**设计（论文）题目**

---ZE08-CH2O 甲醛浓度检测计---

**院（系）：电子与信息工程学院**

**学生姓名：**

**专业班级：**

**学号：**

**指导教师：**

2018 年 5月18日

目 录

[目录 1](#_Toc515023310)

[第一册产品使用说明书 2](#_Toc515023311)

[1.产品概述 2](#_Toc515023312)

[2.技术参数 3](#_Toc515023313)

[3.操作说明 3](#_Toc515023315)

[4.使用事项 3](#_Toc515023316)

[5.使用环境 4](#_Toc515023317)

[第二册设计文件 5](#_Toc515023318)

[1.工艺流程图 5](#_Toc515023319)

[2.元件清单 6](#_Toc515023320)

[第三册技术文件 7](#_Toc515023321)

[1.电路结构 7](#_Toc515023322)

[2.电路图 10](#_Toc515023323)

[3.C语言程序 13](#_Toc515023324)

[4.系统框图 23](#_Toc515023325)

[5.产品创新点 24](#_Toc515023326)

[6.技术参数 25](#_Toc515023327)

# 第一册 产品使用说明书

1.产品概述

基于STC89C52的ZE08-CH2O 甲醛浓度检测计的主要包括：检测的当地空气中的甲醛浓度、显示、报警这三大部分组成。甲醛浓度检测是利用ZE08-CH2O型甲醛传感器作为感应部件，再将检测到的当地空气中的甲醛浓度数据后反馈给芯片。在通过LCD1602进行实时监控显示浓度值。此产品操作简单方便，为人们提供了方便和保障。

**关键词：**甲醛浓度、STC89C52、实时监控、报警

**引言:**

众所周知，甲醛是一种有特殊气味，对人眼、鼻等有刺激作用。只要大于0.08m^3浓度的甲醛对人体健康有很大的危害，在工厂、劣质家具、房屋装修等所留有害气体主要为甲醛。如哮喘，白血病等许多疾病的诱发都与甲醛有关。针对此，设计了一种用于检测室內气体所含甲醛含量的检测系统。该检测系统具有显示甲醛含量，而且一旦甲醛的含量超出设定的值时就会触发报警。该系统设计方案是基于STC89C52单片机，选择的是采用高品质的ZE08-CH2O型甲醛传感器。在系统中甲醛传感器检测空气中的甲醛含量数据，以电信号的形式传送到单片机中，单片机再对此信号进行处理，再将甲醛含量数据显示在LCD1602上，而且增加的报警装置用于直观体现只要甲醛含量超标时就会进行报警提醒让你迅即采取有效措施。本设计体积小，集成度高，操作简单，具有很高的实用价值。

采用单片机对甲醛气体进行控制处理不仅方便、简单，从而减少甲醛对人体的伤害，因此如何利用单片机实时进行对甲醛数据的采集成为一个重要的研究课题。

1. **系统工作原理**

甲醛浓度的检测是采用ZE08-CH2O型甲醛传感器作为感应部件，将传感器检测到的甲醛浓度数据以电信号的形式反馈给STC89C52单片机中，单片机再对此信号进行处理计算，再将甲醛含量数据显示在LCD1602上显示，报警装置用于直观体现只要甲醛含量超标时就会进行报警提醒。

1. **系统组成**

ZE08-CH2O 甲醛浓度检测计主要是由STC89C52最小系统、ZE08-CH2O型甲醛传感器、按键调节、LCD1602和蜂鸣器组成。

2.技术参数

1. 工作电压：直流5V（常用USB接口）
2. 工作电流：10~15mA
3. 甲醛范围：0.0000～5.0000mg/m^3（误差+0.0001mg/m^3）

3.操作说明

1. 上电说明：电路板上排针通过TTL杜邦线直接接连接上DC5V的电源（如手机充电器、电脑、移动电源等USB接口）即可；
2. 按键设置说明：通过两个机械按键进行控制，分别为“加”、“减”来甲醛浓度的上限值（长按可快速增加），达到报警阀值可控性；

