泰 山 学 院

本 科 毕 业

基于 STC89C52 单片机的家用红外遥控系统

所　　在　　学　　院　　信息科学技术学院

专　　业　　名　　称　　自动化

申请学士学位所属学科　　工　学

完　　成　　日　　期　　20XX 年 5 月 1 日

摘 要

随着科技的发展，特别是微电子技术的发展，单片机的应用已经深入到生活的各个领域。本文以 STCC89C52 为控制芯片，DS1302 为时钟芯片，DS18B20 为温度传感器，HS0038 红外一体接收头作为红外接收器，以此接收并处理遥控器按键信号并用来矫正时钟时间。同时，本设计还加入了定时和设定温度上下限的功能，当达到预设时间时执行开关灯命令，当温度低于温度下限或高于温度上限时执行开空调命令，开关器件为两个大功率电磁继电器。本设计中所有外设数据显示于 LCD12864 液晶，形成一个良好的用户界面。硬件电路图和 PCB 用 Protel99SE 绘制，软件开发环境为 Keil V3，设计实物为万用板焊接电路，经调试各项功能均已正常实现。

关键词：红外遥控，单片机，DS1302 ，DS18B20 ，LCD12864

ABSTRACT

With the development of science and technology, especially the development of microelectronics technology, SCM applications have been deeply into all areas of li Taking STCC89C52 as control chip, for DS1302 clock chip, DS18B20 as temperature collector, HS0038 infrared receiving head as integrated infrared receiver, signal receiving and processing the remote control buttons, in order to correct the clock t At the same time, the project also joined the function of timing, and setting tempera of upper and lower, executive orders open to turn off the lights when preset time is reached, when the temperature lower than the temperature is lower or higher than th temperature limit and execution open air conditioning, for electromagnetic relay switch device. The whole project all the data shown in LCD12864 LCD, forming a good user interface. Made in Protel99SE, the hardware circuit diagram and PCB software development environment for KeilV3, project physical for universal plate welding circuit, through debugging functions are normal.

Key words: infrared remote control, microcontroller, DS1302, DS18B20, LCD12864

目 录

1 绪 论

1.1 家用电器的发展及智能家电的前景

电器在国民日常生活和国防领域中占有举足轻重的位置，有着不可或缺的作用，其中，家用电器与人们的生活的关系最为密切。家用电器问世已有百年历史，美国人爱迪生发明了白炽灯，从此开创了家庭用电时代。19 世纪末电磁波被证实存在，奠定了电子学诞生的基础。之后，二极电子管和三级电子管的发明使人们开始了真正的电子应用。50 年代电子工业迅速发展，晶体管尤其是集成电路的发明，使人们进入微电子时代同时家用电器的应用也提高到新的水平。

随着科技的发展家用电器也越来越智能化，家用电器的发展趋势主要有高性能高可靠性，电路集成化功能多样化，随着现代电子技术的发展家用电器与多学科交叉融合向着网络化与智能化发展。总体看来现代家用电器具有更强大的功能，而实现强大功能的保障是具有良好的微处理器，一个智能家电可以实现多个传统家电的功能。多功能是智能家电主要特点，同时，通信功能也是智能家电的发展方向。另外，新型家用电器在环保，审美以及安全性上也对人们提出了新要求。

1.2 电子技术的发展对智能家电的推动

20 世纪以来，电子技术发展迅速，特别是近几年来，现代电子产品以及各种消费电子产品已经渗透到了生活各方面，电子技术的发展有力的推动了科技进步并且提高了人们的生活水平。以单片机为核心的多功能系统正在飞快的改变着人们的生活，特别是一些家用智能产品，不但方便了人们的生活，更加丰富了人们的生活方式。以本文中的家用红外遥控系统为例，该设计除了具有一般的日历功能外，还增加了温度检测功能，但其能成为“智能”之处并不在于此，而是他可以通过遥控按键设定时间值和温度值，程序会按照这些数值自动的控制继电器，从而达到了定时开关灯，定温开关空调的目的。

