

## 实验(六) 温度计

### 一,实验目的:

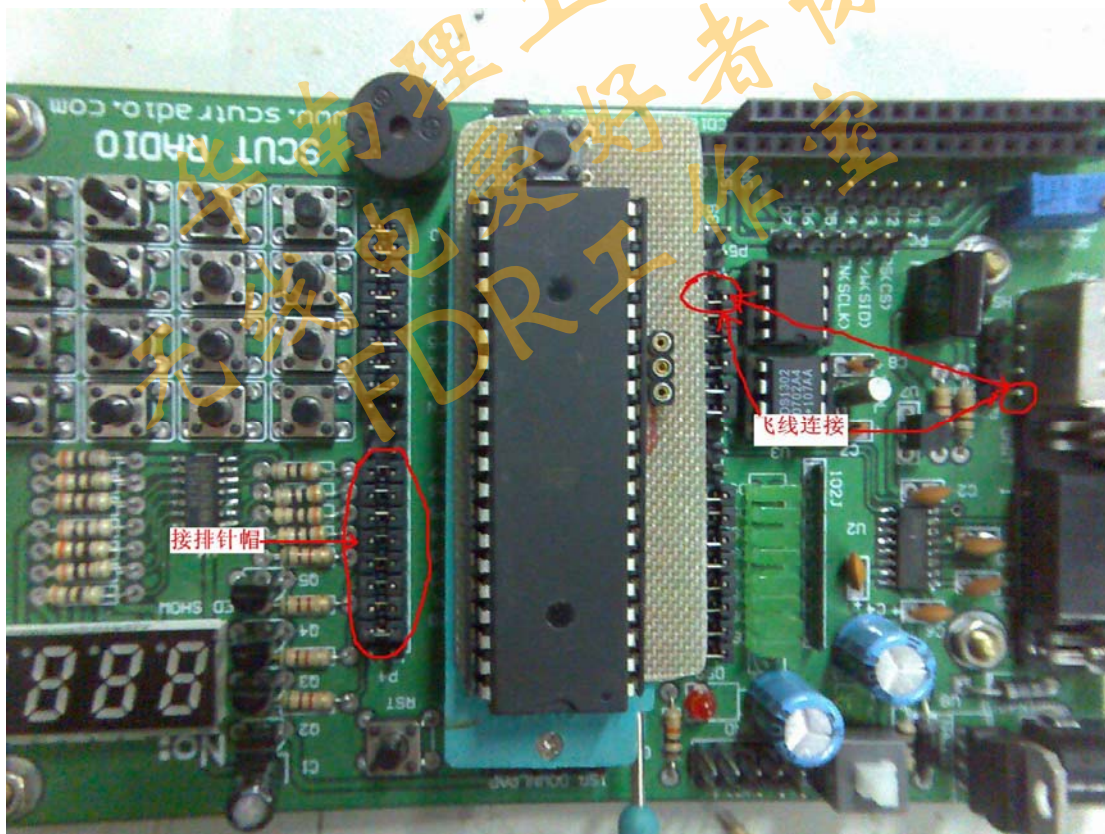
- 1,掌握温度传感器 DS18B20 的使用
- 2,进一步熟悉 AVR 的 IO 口使用
- 3,熟悉 CodeVisionAVR 编译器的使用

### 二,实验原理:

本实验接线方法:把板子上的 P3.7(PORTD.7)口和 P7 的第一个脚(温度传感器的数据口)用飞线连接上并用排针帽把 P0(PORTA)口的排针接上,详细原理图可以参考 CONVERTSCH.pdf 和实验板 总原理图.pdf 两个文档.

DS18B20 是一个数字温度传感器,采用一线数据传输结构,支持总线上挂多个相同器件,而且精度最高可以做到 12 位从-55 到正 125 度,精度为 0.0625 度,足见其性能.关于它的详细介绍可以参考我们给的器件资料.本次实验我们将用板上的温度传感器实现环境温度测量的功能.