4.使用事项

1. 产品电路板远离水源，切勿让水流入内部。
2. 操作该产品前，应该阅读该产品使用说明书
3. 本产品已固定使用电压为直流5伏，切勿使用超过5伏，以防将单片机烧坏。
4. 在非使用期间，应该将产品断电，减少产品寿命损耗。
5. 切勿让3岁幼儿以下玩弄，以保证人身安全。

5.使用环境

1. 产品正常使用环境温度应在0℃至50℃，标准使用温度为25℃。
2. 产品电路板正常使用湿度应在15%RH-90％RH（无凝结），其标准使用湿度为50%。
3. 产品存储的温度为 0～50℃
4. 产品不应在特殊场合使用：强酸强碱、腐蚀性气体、特高温环境等等。
5. 产品应该置放在3岁以下幼儿触碰不到的位置。

# 第二册 设计文件

1. 工艺流程图

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **工 艺 流 程 图** | | | | | 产品名称 | | | | 产品图号 | |
| ZE08-CH2O 甲醛浓度检测计 | | | | 018 | |
| 元器件选择  进行元器件布局  测试电源电压  焊接元器件  测试ZE08-CH  2O型甲醛传感器数据传输  报警装置判断是否超阀值，  实时监控显示  按键设置阀值  整体功能完善  整体组装 | | | | | | | | | | |
| 旧底图  总号 | 更改  标记 | 数量 | 更改  单号 | 签名 | 日期 | |  | 签名 | 日期 | | 第3页 |
|  |  |  |  |  |  | | 拟制 | 018A | 2018/5/20 | | 共7页 |
| 底图  总号 |  |  |  |  |  | | 审核 | 018B | 2018/5/20 | | 第1册 |
|  |  |  |  |  |  | | 标准化 | 018C | 2018/5/20 | | 共1册 |

1. 元件清单

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **元 器 件 清 单** | | | | | | | 产品名称 | | | | | 产品图号 | |
| ZE08-CH2O 甲醛浓度检测计 | | | | | 018 | |
| 序号 | 器件类型 | | | 器件参数 | | | | | | 数量 | | 备注 | |
| 1 | 电容 | | | 10uf电解电容 | | | | | | 1 | |  | |
| 2 | 电容 | | | 30pf瓷片电容 | | | | | | 2 | |  | |
| 3 | 电源接口 | | | DC座 | | | | | | 1 | |  | |
| 4 | 按键 | | | 轻触按键 | | | | | | 3 | |  | |
| 5 | 绿色led灯 | | | 3mm | | | | | | 1 | |  | |
| 6 | 红色led灯 | | | 3mm | | | | | | 1 | |  | |
| 7 | 蜂鸣器 | | | 5V有源 | | | | | | 1 | |  | |
| 8 | 液晶显示屏 | | | LCD1602 | | | | | | 1 | |  | |
| 9 | 甲醛传感器 | | | ZE08-CH2O | | | | | | 1 | |  | |
| 10 | 排针 | | |  | | | | | | 1 | |  | |
| 11 | 排阻 | | | 10K | | | | | | 1 | |  | |
| 12 | 三极管 | | | 8550 | | | | | | 1 | |  | |
| 13 | 电阻 | | | 3K | | | | | | 1 | |  | |
| 14 | 电阻 | | | 330欧 | | | | | | 1 | |  | |
| 15 | 电阻 | | | 10K | | | | | | 1 | |  | |
| 16 | 电阻 | | | 1K | | | | | | 2 | |  | |
| 17 | 开关 | | | 自锁开关 | | | | | | 1 | |  | |
|  | 单片机 | | | STC89C52 | | | | | | 1 | |  | |
|  | 晶振 | | | 11.0592M | | | | | | 1 | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
|  |  | | |  | | | | | |  | |  | |
| 旧底图  总号 | 更改  标记 | | 数量 | 更改  单号 | | 签名 | 日期 | |  | 签名 | | 日期 | | 第4页 |
|  |  | |  |  | |  |  | | 拟制 | 018A | | 2018/5/20 | | 共7页 |
| 底图  总号 |  | |  |  | |  |  | | 审核 | 018B | | 2018/5/20 | | 第1册 |
|  |  | |  |  | |  |  | | 标准化 | 018C | | 2018/5/20 | | 共1册 |

