

# CC2430 芯片和单片机的暖气控制系统设计与实现

Design and Implementation of the Heating Control System  
Based on CC2430 Chip and MCU

陈汉臻 王爱美 罗进文 王兴梅

(兰州交通大学电子信息工程学院,甘肃 兰州 730070)

**摘要:** 针对现有公共建筑采暖系统存在能源浪费以及不能有效节约能源的问题,设计了基于 STC10L08XE 单片机和 CC2430 芯片的暖气控制系统。系统通过采集室内温度,自动调节暖气流量,以实现节能的目的;同时利用 CC2430 芯片完成温度控制主板与手持终端的无线通信。介绍了系统各个模块的软硬件设计,并给出了部分原理图和程序设计流程图。试用结果表明,系统运行稳定,节能效果明显。

**关键词:** ZigBee CC2430 芯片 温度采集 单片机 串口通信 温度传感器

**中图分类号:** TP212+.9 **文献标志码:** A

**Abstract:** In order to solve the problems of energy waste and saving energy ineffectively in heating system of existing public buildings, a heating control system based on single chip computer STC10L08XE and CC2430 chip is designed. By collecting the indoor temperature, and automatically regulating the heating flow, the energy is saved. In the system, CC2430 chip is used to complete the wireless communication between the main temperature control board and the handheld terminal. The designs of the hardware and software of various modules of the system are introduced, and some of the principles and program flowcharts are given. Trial application results show that the system runs stably and outstanding energy-saving effect is obtained.

**Keywords:** ZigBee CC2430 chip Temperature collection MCU Serial communication Temperature sensor

## 0 引言

目前,我国北方城市的供暖普遍采用 24 h 集中供暖方式,但集中供暖存在能源浪费大以及供暖收费困难等问题。随着我国供暖体制改革的深入,居民住宅和公共建筑采暖将逐步转变为按取暖热量收费。在住宅和公共建筑空置的情况下,减少供暖热量有助于节约能源。

本文采用 STC10L08XE 单片机、CC2430 芯片和 DS18B20 温度传感器为核心,设计了暖气控制系统。室内温度传感器采集到的室内温度,通过 RS-485 串口传送到温度控制主板,然后由相应的程序进行处理。通过改变暖气控制阀的开关大小来调节热水流量,以达到控制室内温度的目的。ZigBee 模块主要进行温度控制主板和手持终端的通信,用户在手持终端就可以查看室内温度、当前控制阀档位以及时间等信息,也可以完成各种信息的设定工作。

## 1 工作原理与总体结构设计

系统由温度传感器采集室内温度。温度控制主板按顺序对温度传感器数据进行读取,并将信息显示在液晶模块上。系统能够随着室内温度的变化改变暖气控制阀开关程度,以便调节室内温度。ZigBee 模块分为板载和手持终端,用户通过手持终端可以修改和查看温度控制主板信息。

系统基本结构如图 1 所示。

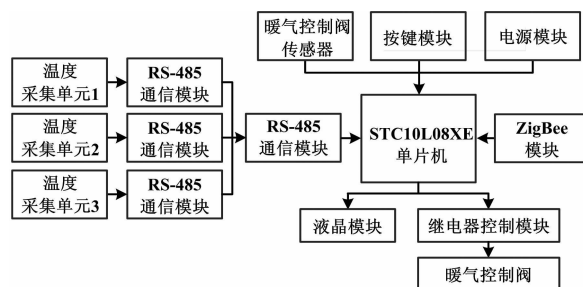


图 1 系统结构原理图

Fig. 1 Principle of the system structure

系统主要分为温度采集模块、RS-485 通信模块、液晶模块、按键模块、电源模块、STC10L08XE 单片机和继电器模块。同时,本系统具有日历功能,可实现对

甘肃省自然科学基金资助项目(编号:1010RJZA065)。

修改稿收到日期:2011-08-15。

第一作者陈汉臻(1985-),男,现为兰州交通大学通信与信息系统专业在读硕士研究生;主要从事无线通信方面的研究。

不同日期或特定日期的暖气控制。

温度采集模块一般放置在室内不同的地方,由于通信距离较远,可采用 RS-485 通信方式将温度信息传输到温度控制主板。

根据 RS-485 总线通信方式,温度控制板采取主动循环方式读取温度信息;然后根据程序设置对温度数据进行处理;最后根据控制暖气控制阀的开关程度,完成温度的调节功能。

