# 基于STC89C52单片机的多功能温度监控报警系统设计与实现

（此处填写作者姓名）

（此处填写作者单位，邮编）

**摘要：** 本文设计并实现了一种以STC89C52单片机为核心的温度监控报警系统。该系统利用DS18B20数字温度传感器精确采集环境温度，通过LCD1602液晶显示模块实时显示当前温度以及预设的温度上下限阈值。用户可通过按键方便地调整温度阈值，阈值参数利用AT24C02 EEPROM进行非易失性存储。当检测到的温度超出设定的阈值范围时，系统将通过LCD显示和蜂鸣器发出声光报警信号。系统硬件电路设计简洁可靠，软件算法高效稳定，具有较高的实用价值和推广前景，可广泛应用于对环境温度有特定要求的场合，如仓库、机房、温室大棚等。

**关键词：** STC89C52；DS18B20；LCD1602；温度监控；阈值报警；AT24C02

**Design and Implementation of a Multifunctional Temperature Monitoring and Alarm System Based on STC89C52 Microcontroller**

**Abstract:** This paper designs and implements a temperature monitoring and alarm system with the STC89C52 microcontroller as its core. The system utilizes the DS18B20 digital temperature sensor to accurately collect ambient temperature, and displays the current temperature along with preset upper and lower temperature thresholds in real-time via an LCD1602 liquid crystal display module. Users can conveniently adjust the temperature thresholds using keys, and these threshold parameters are stored non-volatilely using AT24C02 EEPROM. When the detected temperature exceeds the set threshold range, the system will issue visual and audible alarm signals through the LCD display and a buzzer. The system's hardware circuit design is concise and reliable, and its software algorithms are efficient and stable, demonstrating high practical value and promising application prospects. It can be widely used in environments with specific temperature requirements, such as warehouses, server rooms, and greenhouses.

**Keywords:** STC89C52; DS18B20; LCD1602; Temperature Monitoring; Threshold Alarm; AT24C02

## 0 引言

随着现代工业技术和日常生活的飞速发展，温度作为一项关键的环境参数，在诸多领域如工业生产、仓储物流、农业养殖、智能家居等都扮演着至关重要的角色[1]。对温度进行精确、实时的监测，并在温度异常时及时发出警报，对于保障生产安全、提高产品质量、优化生活环境具有重要意义。传统的温度控制方法往往存在精度不高、响应迟缓、功能单一等问题，难以满足日益增长的智能化、自动化需求。

单片微型计算机（MCU）因其体积小、功耗低、功能强大、成本低廉且易于集成等优点，在测控领域得到了广泛应用[2]。STC89C52是一款广泛应用的8位单片机，具有丰富的I/O端口、定时器/计数器、串行通信接口等资源，非常适合用于开发嵌入式控制系统。DS18B20是一款常用的单总线数字温度传感器，具有测温精度高、接口简单、体积小巧等特点。LCD1602字符型液晶显示模块则因其显示直观、驱动方便而被广泛应用于各种人机交互界面。AT24C02是一款基于I2C总线的EEPROM存储芯片，能够实现数据的非易失性存储。