# 第三册 技术文件

1.电路结构

* 1. **STC89C52最小系统**

STC89C52最小系统由STC89C52芯片、下载电路、时钟电路、电源电路、复位电路构成。STC89C52最高工作时钟频率为 80MHz，片内含存储器 8K Bytes ，致辞反复擦写 1000次，而且此单片机价格便宜、易操作。在传统的单片机中，一般芯片需要采用外部扩展存储，再让单片机存储器进行IIC或者SPI等接口来数据通信与，而STC89C52本身具有4K的EEPROM，而且EEPROM的速度要比其他外部存储读取速度快得多，在此产品中我们就是采用了单片机的EEPROM功能来进行断电保存数据。

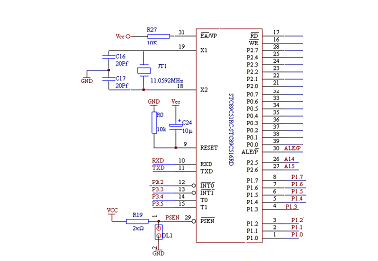


图1.1 单片机最小系统

* 1. **ZE08-CH2O型甲醛传感器**

ZE08-CH2O型电化学甲醛模组是一个具有高灵敏度、高稳定性、抗干扰能力强特点的通用型、小型化模组。利用电化学原理对空气中存在的CH2O进行探测，提供 UART、模拟电压信号、 PWM 波形等多种输出方式，以电信号的形式反馈数据到单片机上，方便使用和处理。ZE08-CH2O的电化学检测技术是十分精良的，适合于精密检测的设计中，主要运用于便携式仪表、空调、智能家居等场所中。