液晶模块是实现人机交互的界面,温度信息和目前的暖气控制阀的开关程度都在液晶模块上显示。按键模块可以调整系统的时间,并根据使用环境确定不同时间的暖气控制阀开关程度。继电器控制模块实现了由 3.3 V 控制 12 V 的 PCB 继电器,间接控制 220 V 电力继电器的功能。

## 2 系统硬件设计

### 2.1 元器件选择

STC10L08XE 单片机是宏晶科技生产的新一代增强型 8051 单片机,它具有高速、低功耗、宽电压的特点,其工作频率为 0~35 MHz,片内有 8 kB 的 Flash 程序存储空间、256 B 的 RAM 数据存储空间和 2 kB 的 EEPROM 存储空间。

温度采集模块采用了 DALLAS 公司的单总线可组网 DS18B20 数字温度传感器。其采用独特的单总线接口方式,即只需要使用一条总线就可以实现与 MCU 的双向通信。

DS18B20 数字温度传感器支持多点组网功能,测温范围为 -55~+125℃。其中,-10~+85℃ 范围内测量结果最小可以精确到 0.5℃,测量结果可以采用 9~12 位数字方式串行输出。

ZigBee 芯片选用 TI 公司的 CC2430 芯片<sup>[1]</sup>。CC2430 沿用了以往的 CC2420 芯片架构,在单个芯片上整合了 ZigBee 射频前端、内存和微控制器。CC2430 具有 128 kB 可编程闪存、8 kB 的 RAM,它由模拟数字转换器(ADC)、定时器、AES128 协同器、看门狗定时器以及 32 kHz 晶振的休眠模式定时器等组成。CC2430 采用 0.18 μm 的 CMOS 工艺生产,在接收和发射模式下,电流损耗分别为 27 mA 和 25 mA。其休眠模式和主动模式能相互转换,适合应用在对电池寿命要求高的场合。

