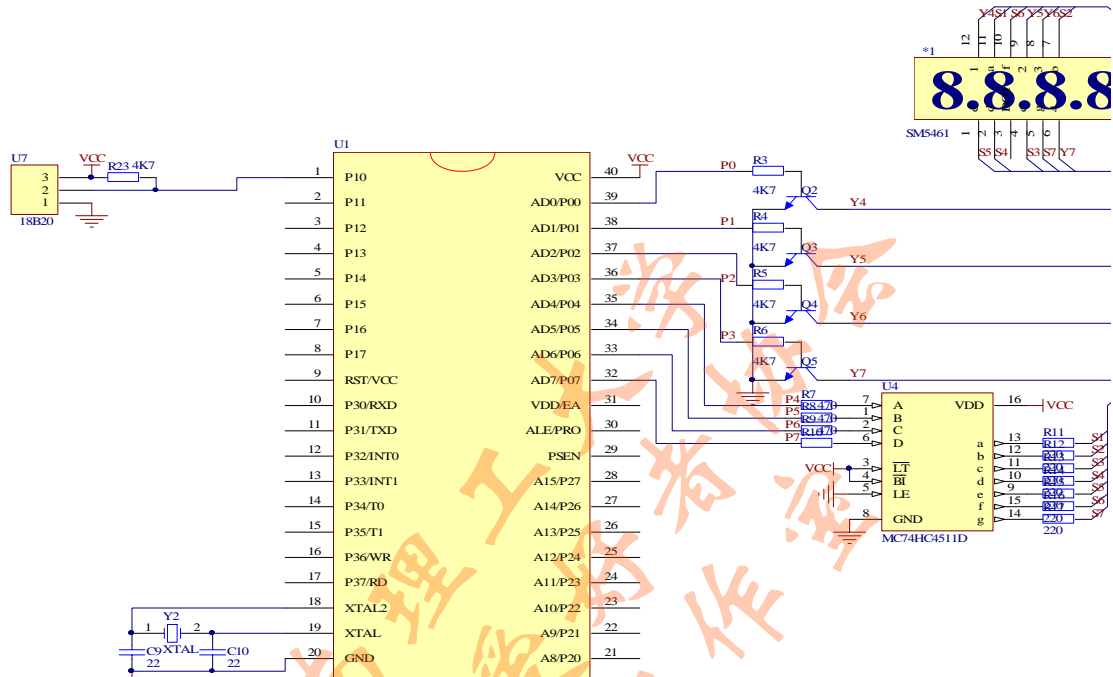


实验十五:DS18B20 温度计实验

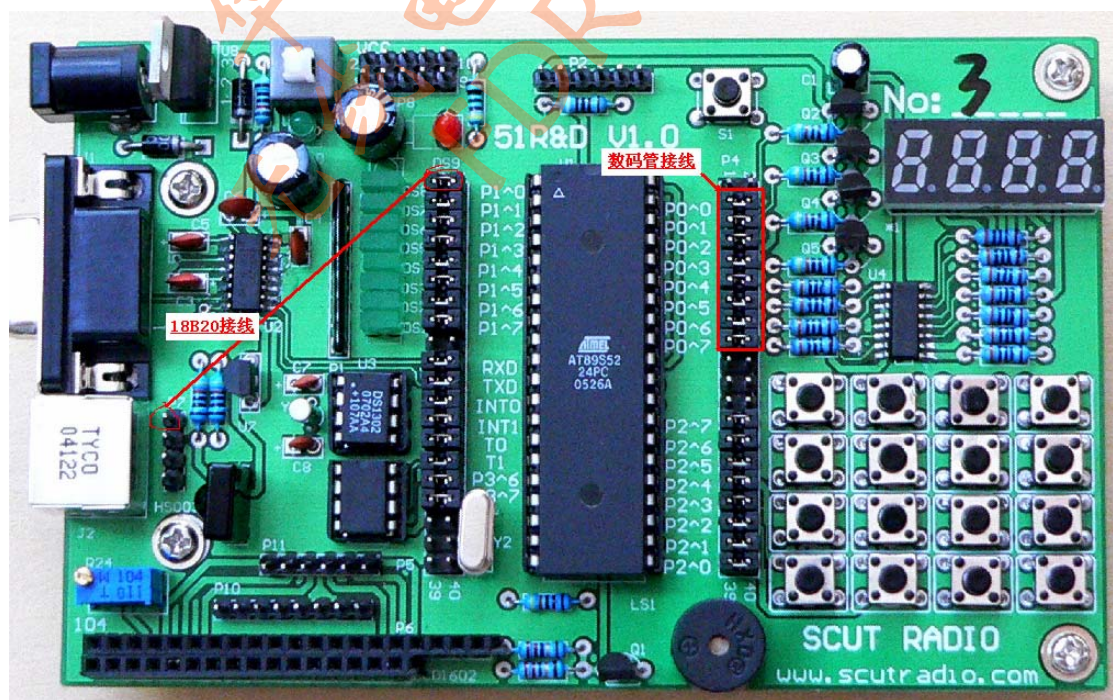
一、实验目的:

熟悉 DS18B20 及其一线传输协议

二、实验电路图:



三、硬件连线图:



四、实验原理:

DS18B20数字温度计是DALLAS公司生产的1-Wire, 即单总线器件, 具有线路简单, 体积小的特点。因此用它来组成一个测温系统, 具有线路简单, 在一根通信线, 可以挂很多这样的数字温度计。DS18B20产品的特点(1)、只要求一个I/O口即可实现通信。(2)、在DS18B20中的每个器件上都有独一无二的序列号。(3)、实际应用中不需要外部任何元器件即可实现测温。(4)、测量温度范围在-55.C到+125.C之间。(5)、数字温度计的分辨率用户可以从9位到12位选择。(6)、内部有温度上、下限告警设置。

DS18B20详细引脚功能描述1 GND地信号; 2 DQ数据输入/输出引脚。开漏单总线接口引脚。当被用着在寄生电源下, 也可以向器件提供电源; 3 VDD可选择的VDD引脚。当工作于寄生电源时, 此引脚必须接地。

DS18B20的使用方法。由于DS18B20采用的是1-Wire总线协议方式, 即在一根数据线实现数据的双向传输, 而对AT89S51单片机来说, 我们必须采用软件的方法来模拟单总线的协议时序来完成对DS18B20芯片的访问。由于DS18B20是在一根I/O线上读写数据, 因此, 对读写的数据位有着严格的时序要求。DS18B20有严格的通信协议来保证各位数据传输的正确性和完整性。该协议定义了几种信号的时序: 初始化时序、读时序、写时序。所有时序都是将主机作为主设备, 单总线器件作为从设备。而每一次命令和数据的传输都是从主机主动启动写时序开始, 如果要求单总线器件回送数据, 在进行写命令后, 主机需启动读时序完成数据接收。数据和命令的传输都是低位在先。

五、实验代码:

```
#include <reg52.h> //单片机的头文件

#define LEDPORT P0 //数码管接线
sbit DS18B20=P1^0; //DS18B20 接线口

unsigned char L_18B20,H_18B20; //分别存放低八位 高八位的数据;
unsigned char fg=0,zhengshu; //fg 为负数标志位,zhengshu 是温度的整数部分,
unsigned char xiaoshu1 ,xiaoshu2; //xiaoshu1 是十分位,xiaoshu2 是百分位

void Reset_18B20(void); //ds18b20 复位程序
void Wait_18B20(); //等待程序, 等待 ds18b20 应答
bit Readbit_18B20(void); //读取一个 bit 的数据
unsigned char Readbyte_18B20(void); //读取一个字节的数
void Writebyte_18B20(unsigned char b); //写一个字节 (控制字)
void Convert(void); //开始转换
void Read_18B20(void); //读取 ds18b20

void Reset_18B20(void)//ds18b20 复位程序
{
    unsigned char i;
    DS18B20=0;
    i=110;
    while(i>0)i--;
```

```
    DS18B20=1;
    i=6;
    while(i>0)i--;
}
```