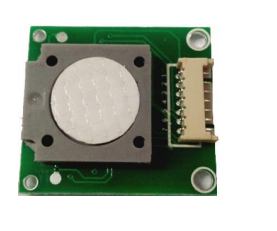


图1.2-1 实物图

①接线说明

与单片机之间连接

pin4接单片机的正极（VCC） pin3接单片机的负极（GND）

pin5接单片机TXD(P3.1) pin6接单片机的RXD（P3.0）

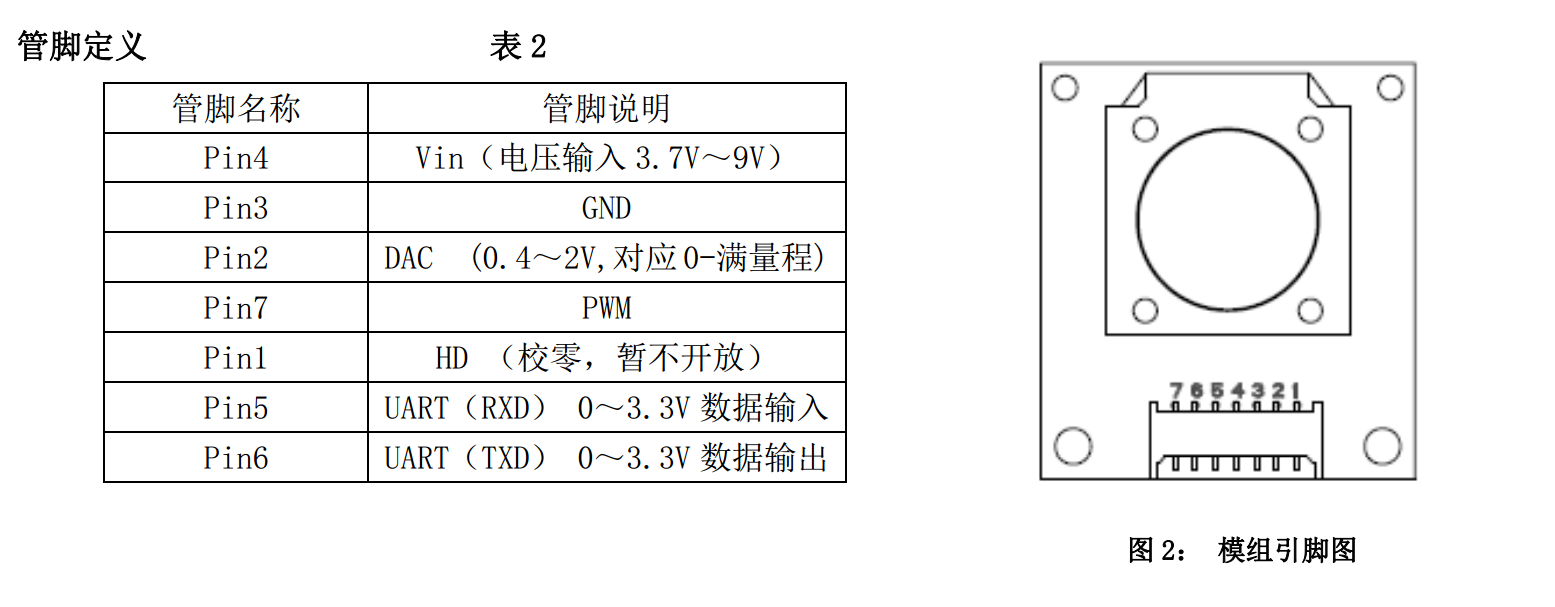
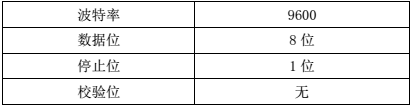
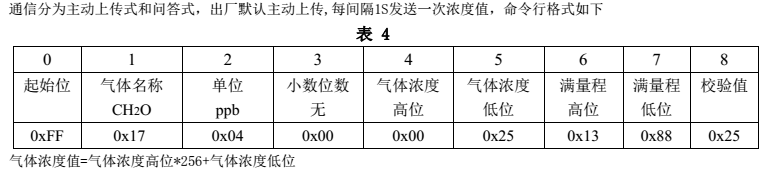


图1.2-2 管脚定义图

②通用设置



③通讯命令



⑤测试注意事项

1.串口TTL通信接线说明：传感器的TXD（pin6）接单片机的RXD，传感器的RXD(pin5)接单片机的TXD ，传感器的VCC（pin4）连接单片机5v正极，GND(pin3)连接单片机的GND

2.测量电压应选用5V 供电

3.单片机型号STC89C52，液晶显示屏型号LCD1602，电路连接按照那个测试图的

4.串口通信波特率设为9600，注意晶振要是11.0592M，否则会引起通信失败

5.传感器上电需要预热5分钟，不然会影响到精确度



图1.2-3 串口测试图

2.电路图

* 1. **电路设计**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **电 路 原 理 图** | | | | | 产品名称 | | | | 产品图号 | |
| ZE08-CH2O 甲醛浓度检测计 | | | | 018 | |
| 原理图    PCB图  C:\Users\Jemin\Desktop\毕业论文（1）\66 ZE08-CH2O 甲醛浓度检测完整资料\仿真\仿真效果图1.png  仿真效果展示 | | | | | | | | | | |
| 旧底图  总号 | 更改  标记 | 数量 | 更改  单号 | 签名 | 日期 | |  | 签名 | 日期 | | 第3页 |
|  |  |  |  |  |  | | 拟制 | 018A | 2018/5/20 | | 共13页 |
| 底图  总号 |  |  |  |  |  | | 审核 | 018B | 2018/5/20 | | 第1册 |
|  |  |  |  |  |  | | 标准化 | 018C | 2018/5/20 | | 共1册 |

* 1. **程序流程设计**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **程 序 流 程 图** | | | | | 产品名称 | | | | 调试项目 | |
| ZE08-CH2O 甲醛浓度检测计 | | | | 018 | |
| 开始  初始化  采集甲醛浓度并处理  LCD显示甲醛浓度和上下阀值  **否**  记录甲醛数据  报警  按键调节阀值  **是** | | | | | | | | | | |
| 旧底图  总号 | 更改  标记 | 数量 | 更改  单号 | 签名 | 日期 | |  | 签名 | 日期 | | 第5页 |
|  |  |  |  |  |  | | 拟制 | 018A | 2018/5/20 | | 共13页 |
| 底图  总号 |  |  |  |  |  | | 审核 | 018B | 2018/5/20 | | 第1册 |
|  |  |  |  |  |  | | 标准化 | 018C | 2018/5/20 | | 共1册 |