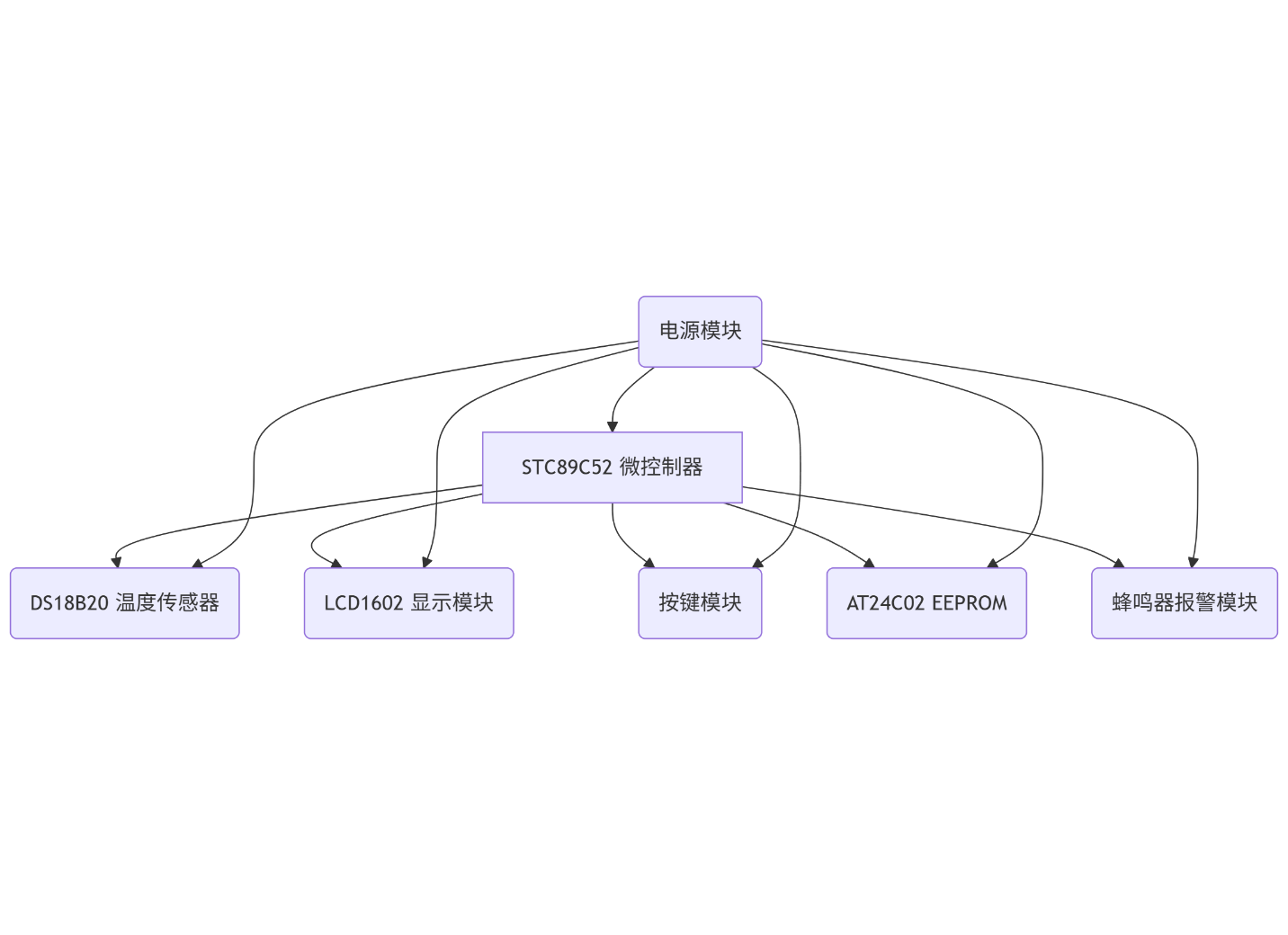
本文基于STC89C52单片机，结合DS18B20温度传感器、LCD1602显示模块、AT24C02存储芯片以及按键输入模块，设计并实现了一套能够实时监测温度、显示温度、设置温度上下限并进行超限报警的多功能温度监控报警系统。该系统旨在提供一种经济实用、稳定可靠的温度监控解决方案。

## 1 系统总体设计方案

本系统以STC89C52单片机为控制核心，主要由以下几个模块组成：

1. **主控制器模块**：采用STC89C52单片机，负责整个系统的协调控制、数据处理和逻辑判断。
2. **温度采集模块**：采用DS18B20数字温度传感器，通过单总线与单片机通信，实现对环境温度的精确测量。
3. **数据显示模块**：采用LCD1602字符型液晶显示屏，用于实时显示当前测量的温度值、设定的温度上下限阈值以及报警状态信息。
4. **参数设置模块**：通过四个独立按键实现，用户可以通过按键调整温度报警的上限和下限阈值。
5. **数据存储模块**：采用AT24C02 EEPROM存储芯片，通过I2C总线与单片机通信，用于存储用户设定的温度阈值，以防止系统掉电后数据丢失。
6. **报警模块**：当温度超出设定的阈值时，通过LCD显示报警信息，并驱动蜂鸣器发出声音报警。
7. **电源模块**：为整个系统提供稳定的+5V工作电压。