在享受越来越智能化的电子产品的时候，我们还要再次感谢电子技术的发展，集成化的功能模块已使编程和硬件设计变得非常方便。以温度采集模块为例，经典的温度采集系统由温度敏感器件采集到模拟量、经过 A/D 转换和单片机电路组成。由于温度传感器输出模拟量那么信号需要经过 A/D 转换，转换为数字量后才

能与单片机等数字接口进行通信，从而使电路变得比较复杂，提高了成本。而 DS18B20 集成了模数转换功能，通过封装只留出一个数据接口，这样就可以直接输出数字量，从而与单片机连接变得非常简单，提高了灵活度，降低了成本增加了应用价值。另外 HS0038 红外一体接受头，更是能把红外信号直接转变为电压信号，省去了大量的外围电路，方便了程序编写。

1.3 家用红外遥控系统设计思路

本设计主要应用于家庭，操作方便功能丰富，能很好的为人们的生活提供方便。

家用红外遥控系统要实现的功能主要有，显示时间值，显示温度值，定时开关功能，设定温度上下限功能，另外，由于采用红外遥控器作为按键，还应具有红外接收功能。根据以上功能要求，来选择所需要的硬件，本设计中所用到的硬件主要有 STC89C52 单片，DS1302 时间芯片，DS18B20 温度传感器，HS0038 红外接收器，LCD 液晶屏，继电器。在软件设计上，各个模块单独编程，这样能够方便调试发现错误，也方便今后程序移植和升级。

2 硬件设计

2.1 STC89C52 单片机介绍

2.1.1STC89C52 引脚简介

STC89C52 管脚及实物图如图 2-1，图 2-2所示。

电源引脚(2根)

GND(20 脚)：接地线；VCC(40 脚)：接＋5V 电源

外接晶振引脚(2根)

XTAL0(18 脚)：振荡电路的输入端；XTAL1(19 脚)：振荡电路的输出端

控制引脚(4根)

ALE/PROG(30 脚)：地址锁存允许信号；EA/VPP(31 脚)：片内外程序存储器选择，低电平读取外部存储器指令，高电平读片内程序指令；RST/VPP(9 脚)：复位引脚 2 个机器周期以上的高电平单片机复位；PSEN(29 脚)：外部存储器读选通信号。