void Wait_18B20()//等待程序，等待 ds18b20 应答

```
{
    unsigned char i;
    while(DS18B20);
    while(~DS18B20);
    i=6;
    while(i>0)i--;
}
```

bit Readbit_18B20(void) //读取一个 bit 的数据

```
{
    unsigned char i;
    bit b;
    DS18B20=0;
    i++;i++;
    DS18B20=1;
    i++;i++;i++;i++;i++;
    b=DS18B20;
    i=13;
    while(i>0)i--;
    return b;
}
```

unsigned char Readbyte_18B20(void) //读取一个字节的的数据

```
{
    unsigned char i,j,b;
    b=0;
    for(i=1;i<=8;i++)
    {
        j=Readbit_18B20();
        b=(j<<7)|(b>>1); //又玩野
    }
    return b;
}
```

void Writebyte_18B20(unsigned char b)//写一个字节（控制字）

```
{
    unsigned char j,i;
    bit temp;
```

```
for(j=1;j<=8;j++)
{
    temp=b&0x01;
    b=b>>1;
    if(temp) // 写“1”
    {
        DS18B20=0;
        i++;i++;i++;i++;i++;
        DS18B20=1;
        i=13;
        while(i>0)i--;
    }
    else //写“0”
    {
        DS18B20=0;
        i=13;
        while(i>0)i--;
        DS18B20=1;
        i++;i++;i++;i++;i++;
    }
}
}

void Convert(void) //开始转换
{
    Reset_18B20();
    Wait_18B20(); //等待应答脉冲
    Writebyte_18B20(0xcc); //跳过序列号脉冲
    Writebyte_18B20(0x44); //转换命令
}

void Read_18B20(void) //读取 ds18b20
{
    Convert();
    Reset_18B20();
    Wait_18B20();
    Writebyte_18B20(0xcc); //跳过 ROM
    Writebyte_18B20(0xbe); //发送读取命令
    L_18B20=Readbyte_18B20(); //读取低八位数据
    H_18B20=Readbyte_18B20(); //读取高八位数据
    if(H_18B20>0x7f) //最高位为 1 时温度是负
    {
        L_18B20=~L_18B20; //补码转换，取反加一
        H_18B20=~H_18B20+1;
    }
}
```

```
        fg=1;          //读取温度为负时 fg=1
    }
    zhengshu=L_18B20/16+H_18B20*16; //整数部分
    xiaoshu1=(L_18B20&0x0f)*10/16; //小数第一位
    xiaoshu2=(L_18B20&0x0f)*100/16%10; //小数第二位
}
/*-----数码管显示函数-----*/
参数:number 要显示的数字
参数:pos    要显示的位置(从左到右为 1--4)
返回值:无
-----*/
void display(uchar number,uchar pos)//数码管显示函数
{
    uchar temp;                //临时变量
    if(number<10 && pos<5 && pos>0)//确定数据合法
    {
        temp=(number<<4)&0xf0;//获得要显示的数据
        temp+=1<<(pos-1); //送位置位
        LEDPORT=temp;      ///送显示数据,开始显示
    }
    else return;
}

void leddisplay(unsigned int count)
{
    unsigned char i=500;
    while(i--)
    {
        if(count>=1000)
        {
            display(count/1000,1);
            delay_ms(1);
            display(count%1000/100,2);
            delay_ms(1);
            display(count%100/10,3);
            delay_ms(1);
            display(count%10,4);
            delay_10us(10);
        }
        else if(count>=100)
        {
            display(count/100,2);
            delay_ms(1);
            display(count%100/10,3);
        }
    }
}
```



```
        delay_ms(1);
        display(count%10,4);
        delay_10us(10);
    }
    else if(count>=10)
    {
        display(count/10,3);
        delay_ms(1);
        display(count%10,4);
        delay_10us(10);
    }
    else
    {
        display(count,4);
        delay_10us(10);
    }
}

void main()
{
    unsigned int temp=0;
    while(1)
    {
        temp=(zhengshu*10+xiaoshu1);/* 显示的温度精度达到0.1度
        Read_18B20();
        leddisplay(temp);
    }
}
```

六、效果图:

