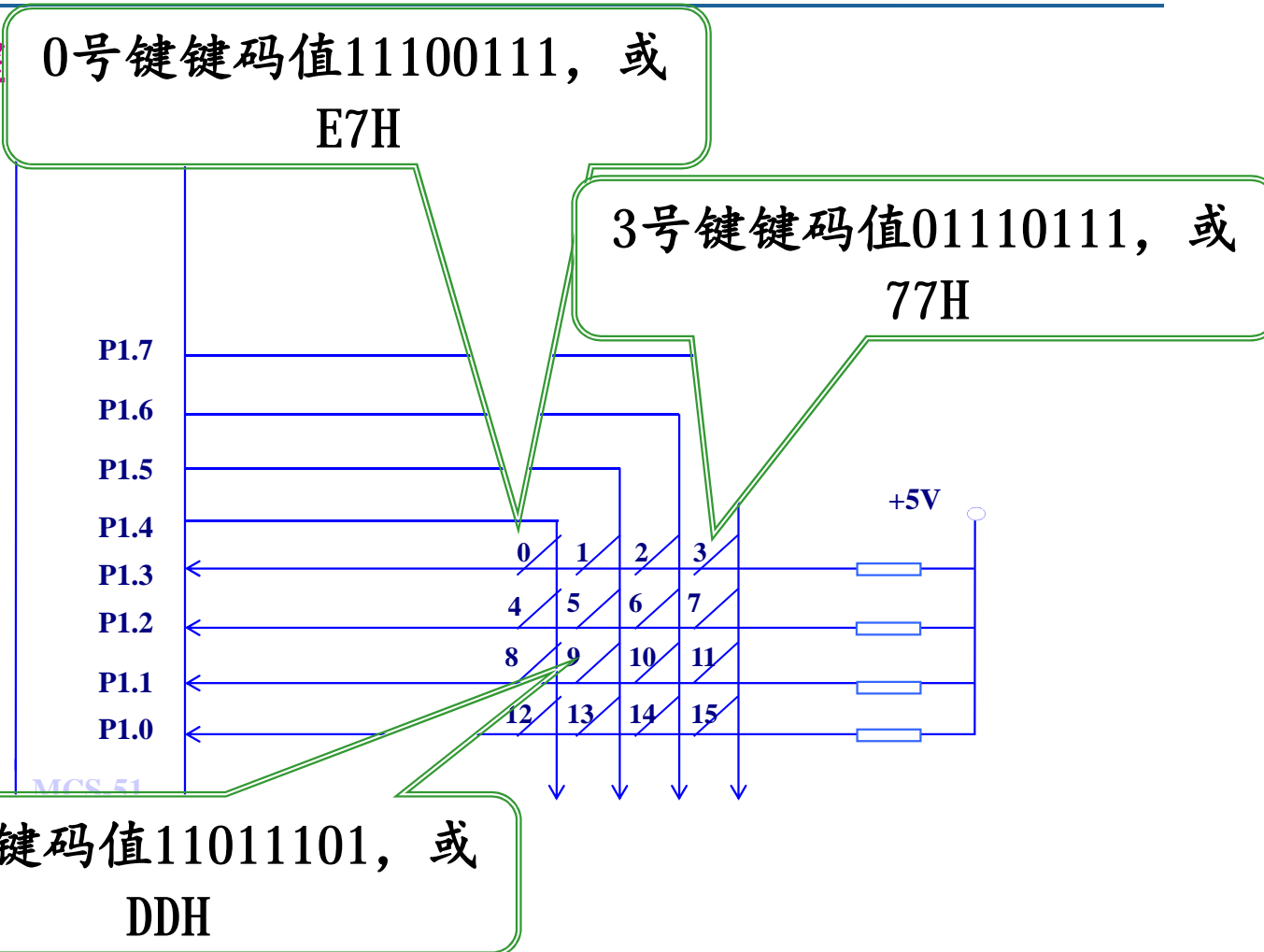
E7H

3号键键码值01110111, 或

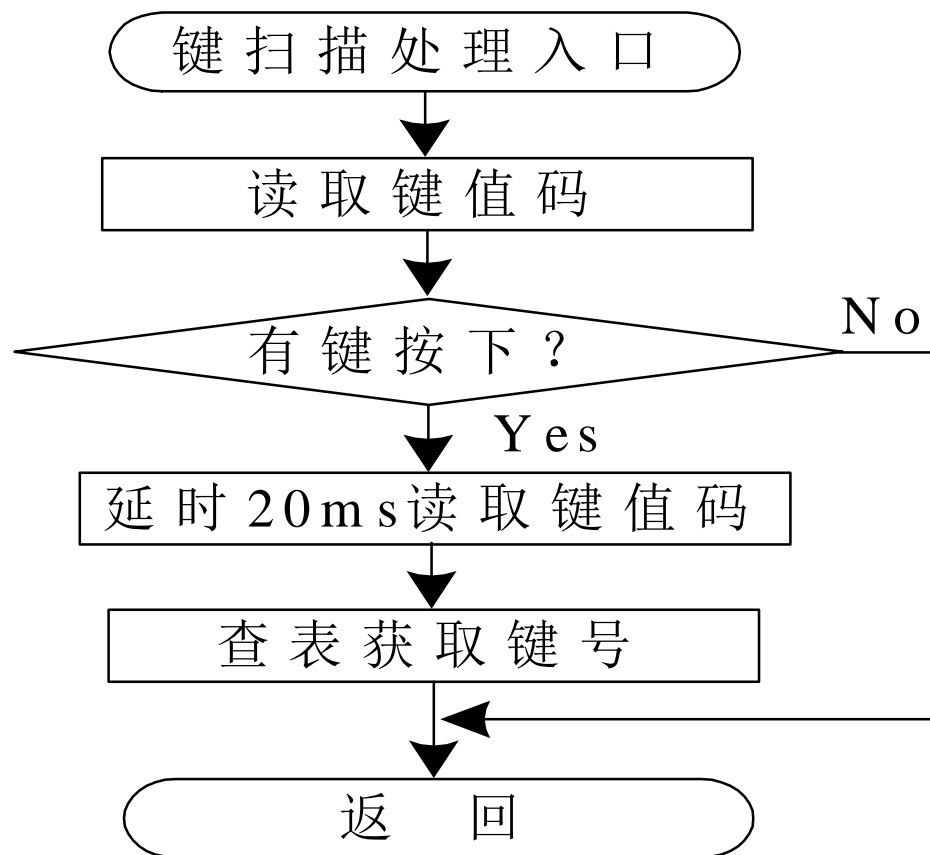
77H

9号键键码值11011101, 或

DDH



2) 反转法编程



反转法键盘程序一获取键值码

```
KS:  MOV  P1, #0FH      ; 列输出0
      MOV  A, P1        ; 读入行信息
      MOV  P1, #0F0H    ; 行输出0
      ORL  A, P1        ; 读入并合成键值码
      CJNE A, #0FFH, KS1
      RET              ; 没有键按下, 返回A=-1
```

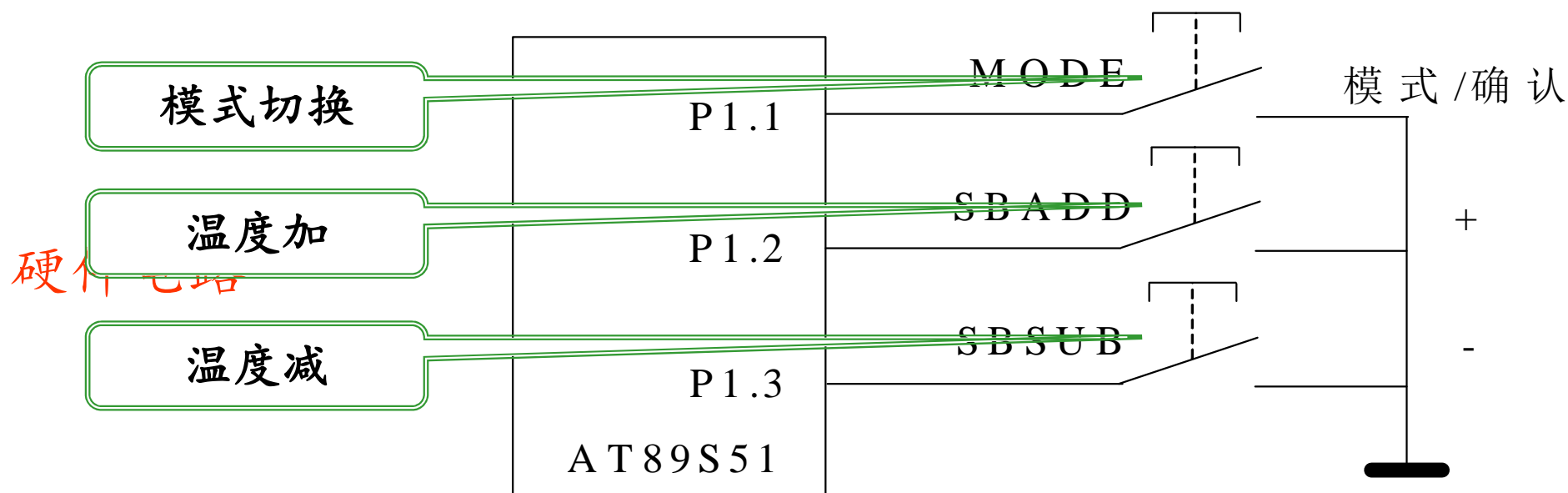
```
KS1:  LCALL D20MS      ; 延时20ms
      MOV  P1, #0FH    ; 列输出0
      MOV  A, P1        ; 读入行信息
      MOV  P1, #0F0H    ; 行输出0
      ORL  A, P1        ; 重新读入并合成键值码
      MOV  R3, A        ; 保存键值码
```

反转法键盘程序—获取键号

```
MOV DPTR, #TAB
MOV R7, #0          ; 预置键号
KS2: MOV A, R7
MOVC A, @A+DPTR    ; 取一个键值码
XRL A, R3
JZ KS3             ; 找到键号, 返回
INC R7
CJNE R7, #16, KS2   ; 未完继续
MOV R7, #0FFH       ; 未找到
KS3: MOV A, R7
RET
TAB: DB E7H, D7H, B7H, 77H, EBH, DBH, BBH, 7BH
      DB EDH, DDH, BDH, 7DH, EEH, DEH, BEH, 7EH
```

教学案例：烘箱温度控制系统键盘子系统设计

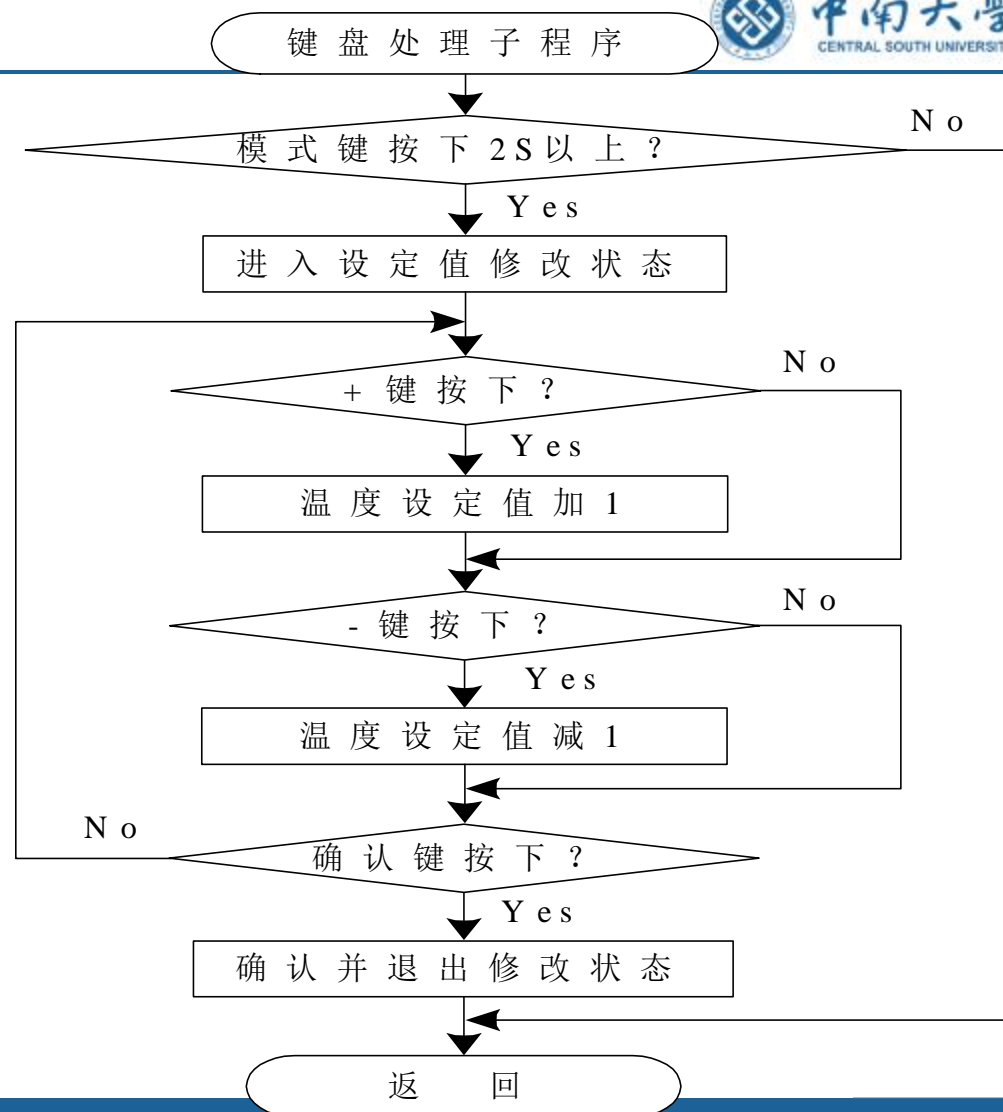
任务：烘箱温度控制系统的温度控制范围是 $20\sim 240^{\circ}\text{C}$ ，要求温度设定值能够通过按键进行设定，设定温度时温度步进值 1°C 。



键盘程序设计

键盘程序要求查询键盘的状态，有键按下时，作出相应的处理。温度控制的设定值为SV，在任何情况下，温度控制程序均根据SV值与温度测量值PV进行比较，实现温度的控制目的。当系统进入温度设定值修改状态后，键盘程序不直接修改设定值SV，而是修改设定值备份数据SVTEMP，退出修改状态时，将修改好的设定值备份数据SVTEMP赋值给设定值SV，设定值修改结束。

键盘处理程序流程图



程序名: KEYPROG

功能: 键盘处理程序, 查询键盘状态, 修改烘箱温度控制设定值

占用资源: 累加器A, 状态寄存器PSW, 数据指针DPTR, 通用寄存器00组

状态标志: FLAG=0, 正常状态; FLAG=1, 设定值修改状态

定义存储单元

FLAG	BIT	F0	; 标志位
MODE	BIT	P1.1	; 模式/确认键
SBADD	BIT	P1.2	; 十键
SBSUB	BIT	P1.3	; 一 键
SV	EQU	30H	; 设定值单元
SVTEMP	EQU	32H	; 设定值备份单元

MODE键按下2s判断程序

等待按键释放过程中调用显示程序，保证键按下时显示正常

跳转时未到2s

```
KEYPROGL:JB    MODE,KEY9; 模式键未按下 退出
            MOV    R3,#100    ; 2S计时
KEY1:       LCALL DISPLAY ; 调用显示程序，延时20mS
            DJNZ   R3,KEY2    ; 2S计时
            SETB   FLAG       ; 2S到，进入设定值修改
KEY2:       JNB    MODE,KEY1; 等待模式键松开释放
            LCALL  DISPLAY    ; 延时20mS
            JNB    FLAG,KEY9 ; 模式键按下未到2S退出
```

十键判断程序

```
KEY3: JB      SBADD,KEY5 ; 十键未按下, 判断其它键
      MOV     A,#240      ; 设定值上限
      XRL     A, SVTEMP
      JZ      KEY4        ; 设定值已达上限则转移
      INC     SVTEMP      ; 设定值加1
KEY4: LCALL    DISPLAY     ; 调用显示程序, 延时20mS
      JNB     SBADD,KEY4 ; 等待十键松开释放
      LCALL    DISPLAY     ; 延时20mS
```


一键判断程序

```
KEY5: JB      SBADD,KEY7 ; 一键未按下, 判断其它键
      MOV     A,#20      ; 设定值下限
      XRL     A, SVTEMP
      JZ      KEY6       ; 设定值已达下限则转移
      DEC     SVTEMP     ; 设定值减1
KEY6: LCALL    DISPLAY    ; 调用显示程序, 延时20mS
      JNB     SBADD,KEY6 ; 等待一键松开释放
      LCALL   DISPLAY     ; 延时20mS
```

MODE键判断程序

等待按键释放过程中调用显示程序，保证键按下时显示正常

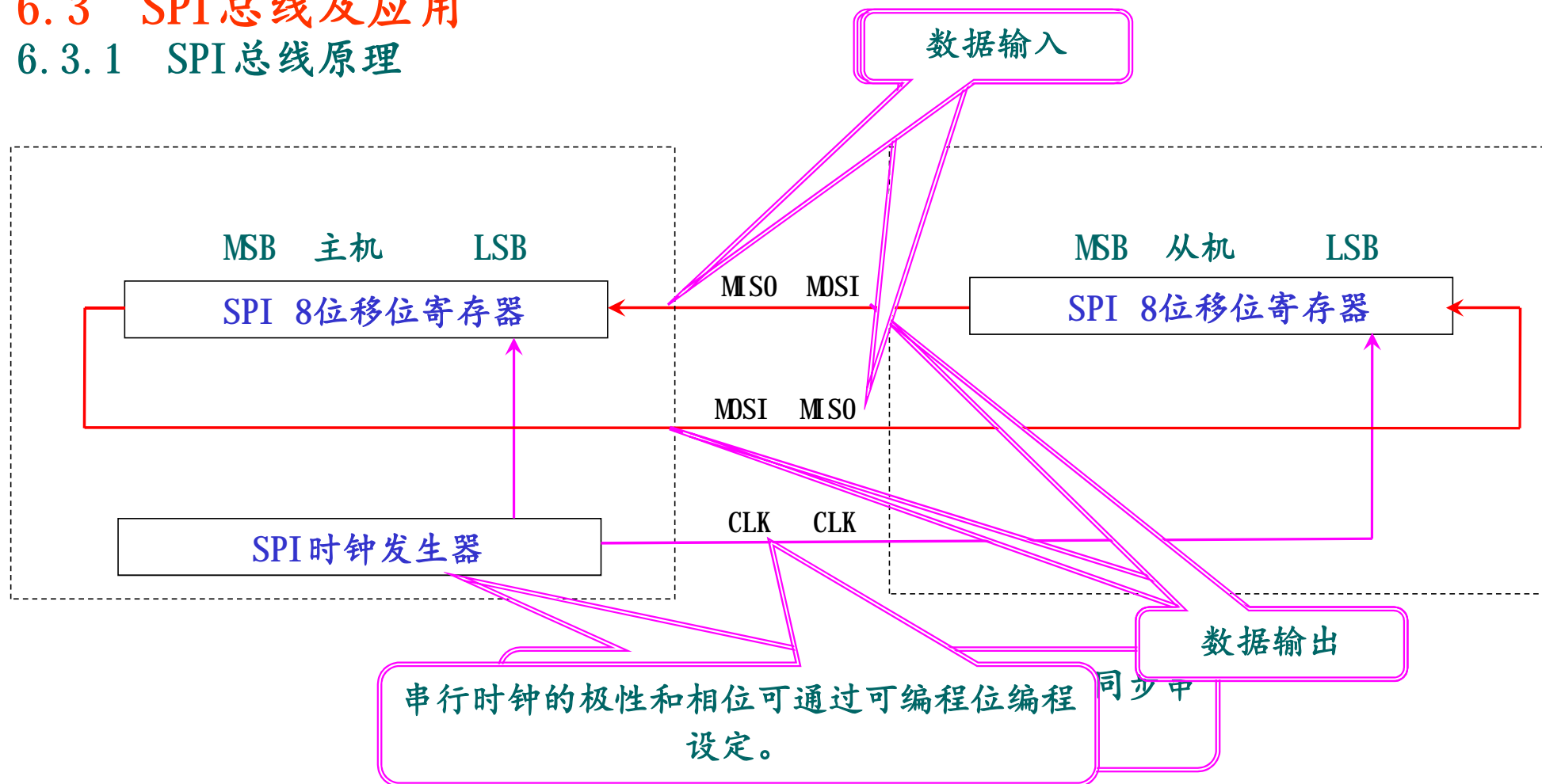
```
KEY7: JB      MODE,KEY3; 确认键未按下，判断其它键
      CLR      FLAG      ;退出设置，值修改状态
      MOV      SV,SVTEMP ;确认修改值
KEY8: LCALL    DISPLAY    ;调用显示程序，延时20mS
      JNB      MODE,KEY8; 等待确认键松开释放
      LCALL    DISPLAY      ;延时20mS
KEY9: RET
```