系统工作流程为：系统上电初始化后，单片机首先从AT24C02中读取之前保存的温度阈值（若首次使用或数据无效则使用默认值）。然后，单片机通过DS18B20采集当前温度，并将温度值和设定的阈值显示在LCD1602上。系统会周期性地检测按键输入，用户可以通过按键修改温度上下限。同时，系统会将当前温度与设定的阈值进行比较，若温度超出范围，则启动报警机制。



**图1 系统模块关系框图**

## 2 系统硬件设计

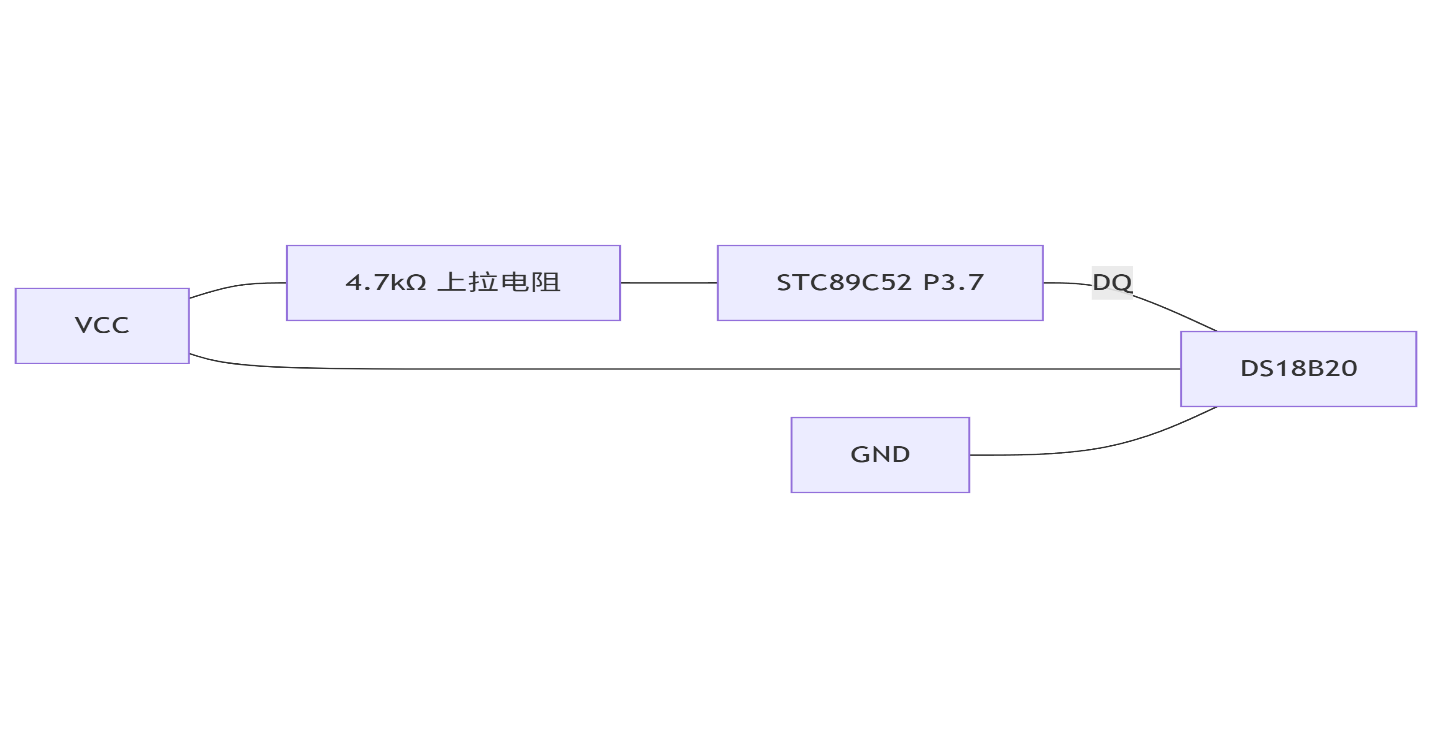
系统硬件电路主要基于普中科技的STC89C52开发板进行搭建和扩展。各主要模块的硬件接口设计如下。