### 2.2 系统主要硬件设计

#### 2.2.1 温度控制主控板设计

温度控制主板主要用来完成对温度传感器传送的数据进行处理,以调整暖气控制阀的开关程度,并通过液晶

和按键实现人机交互。温度控制主板采用 STC10L08XE 单片机。该芯片采用 3.3 V 供电,共有 48 个引脚。

STC10L08XE 连接原理图如图 2 所示。

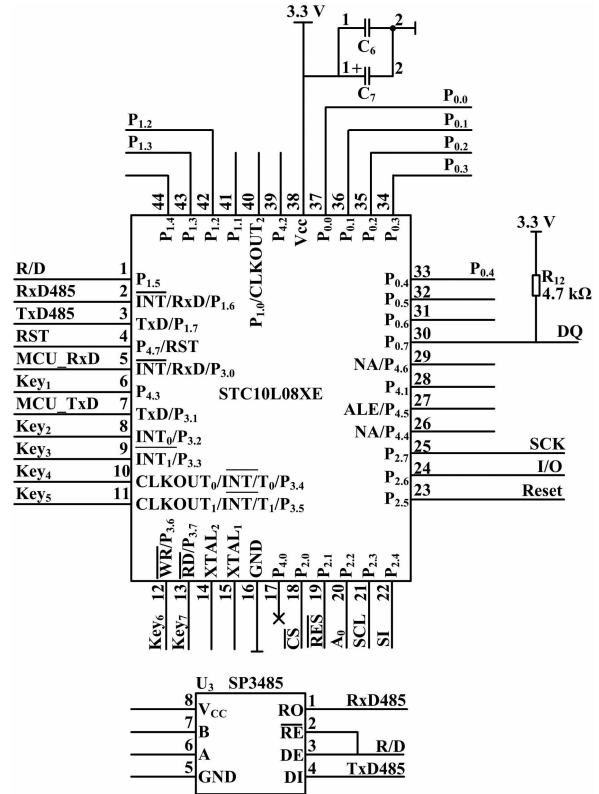


图 2 STC10L08XE 连接原理图

Fig.2 Principle of STC10L08XE connection

本设计中,P<sub>0.0</sub>~P<sub>0.4</sub>引脚用于控制霍尔传感器工作。由于暖气控制阀的霍尔传感器采用 12 V 电源,而单片机采用 3.3 V 电源,因此,为避免因霍尔传感器电压变化影响单片机工作,在单片机和继电器之间采用光耦连接方式。P<sub>1.2</sub>和 P<sub>1.3</sub>引脚用于控制电机工作,同时也采用光耦的方式进行连接。

温度传感器距离主板控制板较远,所以采用 RS-485 进行通信。SP3485 引脚的主要功能包括 A 驱动器输出/接收器输入(同向)和 B 驱动器输出/接收器输入(反相)。主板与温度采集之间通过 RS-485 总线进行通信,即将所有的温度采集板挂载在一条总线上。

#### 2.2.2 ZigBee 模块设计

ZigBee 控制模块是节能系统的重要组成部分,它具有两个功能:一是负责组建 ZigBee 网络;二是通过手持 ZigBee 终端控制主板,从而控制温度调整和查看相应的数据<sup>[2]</sup>。

ZigBee 模块设计原理图如图 3 所示。

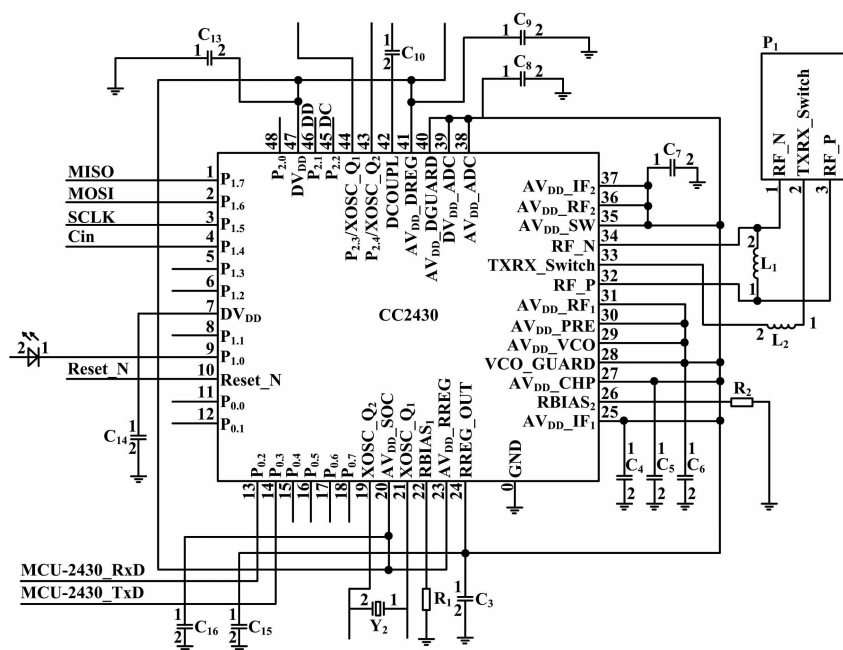


图3 ZigBee 模块设计原理图

Fig. 3 Design principle of the ZigBee module

外围电源向 CC2430 芯片提供 3.3 V 工作电压,以保证芯片 I/O 口、内部模拟电路和电压调节器正常工作。CC2430 芯片 24 (RREG\_OUT) 引脚输出 1.8 V 稳定电压。该电压供应给芯片的 25、27 ~ 31 及 35 ~ 40 引脚。这些端口输入电压将为接收滤波器、模拟测试模块、环状滤波器的第一部分电路和充电泵、接收和发射混频器等电路提供 1.8 V 工作电压。天线部分采用了 PCB 天线,电感  $L_1$  和  $L_2$  构成不相关的非平衡变压器。电阻  $R_1$  和  $R_2$  为偏置电阻,电阻  $R_1$  为 32 MHz 晶振提供一个合适的工作电流,电阻  $R_2$  为 RF 射频端提供合适的工作电流。电路中采用了两个晶振,其中一个为 32 MHz 的石英晶体振荡器,另一个是 32.768 kHz 晶振<sup>[3]</sup>。

CC2430 的  $P_{2.1}$ 、 $P_{2.2}$  引脚为调试接口,同芯片的复位引脚相配合可实现对芯片的编程调试。 $P_1$  端口的高四位被配置为 UART 工作模式。CC2430 通过  $P_1$  口的 UART 可与 PC 机进行通信,以方便用户查看与调试。 $P_0$  端口也配置有 UART 功能,其中, $P_{0.2}$  引脚作为 RX 引脚, $P_{0.3}$  引脚作为 TX 引脚。通过  $P_0$  端口的串口功能,板载 ZigBee 模块与控制板上的 MCU 进行通信,从而实现信息交互。 $P_{1.0}$  连接一个 LED 发光二极管,可用作工作指示灯,其余 I/O 端口均没有被使用<sup>[4]</sup>。

### 2.2.3 电机驱动电路设计

受控电机位于井下通道中,为暖气控制阀提供动力。电机一般采用 220 V 交流电工作,电源回路要受到

温度控制主板的控制。当温度控制主板电压为 3.3 V 时,不足以控制电力继电器工作,为此,在温度控制主板上安装 PCB 继电器,并连接到电力继电器。

STC10L08XE 单片机端口驱动最大为 20 mA,尚不足以使 PCB 继电器可靠吸合。而 TLP621 光耦隔离器发射极的最大输出电流可达 150 mA,能够满足 PCB 继电器吸合的要求。同时,单片机工作电压为 3.3 V,PCB 继电器工作电压为 12 V,所以驱动电路设计采用光耦隔离器 TLP621 将 MCU 和 PCB 继电器隔离,以防止 12 V 工作器件对 3.3 V 工作器件产生影响。

当输入端  $P_{1.3}$  或者  $P_{1.4}$  引脚为低电平时,光耦内的发光二极管导通,此时光耦内的三极管也导通工作,PCB 继电器吸合,电机启动工作。当输入端  $P_{1.3}$  或者  $P_{1.4}$  引脚为高电平时,光耦内的发光二极管截止,此时光耦内的三极管呈截止状态,PCB 继电器则保持常开状态,电机停止工作。

### 2.2.4 液晶显示模块设计

系统采用 LM6059BCW 液晶显示模块。该液晶基于超低压设计,采用 ST7565P 液晶显示控制芯片,支持标准的八位并口或 SPI 串口工作<sup>[5]</sup>。

LM6059BCW 液晶显示模块具有并口和串口两种工作模式。本设计采用了串行口工作方式对液晶模块进行操作。在串行工作模式下,接口电路的设计需要用到五条串口线,它们分别是片选信号线、复位信号线、液晶模块数据命令选择线、SCL 串行时钟线和 SI

数据输入线,分别连接 MCU 的  $P_{2.0} \sim P_{2.4}$  引脚。MCU 通过模拟串行工作的方式对液晶模块进行写入操作,实现对液晶模块的命令控制和数据显示。