仿真

程序

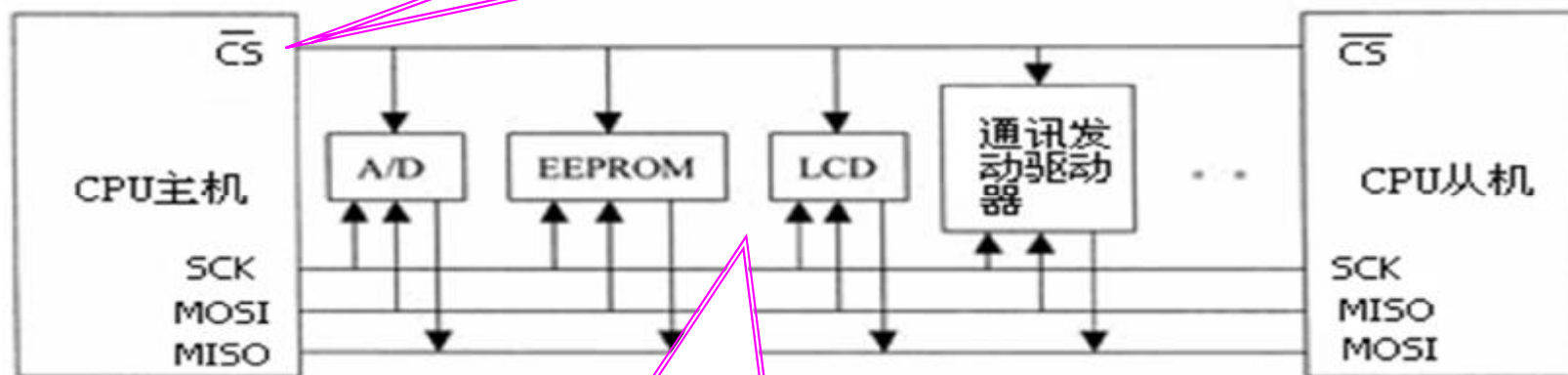
6.3 SPI总线及应用

6.3.1 SPI总线原理



SPI总线接口

主机发出片选信号选择不同的从机。



SPI适用于很多外部器件与主机的通信。

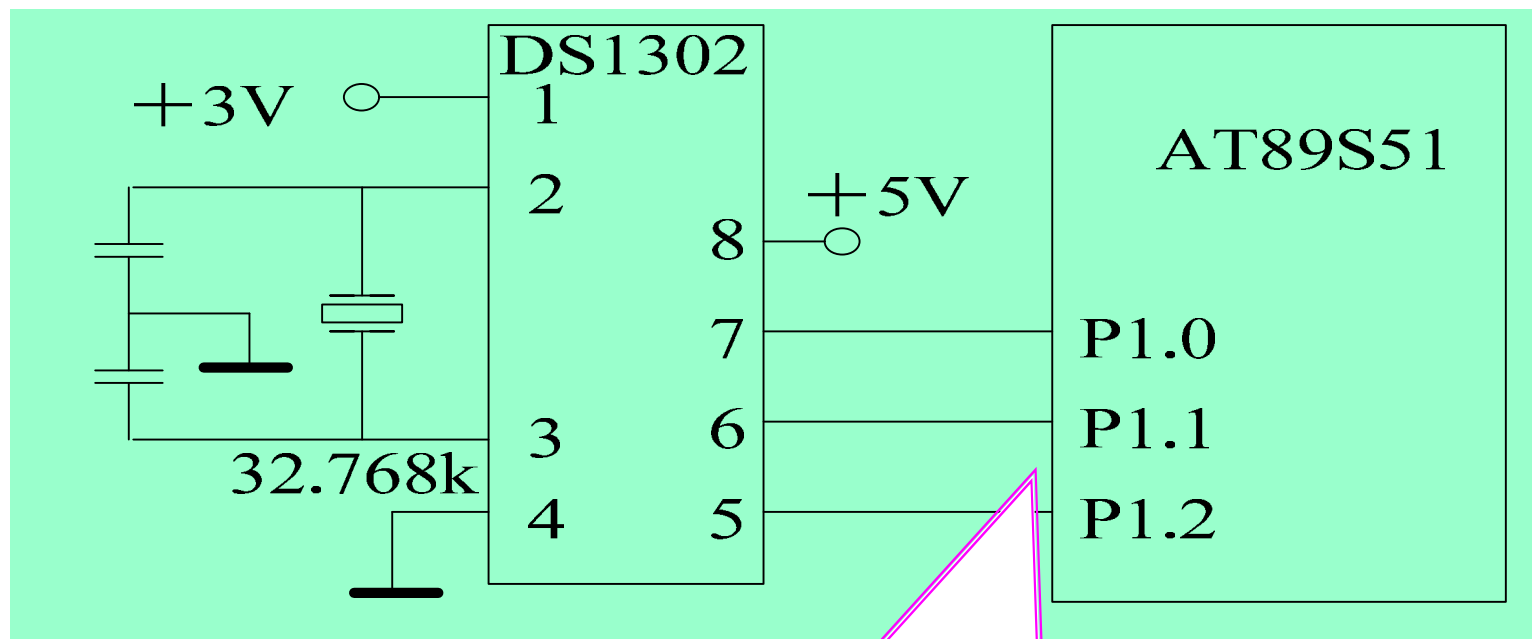
6.3.2 日历时钟芯片DS1302应用

功能：高性能、低功耗、带RAM的实时时钟芯片，它可以对年、月、日、周日、时、分、秒进行计时，且具有闰年补偿功能，工作电压宽达2.5~5.5V。

引脚：

引脚号	引脚名称	功能
1	VCCZ	主电源
2、3	X1, X2	振荡源，外接32768Hz
4	GND	地线
5	/RST	复位/片选线
6	I/O	串行数据输入/输出（双向）
7	SCLK	串行数据输入端
8	VCC1	后备电源

接口电路:

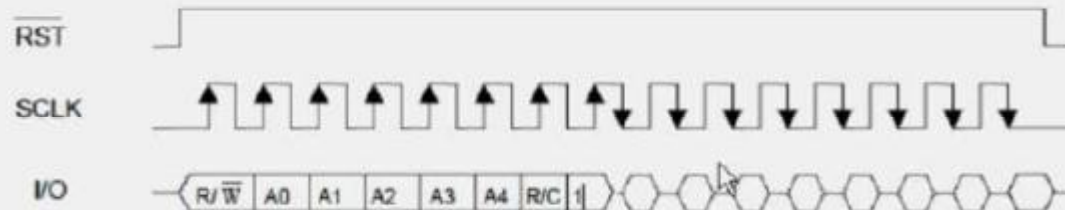


AT89S51没有SPI接口，用P1.0-P1.2模拟SPI。

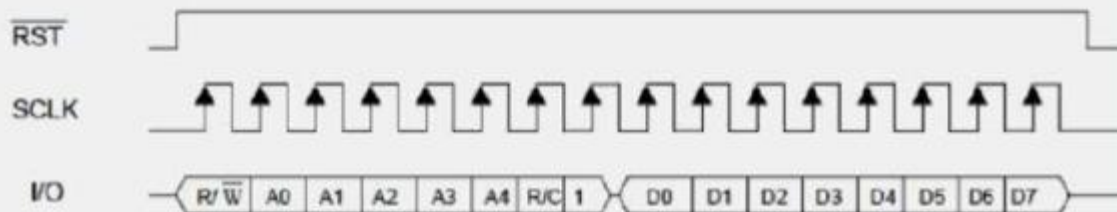
寄存器与命令字

DATA TRANSFER SUMMARY Figure 3

SINGLE BYTE READ



SINGLE BYTE WRITE



In burst mode, \overline{RST} is kept high and additional SCLK cycles are sent until the end of the burst.

数据输入输出

在控制命令字输入后的下一个SCLK时钟的上升沿时数据被写入DS1302，数据输入从低位即位0开始。同样，在紧跟8位的控制指令字后的下一个SCLK脉冲的下降沿读出DS1302的数据，读出数据时从低位0位至高位7。

写入一个字节A

```
WR02:MOV R7,#8
LP021: CLR P1.0
      RRC A
      MOV P1.1,C
      NOP
      SETB P1.0
      NOP
      DJNZ R7,LP02
      RET
```

读出一个字节至A

```
RD02: MOV R7,#8
LP021: CLR P1.0
      NOP
      MOP C,P1.1
      RRC A
      SETB P1.0
      NOP
      DJNZ R7,LP021
      RET
```


读入分钟值至A

```
RDMIN:SETB  P1.2    ;片选
          MOV  A,#82H ;读分钟命令字
          LCALL WR02  ;写入读分钟命令字
          LCALL RD02  ;读分钟值
          CLR   P1.2
          RET
```

写入A中的分钟值

```
WRMIN:SETB  P1.2    ;片选
        PUSH  ACC
        MOV   A,#83H ;写分钟命令字
        LCALL WR02   ;写入写分钟命令字
        POP   ACC
        LCALL WR02   ;写入分钟值
        CLR   P1.2
        RET
```

补充：显示驱动接口芯片MAX7219应用

MAX7219功能：

- 1、MAX7219 是MAXIM 公司生产的7 段共阴极LED 数码管的驱动芯片，每一片7219 最多可驱动8位数码管或64个单独的LED 。
- 2、它集BCD 码译码器、多路扫描器、段驱动和位驱动电路于一体，可保存8位LED 数据。
- 3、外围接口电路简单，使用方便，仅需三根I/O 口线便可提供串行数据输入DIN、时钟信号CLK 和数据锁存信号LOAD。

MAX7219的引脚

DIN: 串行数据输入端，数据在CLK上升沿送入移位寄存器。

DIG0-7: 为阴极开关，可分别接至8组数码管的公共阴极。

GND: 地（两个都应接地）。

LOAD: 数据输入允许端，在LOAD上升沿将数据锁存。

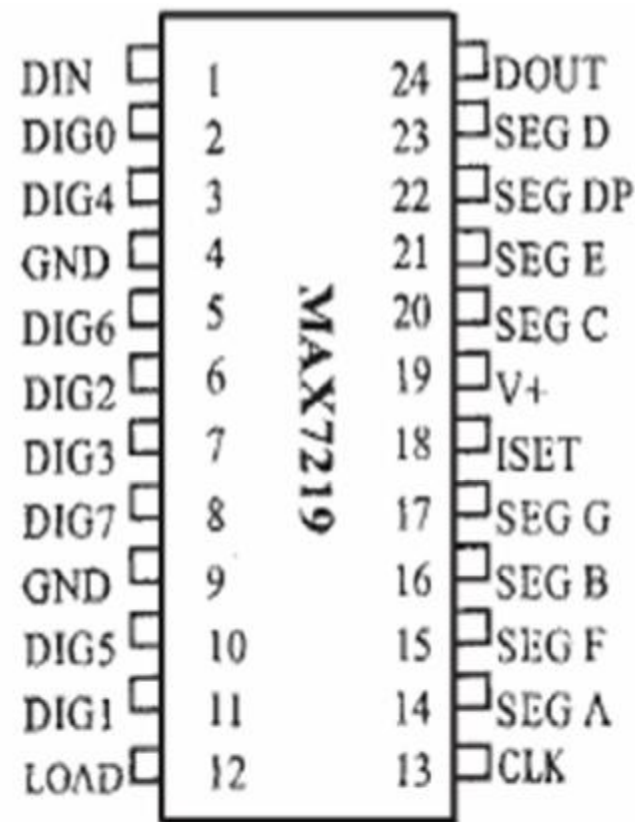
CLK: 串行数据时钟输入端，时钟频率范围0-10MHz。

SEG: 阴极驱动电流源，接字形段码输入。

ISSET: 外接电阻限制峰值电流。

V+: 正电源输入，接+5V。

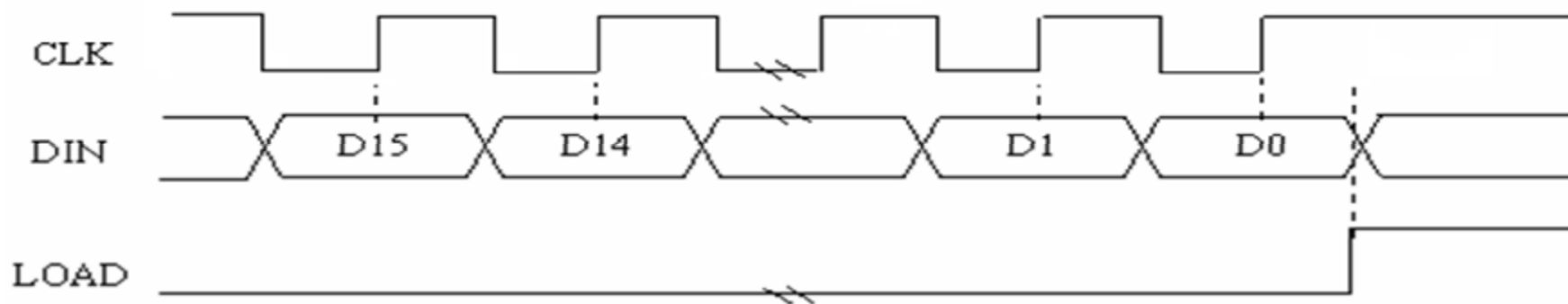
DOUT: 串行数据输出端，用作扩展用。在CLK下降沿输出数据。



MAX7219 引脚图

MAX7219工作原理:

- MAX7219 芯片具有典型的三线串行接口，命令与数据组成16 位字串，从DIN 管脚串入，从DOUT管脚串出。
- 当每一个CLK 脉冲上升沿到来时，串行数据从DIN 管脚进入7219 的内部移位寄存器，在CLK下降沿从DOUT 管脚移出，最先收到的是最高位D15。 LOAD 的上升沿将数据锁存入内部寄存器中。 时序图如下图所示。



MAX7219的数据与寄存器:

16 位串行数据中, D15~D12 为无关位, D11~D8为寄存器地址位, 低8 位为寄存器数据 (LED位驱动寄存器数据或状态寄存器数据)。

串行数据格式表

D 15	D 14	D 13	D 12	D 11	D 10	D 9	D 8	D 7	D 6	D 5	D 4	D 3	D 2	D 1	D 0
×	×	×	×	← 地址 →				← 数据 →							

MAX7219 有14 个可寻址寄存器, 分别为8个位驱动数据寄存器和6 个状态寄存器, 寄存器被初始化后, 如不重新初始化, 则7219 保持初始化的那些状态。

寄存器地址分配表

寄存器	D15- D12	地址				16 进制码
		D11	D10	D9	D8	
空操作	×	0	0	0	0	×0
位 0	×	0	0	0	1	×1
位 1	×	0	0	1	0	×2
位 2	×	0	0	1	1	×3
位 3	×	0	1	0	0	×4
位 4	×	0	1	0	1	×5
位 5	×	0	1	1	0	×6
位 6	×	0	1	1	1	×7
位 7	×	1	0	0	0	×8
译码方式	×	1	0	0	1	×9
显示亮度	×	1	0	1	0	×A
扫描位数	×	1	0	1	1	×B
关闭	×	1	1	0	0	×C
显示测试	×	1	1	1	1	×F

- **数字寄存器：**地址01H—08H，对应LED1—LED8。不译码时，D0—D6分别对应标准7段显示器的A—G，正逻辑显示。译码时，D3—D0为显示数据的BCD码。无论译码与否，D7为1，则该位小数点显示。
- **译码方式寄存器：**地址09H，D0—D7对应LED1—LED8，正逻辑译码。例如D0为1，则LED1工作在译码方式。
- **亮度控制寄存器：**地址0AH，D3—D0分16挡控制亮度，0000B全熄，1111B最亮。D7—D4未用。

- **显示位数寄存器：**地址0BH，由D2—D0内容设置所显示数据的多少，可从1到 8。例如010B控制显示最后3个LED。
- **关闭寄存器：**地址0CH，此时D0=0，7219关闭， D0=1，7219正常工作。关闭模式可被任何控制功能取消。
- **显示测试寄存器：**地址0FH，D0=1为测试方式。在测试方式7219不考虑所有控制寄存器和数字寄存器的内容而接通所有LED（全亮）。此时 若D0=0则7219正常工作。
- **空操作寄存器：**地址00H，用于7219级联。

7219写入子程序

入口：累加器A为寄存器地址，R4为待写入数据。

DIN BIT P1.0

CLK BIT P1.1

LOAD BIT P1.2

WRITE: LCALL SEND ; 写入寄存器地址

 MOV A, R4

 LCALL SEND ; 写入数据

 CLR LOAD

 SETB LOAD ; 锁存数据

 RET

相当于扩充了一条指令 **LCALL WRITE**

入口：累加器A为寄存器地址，R4为待写入数据。

MOV A, #3
MOV R4, #9
LCALL WRITE

寄存器	D15 D12	地址	16 进制码
		D11 D10 D9 D8	
空操作	×	0 0 0 0	×0
位0	×	0 0 0 1	×1
位1	×	0 0 1 0	×2
位2	×	0 0 1 1	×3
位3	×	0 1 0 0	×4
位4	×	0 1 0 1	×5
位5	×	0 1 1 0	×6
位6	×	0 1 1 1	×7
位7	×	1 0 0 0	×8
译码方式	×	1 0 0 1	×9
显示亮度	×	1 0 1 0	×A
扫描位数	×	1 0 1 1	×B
关闭	×	1 1 0 0	×C
显示测试	×	1 1 1 1	×F

字节写入子程序

入口：累加器A为待写入数据字节。

```
SEND:  MOV    R2, #8
SEND1: CLR    CLK
        RLC    A
        MOV    DIN, C ; 发送一位数据
        SETB   CLK    ; 锁存一位数据
        DJNZ   R2, SEND1
        RET
```

6.4 A/D、D/A转换接口

6.4.1 A/D、D/A转换概述

1、 A/D转换器

- **A/D转换**——实现模拟量变换成数字量的设备称为模数转换器（ADC），简称A/D。
- **用途**——广泛应用于单片机的实时测控和智能化仪表等应用系统中
- **种类**——根据A/D转换器的原理可将A/D转换器分成两大类，一类是直接型A/D转换器，一类是间接型A/D转换器。

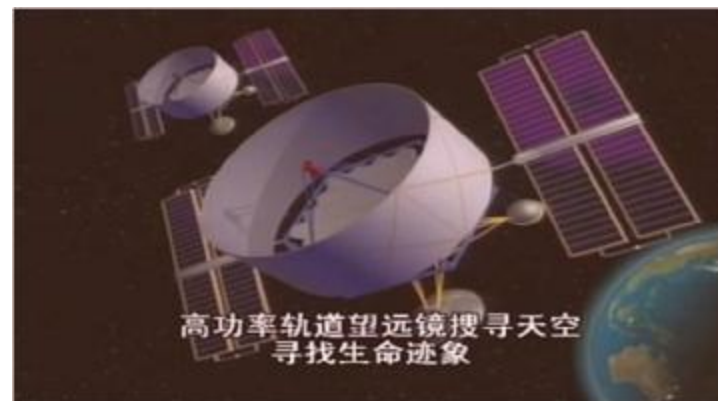
A/D转换器的主要技术指标

1) 分辨率

- 分辨率表示输出数字量变化一个相邻数码所需输入模拟电压的变化量。
- 转换器的分辨率定义为满刻度电压与 2^n 之比，其中 n 为ADC的位数。
- 例如具有12位分辨率的ADC能分辨出满刻度的 $1/2^{12}$ 或满刻度的0.0245%。一个10V满刻度的12位ADC能够分辨输入电压变化的最小值为2.4mV。

2) 转换速率

ADC的转换速率就是能够重复进行数据转换的速度，即每秒转换的次数。而完成一次A/D转换所需的时间（包括稳定时间）则是转换速率的倒数。



常用A/D转换器简介

- A/D转换器按照输出代码的有效位数分为4位、6位、8位、10位、12位、14位、16位、24位和BCD码输出的3位、4位、5位等多种;
- 按照转换速度可以分为超高速（转换时间 $\leq 1\text{ns}$ ）、高速（转换时间 $\leq 1\mu\text{s}$ ）、中速（转换时间 $\leq 1\text{ms}$ ）、低速（转换时间 $\leq 1\text{s}$ ）等几种不同转换速度的芯片;
- 为适应系统集成的需要，有些转换器还将多路转换开关、时钟电路、基准电压源、二/十进制译码器和转换电路集成在一个芯片内，超越了单纯的A/D转换功能。

A/D转换器选择要点

1) 如何确定A/D转换器的位数

- A/D转换器位数的确定与整个测量控制系统所要测量控制的范围和精度有关，但又不能唯一确定系统的精度。
- 估算时，A/D转换器的位数至少要比总精度要求的最低分辨率高一位。
- 实际选取的A/D转换器的位数应与其它环节所能达到的精度相适应。只要不低于它们就行，选得太高既没有意义，而且价格还要高得多。

2) 如何确定A/D转换器的转换速率

- 积分型、电荷平衡型和跟踪比较型A/D转换器转换速度较慢，转换时间从几毫秒到几十毫秒不等，只能构成低速A/D转换器，一般运用于对温度、压力、流量等缓变参量的检测和控制。
- 逐次比较型的A/D转换器的转换时间可从几 μS 到100 μS 左右，属于中速A/D转换器，常用于工业多通道单片机控制系统和声频数字转换系统等。
- 高速A/D转换器适用于雷达、数字通讯、实时光谱分析、实时瞬态记录、视频数字转换系统等。

3) 如何决定是否要加采样保持器

- 原则上直流和变化非常缓慢的信号可不用采样保持器。
- 其它情况都要加采样保持器。



4) 工作电压和基准电压

- 如果选择使用单+5V工作电压的芯片，与单片机系统可共用一个电源就比较方便。
- 基准电压源是提供给A/D转换器在转换时所需要的参考电压，这是为保证转换精度的基本条件。在要求较高精度时，基准电压要单独用高精度稳压电源供给。

5) 正确选用A/D转换器有关量程的引脚

- A/D转换器的模拟量输入有时需要是双极性的，有时是单极性的。输入信号最小值有从零开始，也有从非零开始的。有的A/D转换器提供了不同量程的引脚，只有正确使用，才能保证转换精度。
- 变换量程的双模拟输入引脚和双极性偏置引脚的正确使用。
- 双参考电压引脚的正确使用。
- A/D转换内部比较器反相输入端的正确使用。

2、D/A转换原理

数字量与模拟量的关系式:

$$V_0 = V_{\text{REF}} / 2^n \times D$$

V_{REF} - 参考电压

n - 数字量位数

D - 转换的数字量

实现数字量变成模拟量的设备称为数模转换器简称D/A。

D/A转换器主要技术指标

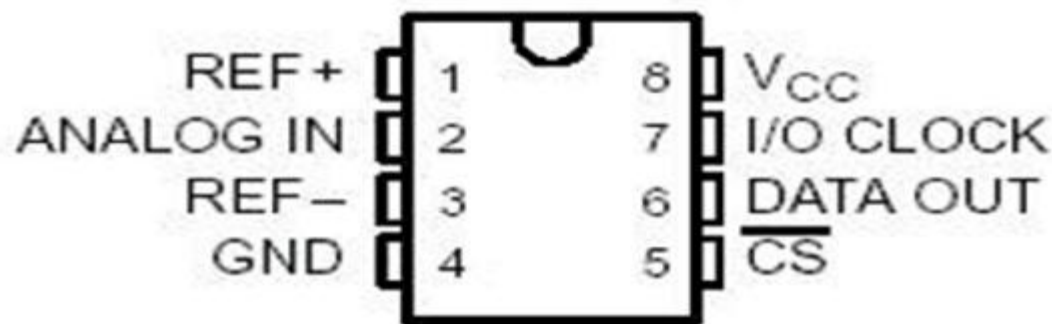
- 1、D/A建立时间（转换速度）
- 2、D/A转换精度（以满刻度的百分数给出）
- 3、分辨率（ 2^{-n} ）

6.4.2 A/D转换器TLC548及其与单片机接口

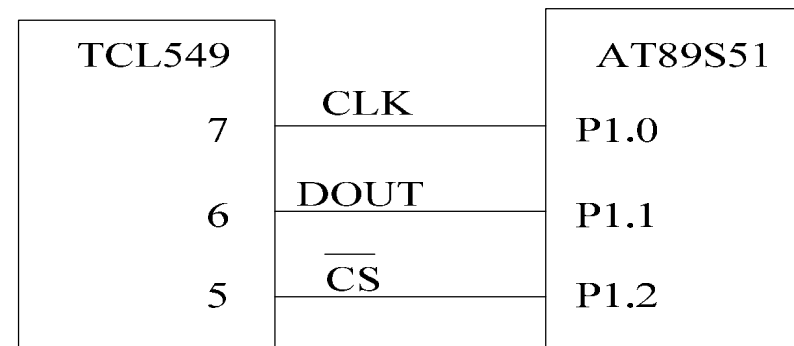
TLC548的性能介绍:

- 8-位A/D转换分辨率, 转换时间最大17微秒
- 差分式参考电压输入
- 每秒总的转换次数 45500
- 芯片有软件控制及采保功能
- 总的非调整误差 . . . 最大 ± 0.5 LSB
- 内部4-MHz典型系统时钟
- 比较宽的电压范围. . . 3 V - 6 V
- 采用CMOS工艺制造, 能耗最大15 毫瓦

引脚与接口



引脚图



接口图

ANALOG IN: 模拟信号输入端

REF+ -: 基准电压输入端

$\overline{\text{CS}}$: 芯片选择输入端

DATA OUT: 转换结果数据串行输出端

I/O CLOCK: 外接输入 / 输出时钟输入端

TLC548工作时序

- (1) CS置低等待两个内部时钟周期后，将转换结果的最高位D7输出到DATAOUT端。
- (2) 前7个CLOCK周期的下降沿依次移出D6、D5、D4、D3、D2、D1、D0。
- (3) 第8个时钟周期的下降沿启动下一次转换。

接口程序

; 初始化:

```
SETB P1.2      ; 置CS为1。  
CLR P1.0        ; 置I/O CLOCK为零。  
MOV R0, #08H    ; 移位计数。
```

; A/D过程:

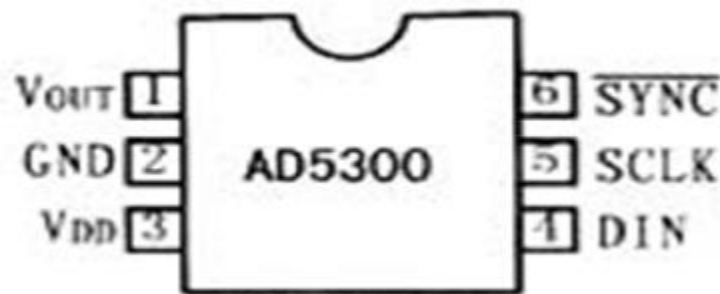
```
ADP:  CLR P1.2      ; 启动  
      NOP           ; 等待1.4 μs  
NXT:  SETB P1.0  
      MOV C, P1.1  
      RLC A  
      CLR P1.0  
      DJNZ R0, NXT  
      SETB P1.2  
      RET
```

6.4.3 D/A转换器AD5300及其与单片机接口

1、AD5300的特点及功能

- 单电源供电，电压范围为2.7~5.5V；
- 低功耗，正常模式下的典型功耗为0.7mW；
- 掉电工作模式下的典型工作电流为200nA；
- 具有上电复位电路；
- 以电源电压VDD为芯片参考电压，从而使DAC具有0V~VDD最充裕的动态输出范围；
- 内含输出缓冲放大器，因而使DAC具有高达1V / μ s的转换速率。

2、引脚图



管脚	名称	功能说明
1	VOUT	模拟电压输出
2	GND	地
3	VDD	电源 (2.5-5.5V)
4	DIN	串行数据输入
5	SCLK	串行时钟输入
6	SYNC	电平触发控制输入

3、16位数据 及写时序

× × PD1 PD0 D7 D6 D5 D4 D3 D2 D1 D0 × × × ×

工作模式:

PD1

0

0

1

1

PD0

0

1

0

1

工作模式

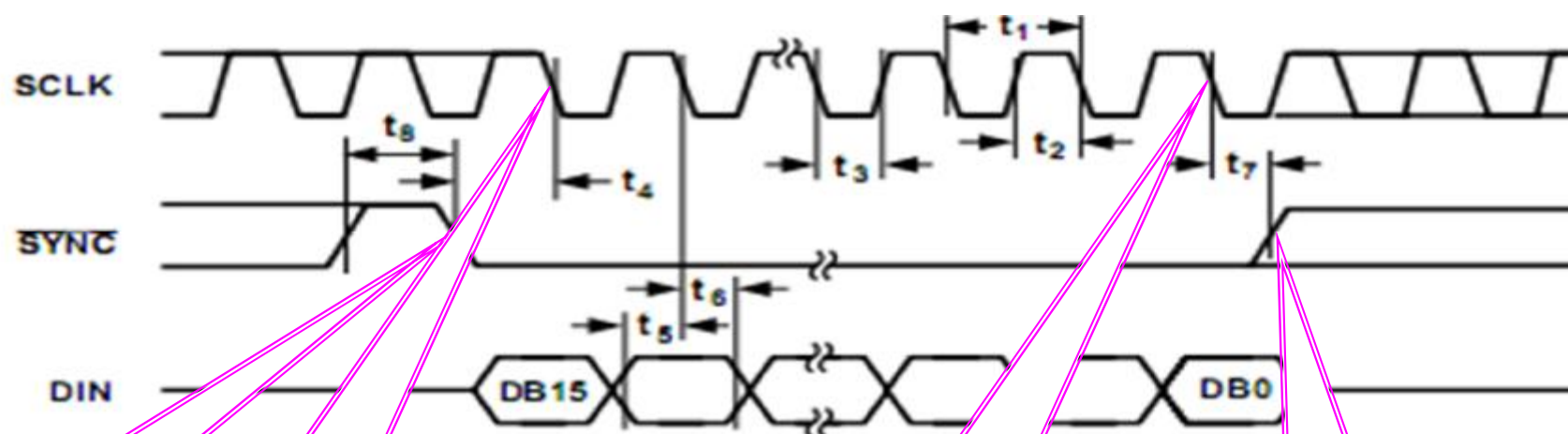
正常模式，输出电压

输出端内接1k电阻到地

输出端内接100k电阻到地

三态

写时序:



通信开始

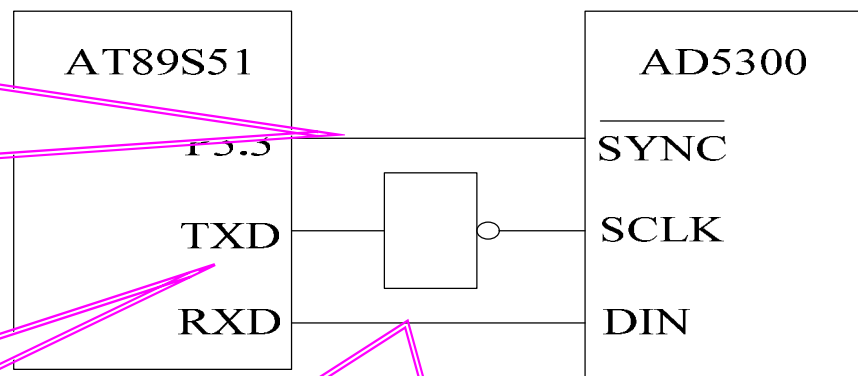
送出DB15

送出DB0

结束

4、接口电路及编程

在串行数据输入过程中，
SYNC必须保持为低电平，
直到通信结束，否则，写
操作无效。



用串行口方式0即可将数据发送
至AD5300。

80C51串口的数据输出是低位先出，而AD5300
的16位移位寄存器则是高位先入，因此，在
80C51向AD5300写操作前应将数据高位低位调
整。

发送程序（数据在R3高、R4低中）

```
SEND: MOV     A,R3
      LCALL    CONT      ; 高低位转换
      CLR     P3.3      ; 置SYNC为0开始。
      MOV     SUBF,A      ; 发送高位。
      JNB     RI,$
      CLR     RI
      MOV     A,R4
      LCALL    CONT
      MOV     SUBF,A      ; 发送低位。
      JNB     RI,$
      CLR     RI
      SETB    P3.3      ; 结束
      RET
```


A内容高低位转换程序

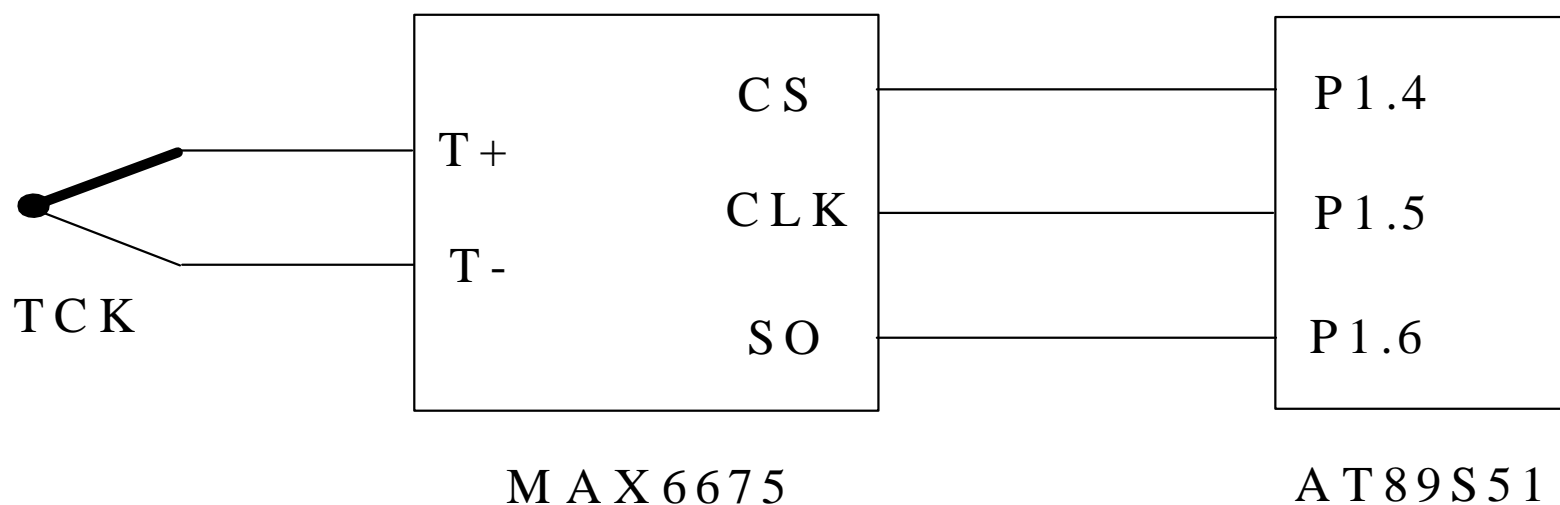
```
CONT:  MOV    R5,A
        MOV    R6,A
        MOV    R7,#8
CONT1:  MOV    A,R5
        RLC    A
        MOV    R5,A
        MOV    A,R6
        RRC    A
        MOV    R6,A
        DJNZ   R7,CONT1
        MOV    A,R6
        RET
```

教学案例：烘箱温度控制系统温度采样程序设计

热电偶测量温度

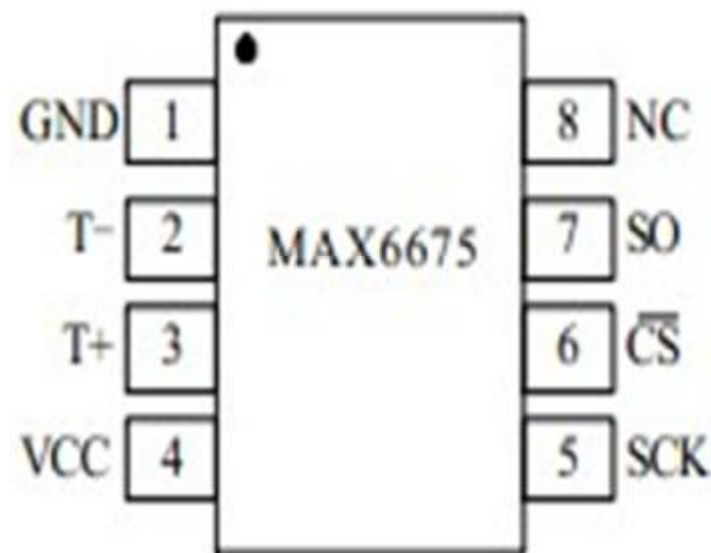
- 优点：结构简单、制造容易、使用方便、测温范围宽、测温精度高
- 缺点：非线性、需要冷端补偿、非数字化输出

任务：烘箱温度控制系统中的温度测量采用MAXIM公司推出的K型热电偶串行模数转换器MAX6675，硬件电路如图所示，要求编写其温度采样程序，读入当前烘箱温度测量值。



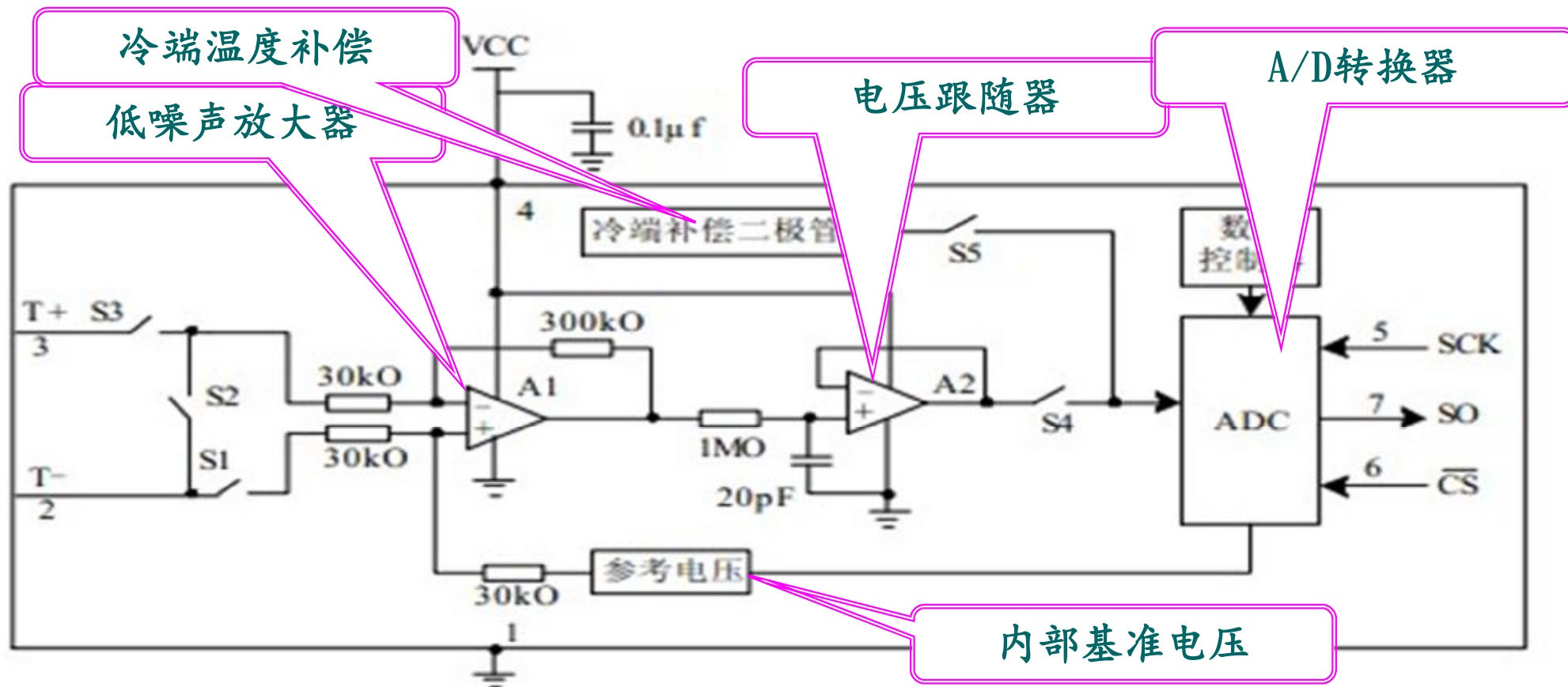
1、MAX6675性能特点

- ① 简单的SPI 串口温度值输出
- ② 0~+1024度的测温范围
- ③ 12位的分辨率
- ④ 片内冷端补偿
- ⑤ 高阻抗差动输入
- ⑥ 热电偶断线检测
- ⑦ 单一+5V的电源电压
- ⑧ 低功耗特性
- ⑨ 工作温度范围-20~+85°C



引脚

2、MAX6675内部原理图





3、MAX6675的16位数据格式

位	空标 志位12 位温度读												热电偶 输入	设备 身份	状态	
位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	MSB											LSB		0	三态

空

温度最高位

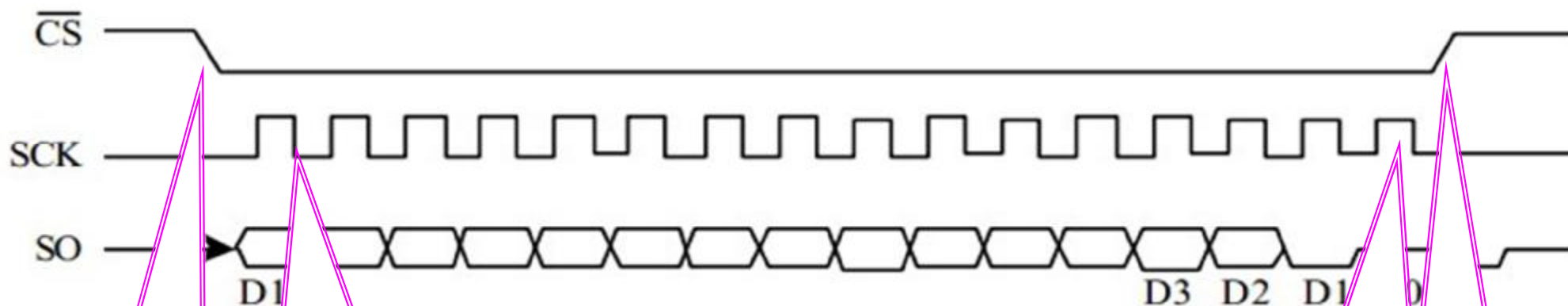
温度整数

设备标识

三态

热电偶输入开路时
为高

4、MAX6675输出16位数据时序



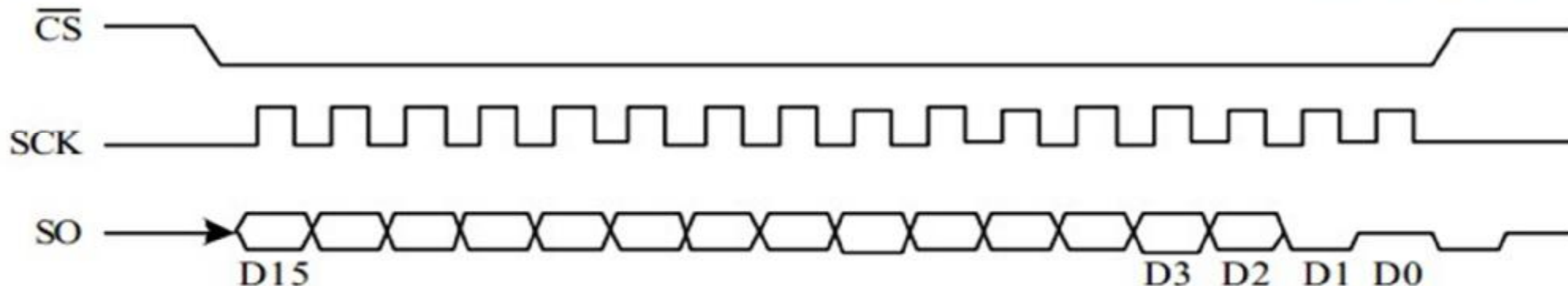
停止
启动

此后每一个SCK脉冲输出一位数据

输出全部16位数据需要16个SCK脉冲

5、采样程序设计

- MAX6675转换一次需要0.22s时间，采样周期确定为1s，由定时中断程序实现。采用程序如下，程序只输入了温度数据的整数位，将无关位和小数位均丢弃。
- **程序名：** SAMPROG
- **功能：** 温度采样程序，首先读入上次A/D转换的结果，然后启动下一次转换。
- **占用资源：** 累加器A，状态寄存器PSW，通用寄存器01组（R2、R3、R4）
- **测量值：** 测量值高位在PVH中，低位在PVL中

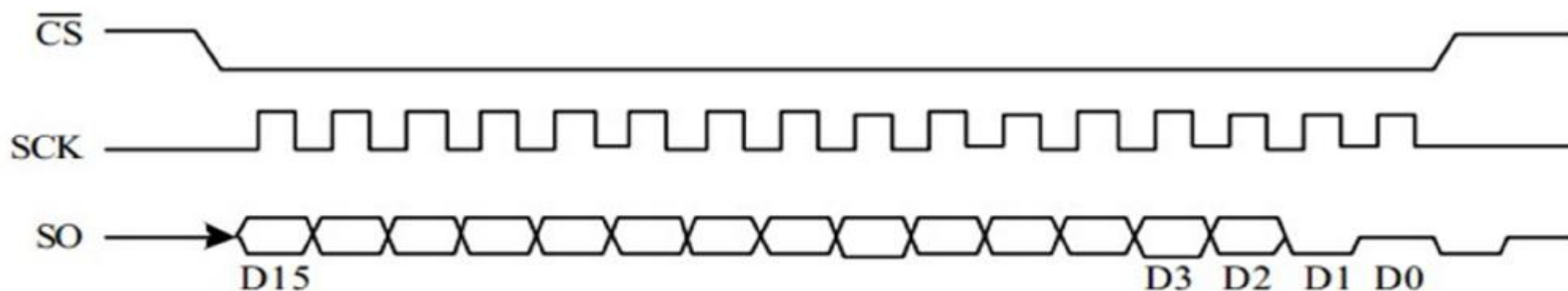


```

SAMPROG: CLR  SCK  ;时钟置为低电平
          CLR  CS   ;CS低, 停止转换, 准备输出数据
          SETB SCK  ;丢弃D15
          CLR  SCK
          MOV  R7, #4; 读数据高4位
RDH:      MOV  C, SO ;读SO端数据, 读D14-D11
          RLC  A    ;累加器左移一位
          SETB SCK
          CLR  SCK
          DJNZ R7, RDH
          MOV  PVH, A; 将数据高位保存
    
```

```

      MOV R7,#8 ; 读数据低8位
RDL:  MOV C,SO  ; 读SO端数据; 读D10-D3
      RLC A     ; 累加器左移一位
      SETB SCK
      CLR SCK
      DJNZ R7,RDL
      MOV PVL,A
      SETB CS   ; CS高电平, 启动新的数据转换
      RET
  
```



补充内容：D/A转换器的双缓冲输出

要求：两路及以上的D/A转换器同时输出

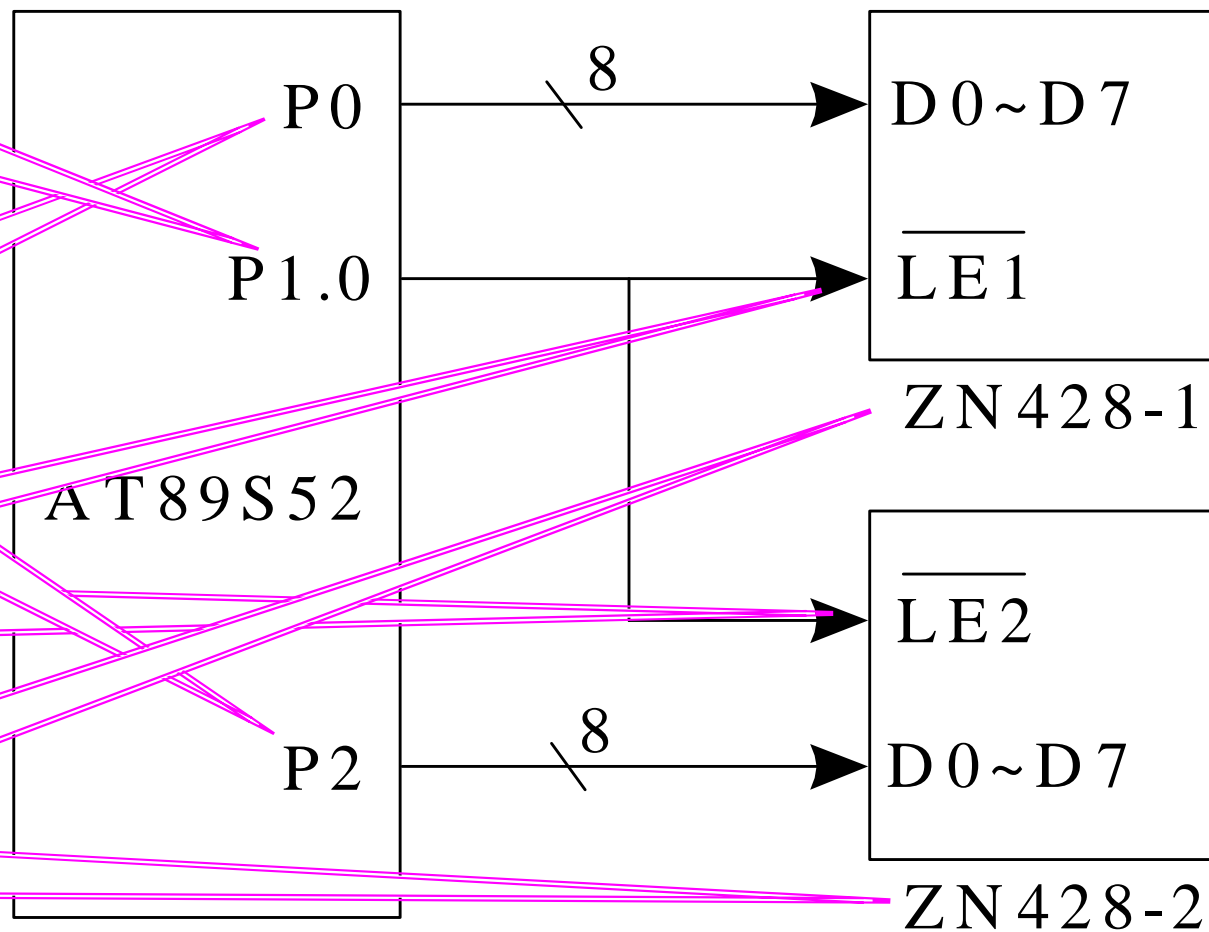


X-Y绘图仪画
斜线

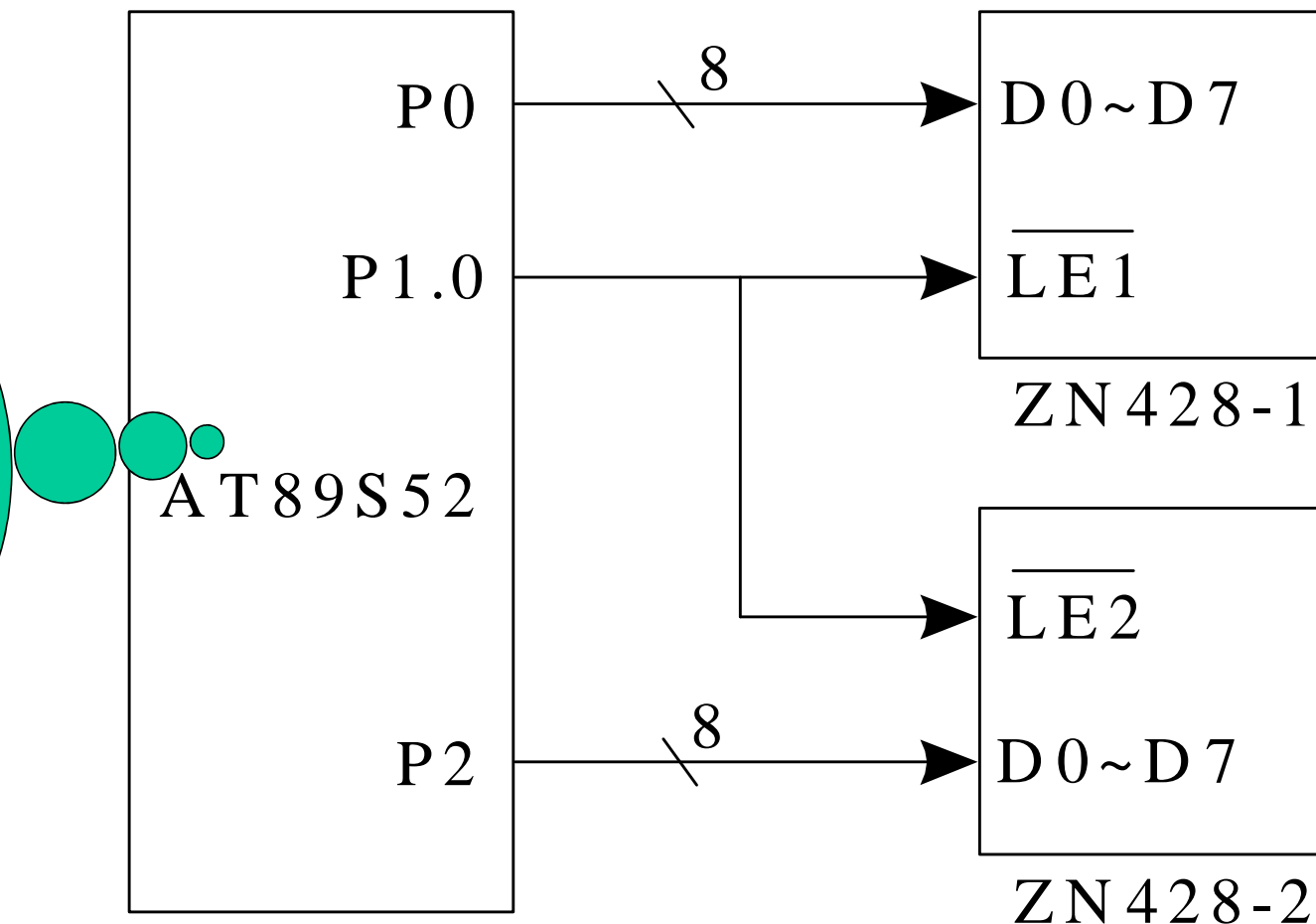
方法：采用双缓冲（两级缓冲）方式

二级缓冲

一级缓冲

数据在LE的上升沿锁存至
内部锁存器ZN428是单通道、8位电压
输出DAC

编程时先将2路数据分别送至P0、P2口，然后再从P1.0发出锁存脉冲。



6.5 功率驱动器件及接口电路

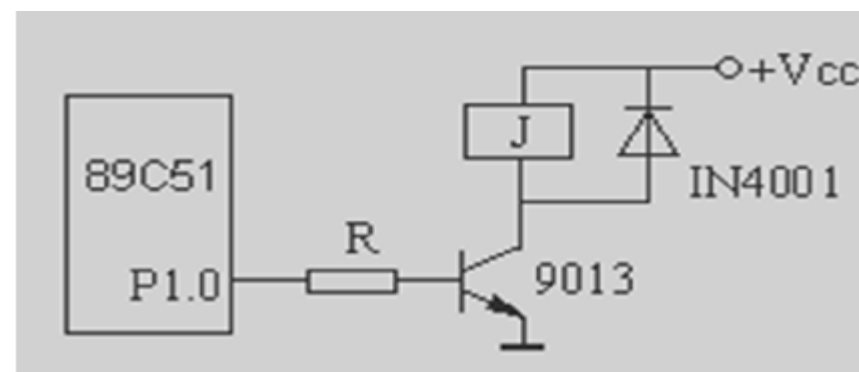
6.5.1 继电器隔离驱动

特点:

优点是隔离可靠，可适应各种交直流负载，缺点是反应迟缓，寿命短（电气寿命一般为10⁵次），因此继电器不宜用在需要快速响应和频繁开关的场合。

电路:

IN4001为续流二极管，用于保护三极管。电阻R的作用是限流，由于89C51有较大的上拉电阻（10k Ω 左右），R也可以省略。



继电器选择要求：

- (1) 体积小，重量轻，能直接焊在印刷电路板上；
- (2) 耗电少，最好能由集成电路、半导体器件直接驱动；
- (3) 线圈电压与控制系统电压兼容，如+5V、+24V等；
- (4) 有较高的可靠性和环境适应能力。

6.5.2 固态继电器接口电路

特点:

控制电流小，工作可靠，寿命长，对外界干扰小，抗干扰能力强，开关速度快，能与集成电路兼容等。

1、 固态继电器的输入控制

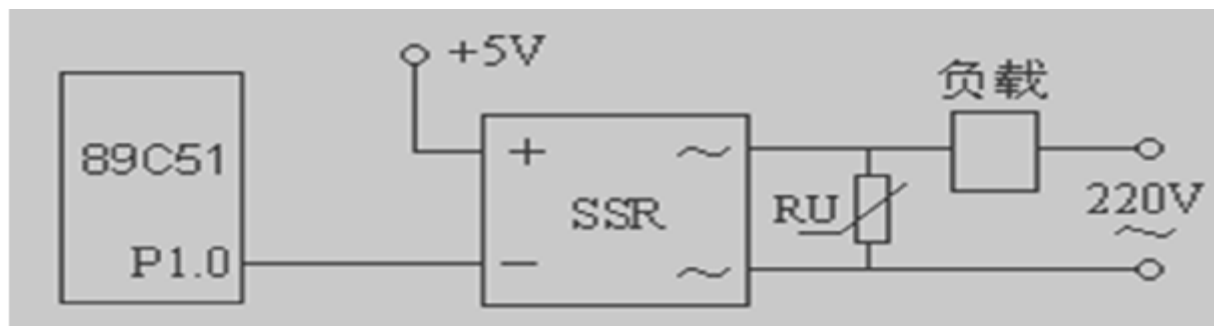
电流控制（限流）型固态继电器的额定输入控制电流 I_s 一般为2—25mA，通过限流电阻 R 控制输入端电流 I_s ，电流的大小随输入电压的变化而线性变化。输入电压确定的情况下，通常选择 R 使 I_s 约为10mA左右。

电压控制（恒流）型固态继电器内的输入部分有一个恒流电路，当输入电压在一个有效范围之内（如 3—32V）时，工作电流基本上稳定在一个数值上。这种方式给固态继电器的应用提供了极大的方便。

2、交流固态继电器

交流输出固态继电器（AC—SSR）的输出器件采用晶闸管或场效应管，对由晶闸管输出的继电器，按其触发方式的不同又分为过零型和随机（移相）型两种。

下图是电压控制型交流固态继电器在单相电路中的典型应用接口，压敏电阻RU的作用是吸收浪涌电压。如果是应用在三相电路中，三个固态继电器的输入端可采用并联（控制电压较低时）或串联（控制电压较高时）的接法。

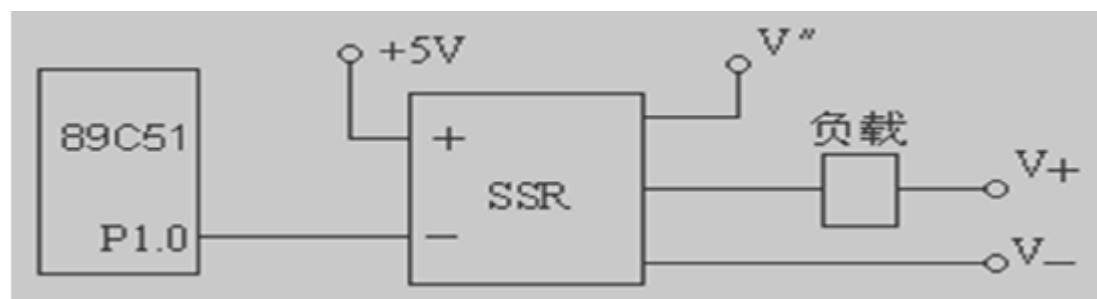


3、 直流固态继电器

直流输出固态继电器（DC—SSR）的输出器件采用大功率晶体管或场效应管，有**二线制**、**三线制**两种不同的输出形式。

下图是是三线制的应用接口电路， V'' 是辅助电源，它的作用是使输出元件达到深度饱和状态，减小压降损失。辅助电压一般小于15V，当负载电压 V_+ 不高时，辅助电源与负载电源可合二为一。

二线制的优点是无辅助电源，使用方便，缺点是输出元件压降大，另外受光电接收管耐压的限制，负载电压也不能太高。

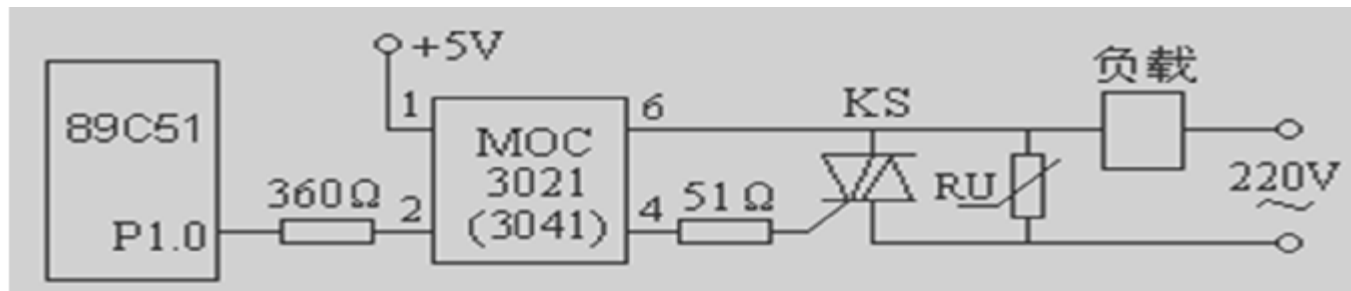


6.5.3 光电耦合元件隔离驱动接口电路

特点：可直接驱动小于100mA的功率负载，与晶闸管配合可驱动工作电流很大（超过SSR的额定电流）的功率负载。

典型器件：MOC3021与MOC3041是两种常用的光电耦合器件，其中MOC3021是移相型，而MOC3041是过零触发型。器件的输出光控双向晶闸管耐压400V，额定电流100mA。

电路：360Ω电阻起输入限流作用，MOC3021/3041的输出控制双向晶闸管KS的通断。KS也可以用两个反向并联的单向晶闸管代替，此时MOC3021/3041应该串接在两个晶闸管的门极之间。

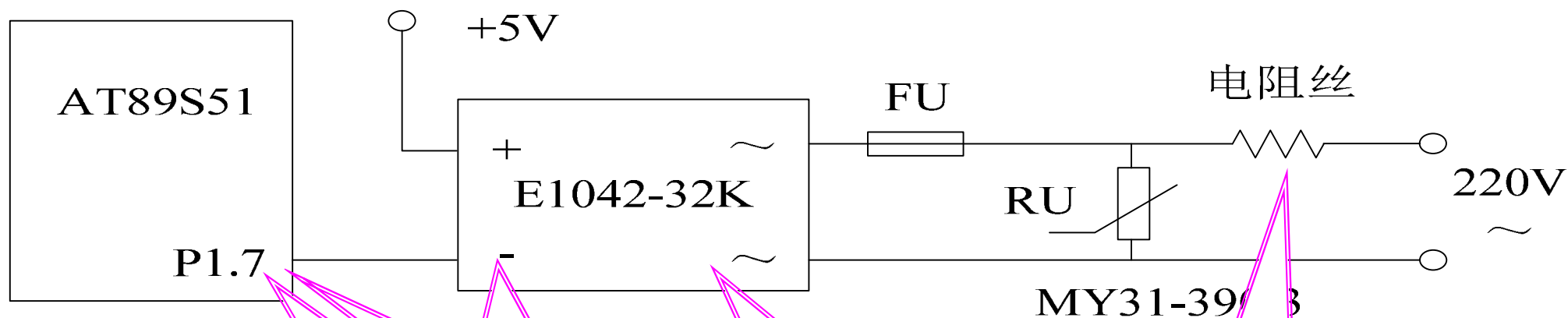


教学案例：烘箱温度控制系统功率控制电路设计

任务：设计制作一个烘箱温度控制系统加热电路，控制对象烘箱容量为20升，加热功率为1kW，加热电源为单相交流220V。

分析：功率不大，为方便采用PWM进行功率调节与控制，优先选用交流固态继电器进行隔离驱动。

电路设计



3~32V (控制 P1.7 输出)

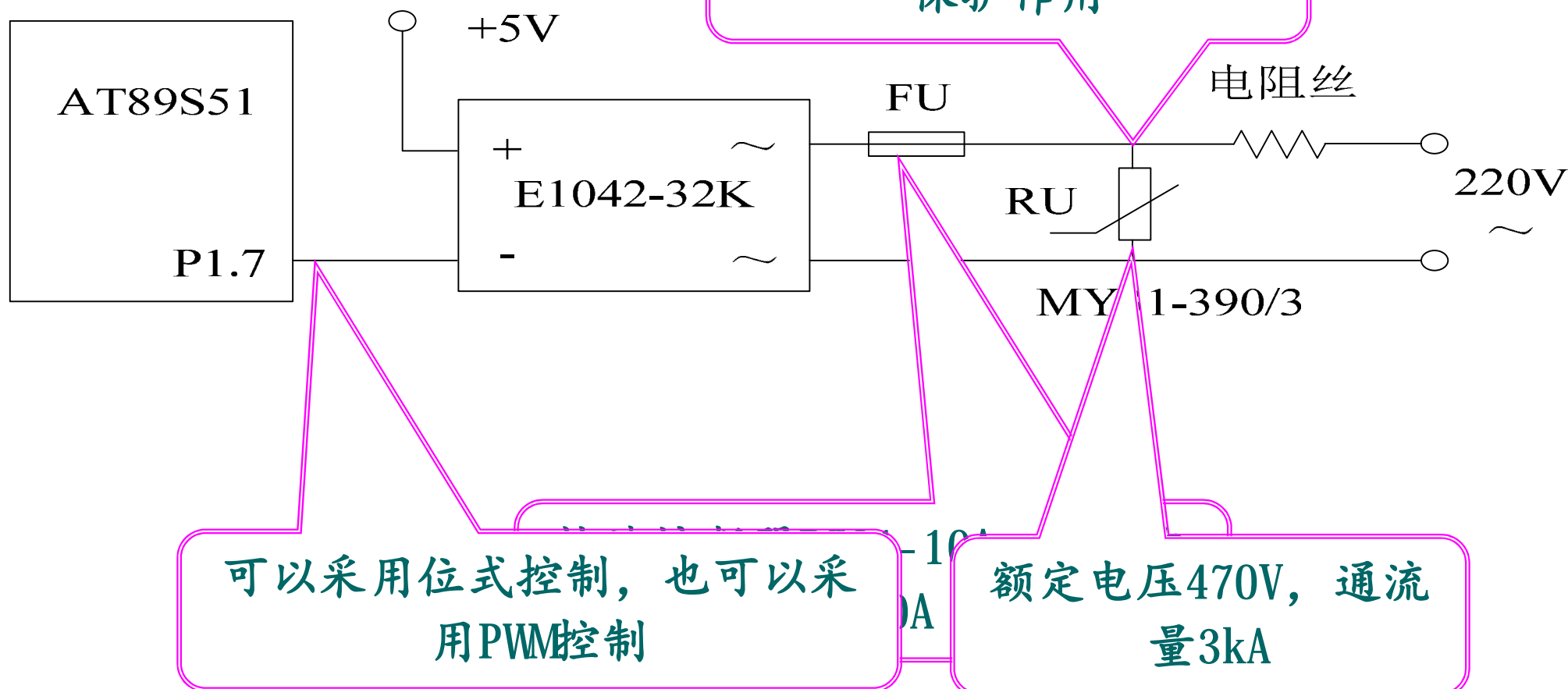
P1.7 输出低电平时 SSR 导通，
负载加热

负载功率 1kW



电路设计

压敏电阻起浪涌电压吸收
保护作用





06

END



THANKS



《单片机与接口技术》

主讲人：李刚