### 2.1 微控制器模块

系统核心控制器选用STC89C52单片机。该单片机是一款高性能、低功耗的CMOS 8位微控制器，拥有8K字节的系统可编程Flash存储器、512字节RAM、32个可编程I/O口线、三个16位定时器/计数器、一个全双工串行通信口等资源[3]。其丰富的片上资源足以满足本系统的设计需求。本系统采用开发板上集成的最小系统，包括晶振电路（通常为11.0592MHz或12MHz，便于串口通信波特率设置和定时）和复位电路。

### 2.2 DS18B20温度传感器模块

DS18B20是一款单总线数字温度传感器，其测温范围为-55℃到+125℃，在-10℃到+85℃范围内的精度为±0.5℃。它通过单根数据线（DQ）即可与微控制器进行双向通信[4]。在本系统中，DS18B20的DQ引脚连接到STC89C52单片机的P3.7口。根据单总线协议，DQ引脚需要一个上拉电阻（通常为4.7kΩ）连接到VCC。



**图2 DS18B20连接示意图**

### 2.3 LCD1602显示模块

LCD1602是一种常用的字符型液晶显示模块，可以显示2行，每行16个字符[5]。它有8位数据总线（D0-D7）和3条控制线RS（寄存器选择）、RW（读/写选择）、EN（使能）。在本系统中，LCD1602的数据端口P0（D0-D7）直接连接到STC89C52的P0口，控制引脚RS、RW、EN分别连接到单片机的P2.6、P2.5、P2.7口。P0口作为数据输出时，需要外接上拉电阻排（开发板通常已集成）。LCD的对比度调节引脚VO通常通过一个可调电阻接地。

### 2.4 AT24C02 EEPROM模块

AT24C02是一款2K位的串行EEPROM芯片，采用I2C总线接口。它用于存储系统设定的温度上下限阈值，以实现掉电保存功能。其SCL（串行时钟）和SDA（串行数据）引脚分别连接到STC89C52单片机的P2.1和P2.0口。根据I2C总线协议，SCL和SDA引脚也需要上拉电阻连接到VCC。

### 2.5 按键模块

系统设计了4个独立按键用于参数设置，分别连接到单片机的P3.0、P3.1、P3.2、P3.3口。按键另一端接地，单片机I/O口内部或外部上拉。当按键按下时，对应I/O口检测到低电平。这四个按键分别用于：温度上限增加、温度上限减少、温度下限增加、温度下限减少。

### 2.6 报警模块（以LCD显示为参考）

报警模块由LCD显示和蜂鸣器组成。视觉报警通过在LCD1602上显示"OV:H"（超上限）或"OV:L"（超下限）实现。听觉报警通过驱动一个有源蜂鸣器实现。根据开发板原理图，蜂鸣器（Y1）通过一个NPN三极管（如S8050，Q1）由单片机的P1.5口控制。当P1.5输出高电平时，三极管导通，蜂鸣器鸣叫。