### 三,硬件连接:



板子排针接法

本实验的具体接法见上图,把图中标记处用飞线连接上,同时将 P0(PORTA)口的排针帽接上就可以进行这个实验了

### 四,实验代码:

```
//晶振: 4MHz
//芯片:MEGA16L
#include "ds18b20.h"
//num,要显示的数据(0~9);pos,要显示的位置(从右到左)
void display(uchar num,uchar pos)
{
    DDRA=0XFF;//定义 PORTA 口输出
    PORTA=0XFF;//输出高电平
    switch(pos)
    {
        case 3:PORTA=0X01;break;
        case 2:PORTA=0X02;break;
        case 1:PORTA=0X04;break;
        case 0:PORTA=0X08;break;
        default:PORTA=0X00;
    }
    num<<=4;//取数字,送 CD4511
    PORTA|=num;
}
void main()
{
    uint tem=0;
    uchar t=0;
    while(1)
    {
        //display(tem%10,0);delay_ms(5);
        //display((tem/10)%10,1);delay_ms(5);
        display((tem/10)%10,2);delay_ms(5);//显示温度(摄氏度)
        display((tem/100),3);delay_ms(5);
        t++;
        if(t==100){tem=get1820data();t=0;}//大概两秒钟读一次
    }
}
```

以下是 ds18b20.h 中的代码:

```
#ifndef _ds18b20_INCLUDED_
#define _ds18b20_INCLUDED_
#include <mega16.h>
#include <delay.h>

#define DDRT DDRD.7 //DS18B20 引脚定义
#define TODA PORTD.7 //数据输出
#define TIDA PIND.7 //数据输入
uchar tempm=0;//温度正负标记
```

```
//reset ds1820
void res1820 (void)// Reset TX
{
    DDRT=1; //SET PA0 OUTPUT
    TODA=0; //输出底电平
    delay_us(750); //Approx 750 uS
    TODA=1; //PA0 1
    delay_us(15); //15US
}
//wait for presence of 1820
void check1820 (void)// Wait for Presence RX
{
    DDRT=0; //SET PA0 INPUT
    while (TIDA);
    while (!TIDA);
    delay_us(40);
}
//read bit data from 1820
unsigned char read1820bit (void)// read one bit
{
    unsigned char dat;
    DDRT=1; //SET PA0 OUTPUT
    TODA=0;
    delay_us(2);
    TODA=1;
    DDRT=0; //SET PA0 INPUT
    delay_us(2);
    dat=0X01&TIDA;
    delay_us(40);
    return (dat);
}
//read one byte from 1820
unsigned char read1820byte (void)// read one byte
{
    unsigned char i,j,dat;
    dat=0;
    for(i=1;i<=8;i++)
    {
        j=read1820bit();
        dat =(j<<7)|(dat>>1);
    }
    return dat;
}
//write one byte to 1820
```

```
void write1820byte (unsigned char dat)// write one byte
{
    unsigned char j;
    unsigned char testb;
    DDRT=1; //SET PA0 OUTPUT;
    for (j=1;j<=8;j++)
    {
        testb=dat&0x01;
        dat=dat>>1;
        if (testb)
        {
            TODA=0; // Write 1
            delay_us(2);
            TODA=1;
            delay_us(45);
        }
        else
        {
            TODA=0; // Write 0
            delay_us(45);
            TODA=1;
            delay_us(2);
        }
    }
}
//start conver temperature
void start1820 (void) // ds1820 start convert
{
    res1820 ();//复位 1820
    check1820 ();
    write1820byte (0xcc);// skip rom
    write1820byte (0x44);// convert
}
//read temperature from1820
//TH,high 4 bit of temperature
//TL,low 8 bit of temperature
void read1820 (unsigned char *TH,unsigned char *TL) // read temp
{
    res1820 ();//复位 1820
    check1820 ();
    write1820byte (0xcc);// skip rom
    write1820byte (0xbe);// convert
    *TL=read1820byte(); // LSB
    *TH=read1820byte(); // MSB
}
```

```
}  
//读取 ds18b20 的温度值,返回无符号整型数据.  
uint get1820data(void)  
{  
    unsigned char *TL,*TH;  
    uint tem=0;  
    start1820 ();    // ds1820 start convert  
    read1820 (TH,TL); // read temperature  
    if(*TH>7)  
    {  
        *TH=~*TH;  
        *TL=~*TL;  
        tempm=0;//温度为负  
    }  
    else tempm=1;//温度为正  
    tem=*TH; //获得高八位  
    tem<<=8;  
    tem+=*TL;//获得底八位  
    tem=(float)tem*0.625;//转换  
    return tem;    //返回温度值  
}  
#endif
```

下载程序之后,你会看到当前的室内温度显示,每两秒钟读取一次.注意 tempm 是温度正负的标记,在这个实验里没有用到.不过你在需要的时候可以加个判断,就可以知道温度的正负了.

华南理工大学协会  
无线电爱好者协会  
FDR工作室