1. C语言程序

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **程 序 清 单** | | | | | 产品名称 | | | | 调试项目 | |
| ZE08-CH2O 甲醛浓度检测计 | | | | 018 | |
| 1. **主函数main：**   **#include<reg52.h> //头文件**  **#include<intrins.h> //头文件**  **#include"eeprom52.h"**  **#define uchar unsigned char //宏定义**  **#define uint unsigned int**  **#define LCD1602\_dat P0**  **sbit LCD1602\_rs=P2^5;//IO 定义**  **sbit LCD1602\_rw=P2^6;**  **sbit LCD1602\_e=P2^7;**  **sbitbeepio=P2^0; //蜂鸣器IO**  **sbitLed\_1=P1^0; //指示灯IO**  **sbitLed\_2=P1^1;**  **sbitkey\_11=P1^2; //按键控制IO**  **sbitkey\_12=P1^3;**  **uint ch2o\_H,ch2o; //甲醛上限变量，甲醛浓度变量**  **ucharbz,ms,sec,UART\_dat[10],zhi[2],i; //暂存数据切换变量， 50模式变量，串口暂存数据数组，甲醛浓度高、低位数组，循环变量**  **bit beepio1;**  **/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***  **\* 名称 : delay111()**  **\* 功能 : 小延时。**  **\* 输入 : 无**  **\* 输出 : 无**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**  **void delay111(uint T) //延时函数**  **{**  **while(T--);**  **}**  **/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***  **\* 名称 : LCD1602\_write(ucharorder,dat)**  **\* 功能 : 1602写如数据函数**  **\* 输入 : 输入的命令值**  **\* 输出 : 无**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**  **void LCD1602\_write(ucharorder,dat) //1602 一个字节处理**  **{**  **LCD1602\_e=0;**  **LCD1602\_rs=order;**  **LCD1602\_dat=dat;**  **LCD1602\_rw=0;**  **LCD1602\_e=1;**  **delay111(1);**  **LCD1602\_e=0;**  **}**  **/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***  **\* 名称 : LCD1602\_writebye(uchar \*prointer)**  **\* 功能 : 1602写入数据函数指针式**  **\* 输入 : 输入的命令值**  **\* 输出 : 无**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**  **void LCD1602\_writebyte(uchar \*prointer) //1602 字符串处理**  **{**  **while(\*prointer!='\0')**  **{**  **LCD1602\_write(1,\*prointer);**  **prointer++;**  **}**  **}**  **/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***  **\* 名称 : LCD1602\_cls()**  **\* 功能 : 初始化1602液晶**  **\* 输入 : 无**  **\* 输出 : 无**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**  **void LCD1602\_cls() //1602 初始化**  **{**  **LCD1602\_write(0,0x01); //1602 清屏指令**  **delay111(1500);**  **LCD1602\_write(0,0x38); // 功能设置 8位、5\*7点阵**  **delay111(1500);**  **LCD1602\_write(0,0x0c); //设置光标不显示开关、不显示光标、字符不闪烁**  **LCD1602\_write(0,0x06);**  **LCD1602\_write(0,0xd0);**  **delay111(1500);**  **}**  **void show() //显示数据**  **{**  **LCD1602\_write(0,0x80);**  **LCD1602\_writebyte("CH2O:"); //显示当前的甲醛浓度值**  **LCD1602\_write(1,0x30+ch2o/10000%10);**  **LCD1602\_writebyte(".");**  **LCD1602\_write(1,0x30+ch2o/1000%10);**  **LCD1602\_write(1,0x30+ch2o/100%10);**  **LCD1602\_write(1,0x30+ch2o/10%10);**  **LCD1602\_write(1,0x30+ch2o%10);**  **LCD1602\_writebyte("mg/m3");**  **LCD1602\_write(0,0xc0);**  **LCD1602\_writebyte("CH2O\_H:"); //显示甲醛上限值**  **LCD1602\_write(1,0x30+ch2o\_H/100%10);**  **LCD1602\_writebyte(".");**  **LCD1602\_write(1,0x30+ch2o\_H/10%10);**  **LCD1602\_write(1,0x30+ch2o\_H%10);**  **LCD1602\_writebyte("mg/m3"); //毫克每立方米**  **}**  **void proc() //超限处理**  **{**  **if(ch2o/100>=ch2o\_H) //如果当前甲醛浓度超出甲醛浓度上限**  **{**  **Led\_1=1; //对应指示灯点亮**  **Led\_2=0;**  **beepio1=1; //开启蜂鸣器报警**  **}else //否则**  **{**  **Led\_1=0; //对应指示灯点亮**  **Led\_2=1;**  **beepio1=0; //开启蜂鸣器报警**  **}**  **}**  **void key()**  **{**  **if(!key\_11) //判断按下**  **{**  **ms=sec=0;**  **delay111(888); //按键延时去抖**  **if(!key\_11) //判断按下**  **{**  **if(ch2o\_H<700)ch2o\_H++;**  **ms=sec=0; //清楚按键按下计时重新计时**  **while(!key\_11) //按键判断释放**  **{**  **show();**  **proc();**  **}**  **SectorErase(0x2000); //保存上限值**  **byte\_write(0x2000,ch2o\_H/256);**  **byte\_write(0x2001,ch2o\_H%256);**  **}**  **}**  **if(!key\_12) //判断按下**  **{**  **ms=sec=0;**  **delay111(888); //按键延时去抖**  **if(!key\_12) //判断按下**  **{**  **if(ch2o\_H>0) ch2o\_H--;**  **ms=sec=0;**  **while(!key\_12) //按键判断释放**  **{**  **show();**  **proc();**  **}**  **SectorErase(0x2000); //保存上限值**  **byte\_write(0x2000,ch2o\_H/256);**  **byte\_write(0x2001,ch2o\_H%256);**  **}**  **}**  **}**  **void main()**  **{**  **ucharh,l;**  **LCD1602\_cls(); //LCD 1602 调用初始化程序**  **TMOD=0x21; //配置定时**  **TH0=0x4c; //50ms 定时赋值**  **TL0=0x00;**  **ET0=1;**  **TR0=1;**  **TH1=0xfd; //串口波特率定时初始**  **TL1=0xfd;**  **SCON=0x50; //只发送**  **EA=1; //打开定时总中断**  **ES=1; //打开串口中断**  **TR1=1;**  **h=byte\_read(0x2000);**  **l=byte\_read(0x2001); //读取保存的数据**  **ch2o\_H=h\*256+l;**  **if(ch2o\_H>700)**  **ch2o\_H=80;**  **while(1)**  **{**  **show(); //显示函数**  **key(); //按键处理程序**  **proc(); //超限处理程序**  **}**  **}**  **void UART\_1() interrupt 1**  **{**  **TH0=0x4c;**  **TL0=0x00;**  **ms++;**  **if(ms%2==0) //长按按键计数处理**  **{**  **if(sec>1)**  **{**  **if(!key\_11) //设置按键长按处理**  **{**  **if(ch2o\_H<700)ch2o\_H++;**  **}**  **if(!key\_12)**  **{**  **if(ch2o\_H>0) ch2o\_H--;**  **}**  **}**  **}**  **if(ms%10==0)**  **{**  **if(beepio1==1) //蜂鸣器报警处理**  **{**  **beepio=!beepio;**  **}else**  **{**  **beepio=1;**  **}**  **}**  **if(ms>19) //1s 定时**  **{**  **ms=0;**  **sec++;**    **}**  **}**  **void UART\_4() interrupt 4 //处理接收的串口数据**  **{**  **if(RI)**  **{**  **RI=0; //RI 置0**  **UART\_dat[bz]=SBUF; //暂存串口数据**  **switch(bz)**  **{**  **case 0:**  **if(UART\_dat[bz]==0xFF)**  **{**  **bz=1;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **case 1:**  **if(UART\_dat[bz]==0x17)**  **{**  **bz=2;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **case 2:**  **if(UART\_dat[bz]==0x04)**  **{**  **bz=3;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **case 3:**  **if(UART\_dat[bz]==0x00)**  **{**  **bz=4;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **case 4:**  **bz=5;**  **break;**  **case 5:**  **bz=6;**  **break;**  **case 6:**  **if(UART\_dat[bz]==0x13)**  **{**  **bz=7;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **case 7:**  **if(UART\_dat[bz]==0x88)**  **{**  **bz=8;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **case 8: //校验接收的数据正确则处理接收的数据**  **if((UART\_dat[bz]+UART\_dat[1]+UART\_dat[2]+UART\_dat[3]+UART\_dat[4]+UART\_dat[5]+UART\_dat[6]+UART\_dat[7])%256==0)**  **{**  **ch2o=UART\_dat[4]\*256+UART\_dat[5];**  **ch2o=ch2o\*13.392857;**  **bz=0;**  **}else**  **{**  **bz=0;**  **}**  **break;**  **}**  **}**  **}**   1. **EEPROM程序**   **#ifndef \_EEPROM52\_H\_**  **#define \_EEPROM52\_H\_**  **#include <intrins.h>**  **/\*\*\*\*\*\*\*\*STC89C51扇区分布\*\*\*\*\*\*\***  **第一扇区：1000H--11FF**  **第二扇区：1200H--13FF**  **第三扇区：1400H--15FF**  **第四扇区：1600H--17FF**  **第五扇区：1800H--19FF**  **第六扇区：1A00H--1BFF**  **第七扇区：1C00H--1DFF**  **第八扇区：1E00H--1FFF**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**  **/\*\*\*\*\*\*\*\*STC89C52扇区分布\*\*\*\*\*\*\***  **第一扇区：2000H--21FF**  **第二扇区：2200H--23FF**  **第三扇区：2400H--25FF**  **第四扇区：2600H--27FF**  **第五扇区：2800H--29FF**  **第六扇区：2A00H--2BFF**  **第七扇区：2C00H--2DFF**  **第八扇区：2E00H--2FFF**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**  **#define RdCommand 0x01 //定义ISP的操作命令**  **#define PrgCommand 0x02**  **#define EraseCommand 0x03**  **#define Error 1**  **#define Ok 0**  **#define WaitTime 0x01 //定义CPU的等待时间**  **sfr ISP\_DATA=0xe2; //寄存器申明**  **sfr ISP\_ADDRH=0xe3;**  **sfr ISP\_ADDRL=0xe4;**  **sfr ISP\_CMD=0xe5;**  **sfr ISP\_TRIG=0xe6;**  **sfr ISP\_CONTR=0xe7;**  **/\* ================ 打开 ISP,IAP 功能 ================= \*/**  **void ISP\_IAP\_enable(void)**  **{**  **EA = 0; /\* 关中断 \*/**  **ISP\_CONTR = ISP\_CONTR & 0x18; /\* 0001,1000 \*/**  **ISP\_CONTR = ISP\_CONTR | WaitTime; /\* 写入硬件延时 \*/**  **ISP\_CONTR = ISP\_CONTR | 0x80; /\* ISPEN=1 \*/**  **}**  **/\* =============== 关闭 ISP,IAP 功能 ================== \*/**  **void ISP\_IAP\_disable(void)**  **{**  **ISP\_CONTR = ISP\_CONTR & 0x7f; /\* ISPEN = 0 \*/**  **ISP\_TRIG = 0x00;**  **EA = 1; /\* 开中断 \*/**  **}**  **/\* ================ 公用的触发代码 ==================== \*/**  **void ISPgoon(void)**  **{**  **ISP\_IAP\_enable(); /\* 打开 ISP,IAP 功能 \*/**  **ISP\_TRIG = 0x46; /\* 触发ISP\_IAP命令字节1 \*/**  **ISP\_TRIG = 0xb9; /\* 触发ISP\_IAP命令字节2 \*/**  **\_nop\_();**  **}**  **/\* ==================== 字节读 ======================== \*/**  **unsigned char byte\_read(unsigned intbyte\_addr)**  **{**  **EA = 0;**  **ISP\_ADDRH = (unsigned char)(byte\_addr>> 8);/\* 地址赋值 \*/**  **ISP\_ADDRL = (unsigned char)(byte\_addr& 0x00ff);**  **ISP\_CMD = ISP\_CMD & 0xf8; /\* 清除低3位 \*/**  **ISP\_CMD = ISP\_CMD | RdCommand; /\* 写入读命令 \*/**  **ISPgoon(); /\* 触发执行 \*/**  **ISP\_IAP\_disable(); /\* 关闭ISP,IAP功能 \*/**  **EA = 1;**  **return (ISP\_DATA); /\* 返回读到的数据 \*/**  **}**  **/\* ================== 扇区擦除 ======================== \*/**  **void SectorErase(unsigned intsector\_addr)**  **{**  **unsigned intiSectorAddr;**  **iSectorAddr = (sector\_addr& 0xfe00); /\* 取扇区地址 \*/**  **ISP\_ADDRH = (unsigned char)(iSectorAddr>> 8);**  **ISP\_ADDRL = 0x00;**  **ISP\_CMD = ISP\_CMD & 0xf8; /\* 清空低3位 \*/**  **ISP\_CMD = ISP\_CMD | EraseCommand; /\* 擦除命令3 \*/**  **ISPgoon(); /\* 触发执行 \*/**  **ISP\_IAP\_disable(); /\* 关闭ISP,IAP功能 \*/**  **}**  **/\* ==================== 字节写 ======================== \*/**  **void byte\_write(unsigned intbyte\_addr, unsigned char original\_data)**  **{**  **EA = 0;**  **// SectorErase(byte\_addr);**  **ISP\_ADDRH = (unsigned char)(byte\_addr>> 8); /\* 取地址 \*/**  **ISP\_ADDRL = (unsigned char)(byte\_addr& 0x00ff);**  **ISP\_CMD = ISP\_CMD & 0xf8; /\* 清低3位 \*/**  **ISP\_CMD = ISP\_CMD | PrgCommand; /\* 写命令2 \*/**  **ISP\_DATA = original\_data; /\* 写入数据准备 \*/**  **ISPgoon(); /\* 触发执行 \*/**  **ISP\_IAP\_disable(); /\* 关闭IAP功能 \*/**  **EA =1;**  **}**    **#endif** | | | | | | | | | | |
| 旧底图  总号 | 更改  标记 | 数量 | 更改  单号 | 签名 | 日期 | |  | 签名 | 日期 | | 第12页 |
|  |  |  |  |  |  | | 拟制 | 018A | 2018/5/20 | | 共13页 |
| 底图  总号 |  |  |  |  |  | | 审核 | 018B | 2018/5/20 | | 第1册 |
|  |  |  |  |  |  | | 标准化 | 018C | 2018/5/20 | | 共1册 |

1. 系统框图

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **系统框图** | | | | | 产品名称 | | | | 调试项目 | |
| ZE08-CH2O 甲醛浓度检测计 | | | | 整体调试 | |
| ZE08-CH2O 甲醛传感器  核心板  设备电源  5V  LCD1602  显示现在的甲醛浓度和阀值  实时监测  浓度是否超阀值 | | | | | | | | | | |
| 旧底图  总号 | 更改  标记 | 数量 | 更改  单号 | 签名 | 日期 | |  | 签名 | 日期 | | 第13页 |
|  |  |  |  |  |  | | 拟制 | 018A | 2018/5/20 | | 共13页 |
| 底图  总号 |  |  |  |  |  | | 审核 | 018B | 2018/5/20 | | 第1册 |
|  |  |  |  |  |  | | 标准化 | 018C | 2018/5/20 | | 共1册 |

1. 产品创新点

本文设计的系统是基于STC12S5A60S2的带有存储器功能的数字温度计，该系统利用EEPROM来进行储存温度数据，大大的减少了生产成本。

面向群体:所有群体。

1. 技术参数
2. 输入电压：适配器5V输入
3. 是否含充电电池：含