### 2.7 复位与晶振电路

系统采用标准的单片机复位电路，通常为上电复位及按键手动复位。晶振电路为单片机提供工作时钟，本设计中采用开发板上常见的11.0592MHz晶振。

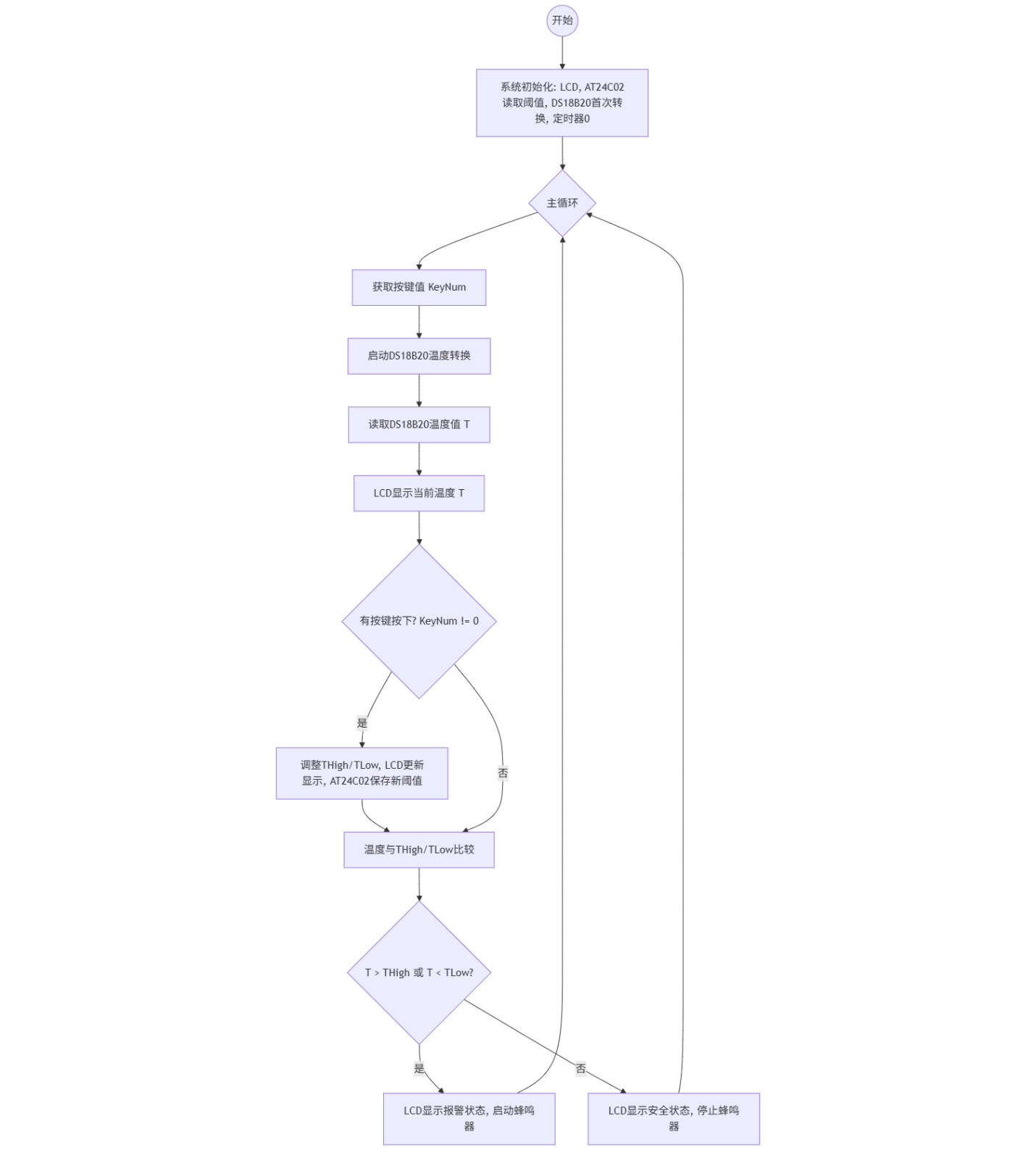
## 3 系统软件设计

系统软件采用C语言编写，在Keil C51集成开发环境下进行编译和调试。软件主要包括主程序模块、DS18B20驱动模块、LCD1602驱动模块、AT24C02驱动模块、按键处理模块和定时器中断服务程序模块。

### 3.1 主程序流程 (main.c)

主程序 main() 的流程如图3所示。

1. **系统初始化**：
   * 调用 LCD\_Init() 初始化LCD1602模块。
   * 调用 AT24C02\_ReadByte() 从EEPROM读取之前保存的温度上限 THigh (地址0) 和下限 TLow (地址1)。
   * 对读取的阈值进行有效性检查（如上限是否大于125℃，下限是否小于-55℃，上限是否小于等于下限），若无效则恢复为默认值（例如 THigh = 33℃, TLow = 7℃）。
   * 调用 DS18B20\_ConvertT() 启动一次温度转换。
   * 延时一段时间（如1秒）等待温度转换完成。
   * 在LCD上显示固定的标签字符串，如"T:"、"TH:"、"TL:"，并显示读取或默认的阈值。
   * 调用 Timer0\_Init() 初始化定时器0，用于周期性扫描按键。
2. **主循环**：
   * 调用 Key() 获取按键值 KeyNum。
   * 调用 DS18B20\_ConvertT() 启动下一次温度转换。
   * 调用 DS18B20\_ReadT() 读取上一次转换完成的温度值 T。
   * 处理温度值的正负号，并将其整数部分和小数部分（保留两位）分别格式化后显示在LCD的第一行。例如，+027.18。
   * **按键处理与阈值设置**：
     + 如果 KeyNum 不为0，表示有按键按下。
     + 根据 KeyNum 的值（1、2、3、4）分别对 THigh 或 TLow 进行加减操作。
     + 进行阈值范围检查（例如，THigh 不超过125℃，TLow 不低于-55℃，THigh 必须大于 TLow）。
     + 更新LCD上显示的阈值。
     + 调用 AT24C02\_WriteByte() 将修改后的 THigh 和 TLow 写回EEPROM，并加入适当延时确保写入完成。
   * **温度比较与报警**：
     + 如果当前温度 T > THigh，则在LCD上显示 "OV:H"，并驱动蜂鸣器（通过P1.5控制）。
     + 如果当前温度 T < TLow，则在LCD上显示 "OV:L"，并驱动蜂鸣器。
     + 如果温度在阈值范围内，则显示 "Safe"，并停止蜂鸣器。
   * 循环执行。