### 3 系统软件设计

系统软件部分主要分为 RS-485 通信模块和板载 ZigBee 节点串口通信模块软件设计两部分。温度控制主板总体软件流程图如图 4 所示。

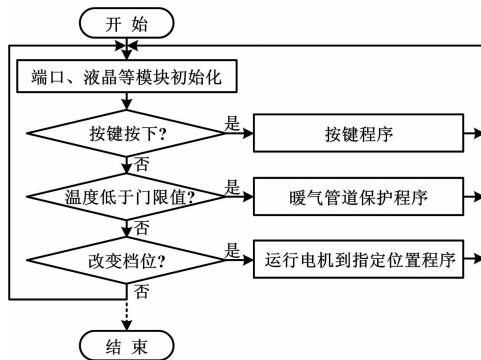


图 4 温度控制板软件流程图

Fig. 4 Software flowchart of the temperature control board

#### 3.1 RS-485 通信模块程序

每一个温度控制主板都有三个温度采集点,所有的温度采集点均通过一条总线和温度控制主板进行通信。它们之间的通信方式是采用温度控制主板分别读取各温度采集点的温度。首先,温度控制主板发送需要读取温度采集点编号,温度采集点响应后将准备好的数据发送到总线上;然后温度控制主板接收数据,并通知温度采集点通信结束,释放总线;最后温度采集点需要给主板发送一个确认信息。这样就完成了一个通信过程。

#### 3.2 板载 ZigBee 节点串口通信

CC2430 有 USART<sub>0</sub> 和 USART<sub>1</sub> 两个串行通信接口。这两个串口既可以在 UART(异步通信)模式下工作,也可以在 SPI(同步通信)方式下工作,模式的选择由串口控制/状态寄存器的 UxCSR.MODE 决定。UART 模式可以选择两线连接(RXD 和 TXD)或四线连接(RXD、TXD、RTS 和 CTS),其中,RTS 和 CTS 用于流量控制<sup>[6]</sup>。

UART 模式具有以下操作特点:八位或者九位数据;奇校验、偶校验或者无奇偶校验;配置起始位和停止位电平;配置 LSB 或者 MSB 首先传送;独立收发中断和 DMA 触发;奇偶校验和帧校验出错状态<sup>[7]</sup>。

UART 模式提供全双工传送,接收器中的位同步不影响发送功能。传送一个 UART 字节包含一个起始位、八个数据位、一个作为可选项的第九位数据或者奇

偶校验位再加上一个(或两个)停止位。UART 操作由 UART 控制、状态寄存器 UxCSR 以及 UART 控制寄存器控制。当 UxCSR.MODE 被设置为 1 时,就选择 UART 模式<sup>[8]</sup>。

板载节点串口程序如下。

##### ① 串口初始化

初始化是对与串口相关的寄存器进行配置,主要包括模式的选择、串口使用端口的选择和端口传输方向的设置。

串口初始化代码如下。

```

void Uart0Init( void)
{
    PERCFG &= 0xfe;           //设置串口使用端口
    POSEL |= 0x0c;
    PODIR |= 0x08;           //设置传输方向
    UOCSR |= 0x80;           //模式选择
    UOGCR |= 0x08;           //BAUD_E = 8
    UOBAUD |= 59; //BAUD_M = 59; 波特率设置为 9 600
    UTX0IF = 1;
    IEN0 |= 0x84;           //开总中断和串口 0 接收中断
    UOCSR |= 0x40;           //使能接收
}
  
```

##### ② UART 发送数据

当 USART 收/发数据缓冲器 UxBUF 写入数据时,该字节被发送到输出引脚 TXD<sub>x</sub>。当字节传送开始时,UxCSR.ACTIVE 设置为 1;而当字节传送接收时,UxCSR.ACTIVE 清零。当传送结束时,TX\_BYTE 设置为 1。当 USART 收/发数据缓冲寄存器就绪,准备接收新的发送数据时,就产生了一个中断请求。该中断在传送开始之后立刻发生,因此,当字节正在发送时,新的字节能够装入数据缓冲器。串口发送程序代码如下。

```

int8 uarttx( int8 c)
{
    UOCSR &= 0xbf;           //禁止接收,使能发送
    while( ! UTX0IF);
    UTX0IF = 0;
    return( U0DBUF = c);
    UOCSR |= 0x40;           //重新使能接收
}
  
```

##### ③ UART 接收数据

当将 1 写入到 UxCSR.RE 位时,在 UART 上的数据接收就开始了;然后 USART 会在输入引脚 RXD<sub>x</sub> 中寻找有效起始位,并且设置 UxCSR.ACTIVE 位为 1。当

(下转第 24 页)