**图3 主程序流程图**

### 3.2 DS18B20驱动模块 (DS18B20.c, OneWire.c)

* OneWire.c 封装了单总线通信的基本操作，包括总线初始化 (OneWire\_Init())、发送一位 (OneWire\_SendBit())、接收一位 (OneWire\_ReceiveBit())、发送字节 (OneWire\_SentByte()) 和接收字节 (OneWire\_ReceiveByte())。这些函数严格按照DS18B20的时序要求进行操作。
* DS18B20.c 基于 OneWire.c 提供的函数实现对DS18B20的具体操作：
  + DS18B20\_ConvertT(): 发送跳过ROM指令(0xCC)和启动温度转换指令(0x44)，命令DS18B20开始进行温度转换。
  + DS18B20\_ReadT(): 发送跳过ROM指令(0xCC)和读暂存器指令(0xBE)，然后从DS18B20读取9个字节的暂存器数据，提取前两个字节（温度的低字节TLSB和高字节TMSB），将其合成为16位带符号整数，最后除以16.0得到浮点型的摄氏温度值。

### 3.3 LCD1602驱动模块 (LCD1602.c)

该模块包含LCD1602的初始化及显示功能函数：

* LCD\_Init(): 初始化LCD1602，设置其工作模式为8位数据接口、两行显示、5x7点阵字符，开显示，关光标，清屏等。
* LCD\_WriteCommand(): 向LCD写入命令。
* LCD\_WriteData(): 向LCD写入数据（字符）。
* LCD\_SetCursor(): 设置光标在LCD上的显示位置。
* LCD\_ShowChar(): 在指定位置显示单个字符。
* LCD\_ShowString(): 在指定位置显示字符串。
* LCD\_ShowNum(): 在指定位置显示无符号十进制数字。
* LCD\_ShowSignedNum(): 在指定位置显示带符号的十进制数字。
* LCD\_ShowHexNum(): 在指定位置显示十六进制数字。
* LCD\_ShowBinNum(): 在指定位置显示二进制数字。
* LCD\_Delay(): LCD操作所需的延时函数。
* LCD\_Pow(): 计算幂次方，用于数字显示时的位值计算。

### 3.4 AT24C02驱动模块 (AT24C02.c, I2C.c)

* I2C.c 实现了I2C总线通信的基本操作，包括起始信号 (I2C\_Start())、停止信号 (I2C\_Stop())、发送字节 (I2C\_SendByte())、接收字节 (I2C\_ReceiveByte())、发送应答 (I2C\_SendAck()) 和接收应答 (I2C\_ReceiveAck())。
* AT24C02.c 调用I2C总线函数与AT24C02进行通信：
  + AT24C02\_WriteByte(unsigned char WordAddress, unsigned char Data): 向AT24C02的指定地址 WordAddress 写入一个字节数据 Data。
  + AT24C02\_ReadByte(unsigned char WordAddress): 从AT24C02的指定地址 WordAddress 读取一个字节数据。

### 3.5 按键处理模块 (Key.c)

* Key\_GetState(): 直接读取P3口的0-3位，返回当前按下的键号（1-4），无按键则返回0。此函数不进行消抖。
* Key\_Loop(): 这是一个在定时器中断中调用的按键扫描函数。它通过比较当前按键状态 NowState 和上一次的按键状态 LastState，实现松手检测。当检测到按键从按下到松开的瞬间，才将对应的键值赋给全局变量 Key\_KeyNumber。这种方式可以有效避免按键抖动和长按重复触发的问题。
* Key(): 主程序通过调用此函数获取 Key\_KeyNumber 的值，并在获取后将其清零，避免重复处理。

### 3.6 定时器中断服务程序 (Timer0.c, main.c 中断部分)

* Timer0\_Init(): 初始化定时器T0为模式1（16位定时模式），设置定时初值使其大约每1ms（或根据实际晶振和配置调整）产生一次中断，并开启T0中断和总中断。
* Timer0\_Routine() interrupt 1: 定时器T0的中断服务程序。在此程序中：
  + 重装定时器初值以维持定时周期。
  + 通过一个静态计数器 T0Count 进行分频，例如每20次中断（约20ms）调用一次 Key\_Loop() 函数，进行按键状态的扫描和更新。

## 4 系统测试与结果分析

为验证系统的功能和性能，对搭建好的硬件平台进行了通电测试。测试中主要观察LCD的显示内容、按键调节阈值的响应以及超限报警的准确性。

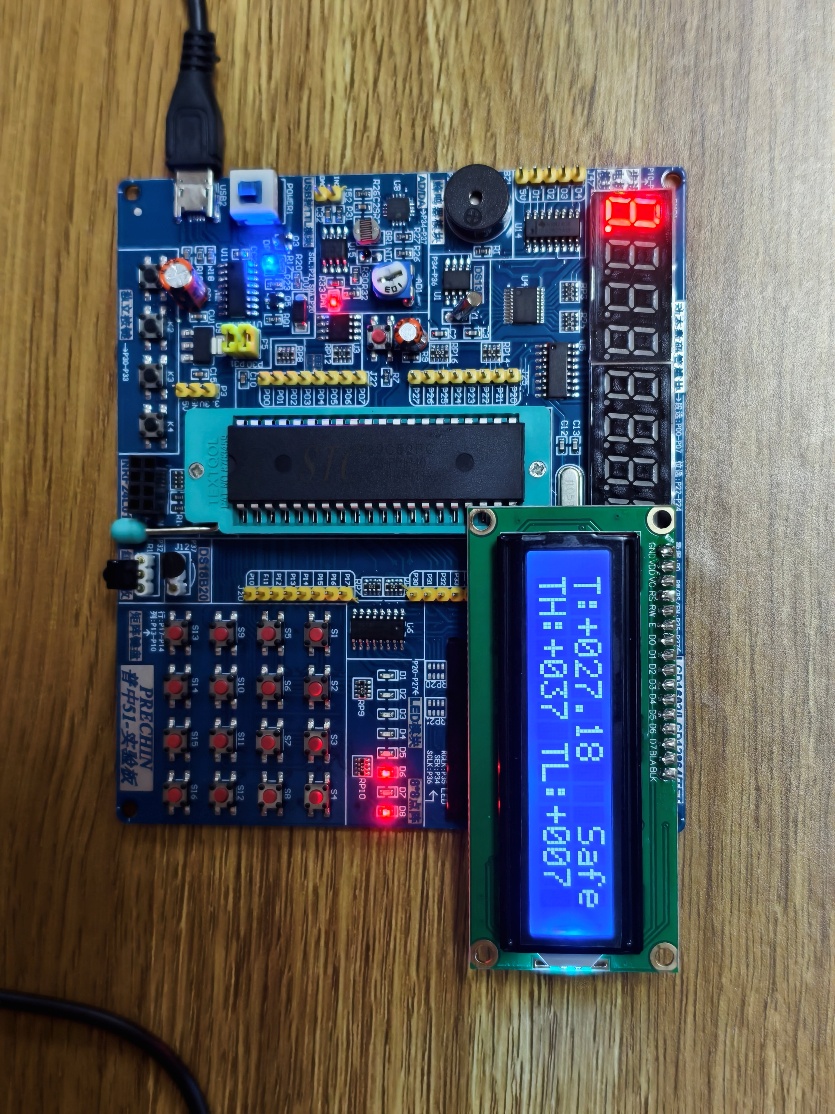
**测试环境：**

* 硬件平台：基于STC89C52单片机的开发板，连接DS18B20温度传感器、LCD1602显示屏、四个独立按键、AT24C02 EEPROM及蜂鸣器。
* 电源：+5V直流电源。

**测试步骤与现象：**

1. **上电初始化测试**：系统上电后，LCD1602正常显示。若为首次运行或EEPROM数据无效，则温度阈值显示为默认值（如TH:+033, TL:+007）。若之前已设置过阈值并掉电，则LCD应显示上次保存的阈值。当前温度正常显示。
2. **温度显示测试**：用手触摸DS18B20传感器或使用热源/冷源改变其周围温度，观察LCD上显示的温度值是否随实际温度变化而相应变化，并与标准温度计进行对比，验证其准确性。DS18B20的精度较高，显示值应能较好地反映实际温度。
3. **阈值设置与存储测试**：
   * 通过K1键（P3.1）增加温度上限TH，观察LCD显示是否相应增加，并测试其上限（125℃）。
   * 通过K2键（P3.0）减少温度上限TH，观察LCD显示是否相应减少，并测试其下限（不能低于TLow）。
   * 通过K3键（P3.2）增加温度下限TL，观察LCD显示是否相应增加，并测试其上限（不能高于THigh）。
   * 通过K4键（P3.3）减少温度下限TL，观察LCD显示是否相应减少，并测试其下限（-55℃）。
   * 设置一组阈值后，断开系统电源再重新上电，观察LCD显示的阈值是否为断电前设置的值，验证AT24C02的非易失性存储功能。
4. **报警功能测试**：
   * **正常状态**：设置合适的温度阈值，使当前环境温度处于上下限之间。此时LCD应显示 "Safe"，蜂鸣器不响。如图4(a)所示，当前温度+27.18℃，阈值为TH:+037℃, TL:+007℃，系统显示 "Safe"。
   * **超上限报警**：通过按键将温度上限TH调低至低于当前温度（例如，当前温度为+27.12℃，设置TH为+022℃，TL为+007℃）。此时，LCD应显示 "OV:H"，蜂鸣器鸣叫。如图4(b)所示。
   * **超下限报警**：通过按键将温度下限TL调高至高于当前温度（例如，当前温度为+26.93℃，设置TH为+038℃，TL为+030℃）。此时，LCD应显示 "OV:L"，蜂鸣器鸣叫。如图4(c)所示。

**实验结果图示：**

****

**图4 系统测试结果**

结果分析：

从上述测试过程和结果图可以看出，系统能够准确测量并显示当前温度，用户可以通过按键方便地设置温度的上下限阈值，并且设置的阈值能够通过AT24C02 EEPROM进行掉电保存。当实际温度超出设定的阈值范围时，系统能够及时在LCD上显示相应的报警信息（"OV:H" 或 "OV:L"），并通过蜂鸣器发出声音报警。按键响应灵敏，LCD显示清晰，报警逻辑正确，系统整体运行稳定可靠，达到了预期的设计目标。

## 5 结论

本文成功设计并实现了一款基于STC89C52单片机的多功能温度监控报警系统。该系统集温度采集、实时显示、阈值设定、数据存储及超限报警功能于一体。通过对DS18B20数字温度传感器、LCD1602显示模块、AT24C02 EEPROM存储器以及按键模块的有效整合与编程控制，实现了对环境温度的精确监控和智能化管理。实验测试结果表明，该系统功能完善，运行稳定，报警及时准确，人机交互友好，具有较高的实用性和应用价值。

## 参考文献

[1] 孙志远, 尹张轩, 王印申, 等. 基于STC89C52单片机的多功能电子日历设计[J]. 电子技术与软件工程, 2024(6): 48-50.

[2] 王晓晨. 单片机原理及应用[M]. 北京: 电子工业出版社, 2018.

[3] STC. STC89Cxx系列单片机器件手册[EB/OL].

[4] Maxim Integrated (原Dallas Semiconductor). DS18B20 Programmable Resolution 1-Wire Digital Thermometer [EB/OL].

[5] HD44780U (LCD-II). Dot Matrix Liquid Crystal Display Controller/Driver [EB/OL].

[6] Atmel Corporation. AT24C01A/02/04/08A/16A, 2-Wire Serial EEPROM [EB/OL].

## 6 数据提交

本实验所有数据及代码均上传至Github可供下载参考，链接如下：