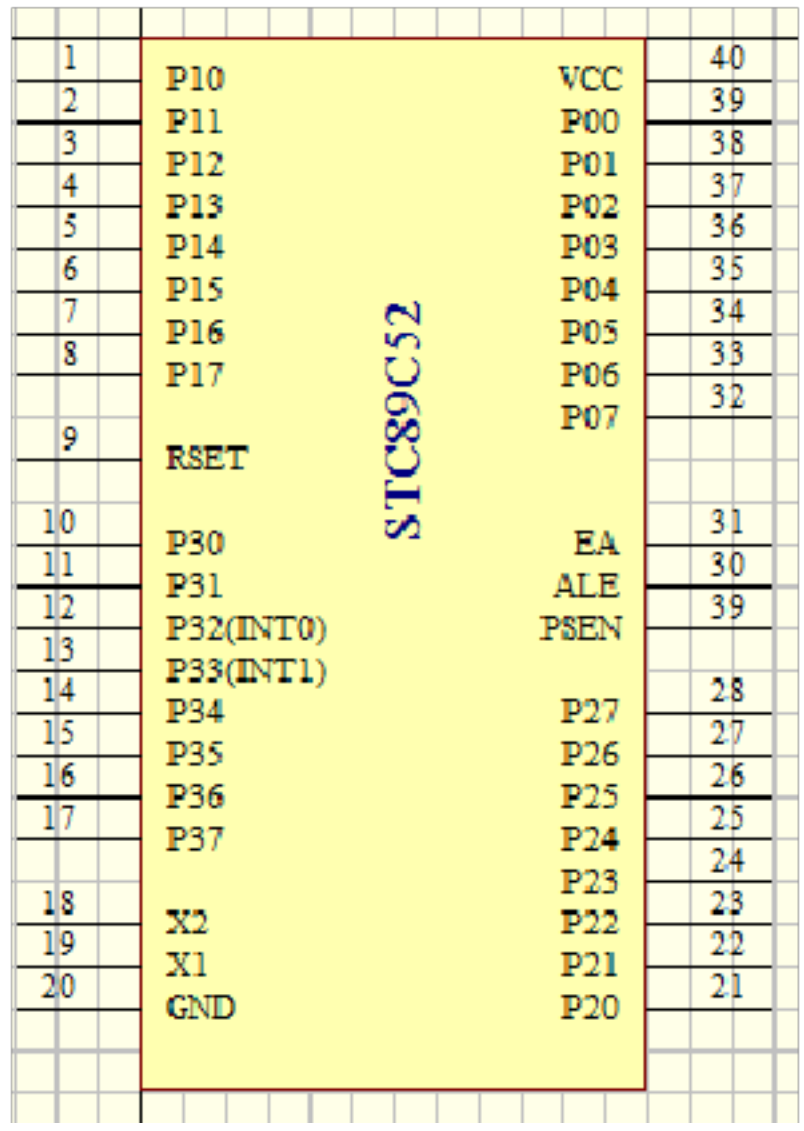


图 2-1 STC89C52 管脚图



图 2-2 STC89C52 实物图

I/O口(32根)

该单片机有 4 组（P0、P1、P2、P3）I/O口，每组 8 位，共 32 根引脚。P0 口(39脚～32 脚)是一个具有 8 位漏极开路的双向 I/O口。当做输出口使用时，每组能驱动 8 路 TTL 电平。

P1 口(1脚～8 脚)是一个 8 位双向 I/O 口且具有上拉电阻。可以用来驱动 TLL

逻辑电平。

P2 口(21脚～28 脚)是一个 8 位双向 I/O 口且具有内部上拉电阻，P2 口可以用来驱动 TTL 逻辑电平。

P3 口(10脚～17 脚)是 8 位准双向 I/O口，同时 P3 口具有复用功能，P3.0 串行输入(RXD) 、P3.1 串行输出(TXD) 、P3.2 外部中断 0(INT0)、P3.3 外中断 1(INT1)、P3.4 定时/计数器 0、P3.5 定时/计数器 1、P3.6 外部数据存储器写选通、P3.7 外部数据存储器读选通。

2.1.2STC89C52 主要功能特点

STC89C52 是宏晶科技生产的一种高性价比低功耗的 8 位单片机，片内集成 8KFlash程序存储器，STC89C52 使用经典的 MCS-51 内核，程序上完全兼容。但该单片机还是做了不少改进，缩短了指令执行周期，增加了定时器 2，增加了看门狗功能。在单芯片上，8 位的 CPU 配合 8K 的程序存储器，使得 STC89C52 能满足基本的控制需要而不需要再外扩存储器。 以下是 STC89C52 的基本特点：

(1) 相对于一个机器周期占 12 个时钟周期的 8051 增加 6 时钟/机器周期功能。

(2) 工作电压宽度在 5.5V - 3.4V

(3) 工作频率 0 - 40 MHz 最高可达到 48MHz 。

(4) 片内集成 8K 字节的程序存储器和 512B 的 RAM 。

(5) 通用 I/O口（32 个），复位后 P1~P4 是准双向口并带有弱上拉电阻，P0 口是开漏输出,作为地址线使用时不用接上拉电阻而作为普通 I/O口使用时必须使用上拉电阻。

(6) 具有 3 个 16 位定时器/计数器，定时器 0 可以拆分成两个 8 位使用。

(7) 具有四个外部中断,可以设置成下降沿或低电平模式触发,掉电模式下外部中断的低电平模式可以唤醒系统。

(8) 具有一个通用同步串口,如有需要可以通过普通 I/O 口模拟串口进行扩展。

(9) 正常工作温度范围 0 - 75℃，封装为 PLCC-44 和 PDIP-40 两种，本设计为 PDIP-40 封装。

2.2 DS18B20 温度传感器介绍

2.2.1DS18B20 引脚及特点

DS18B20 的引脚及封装如图 2-3所示。

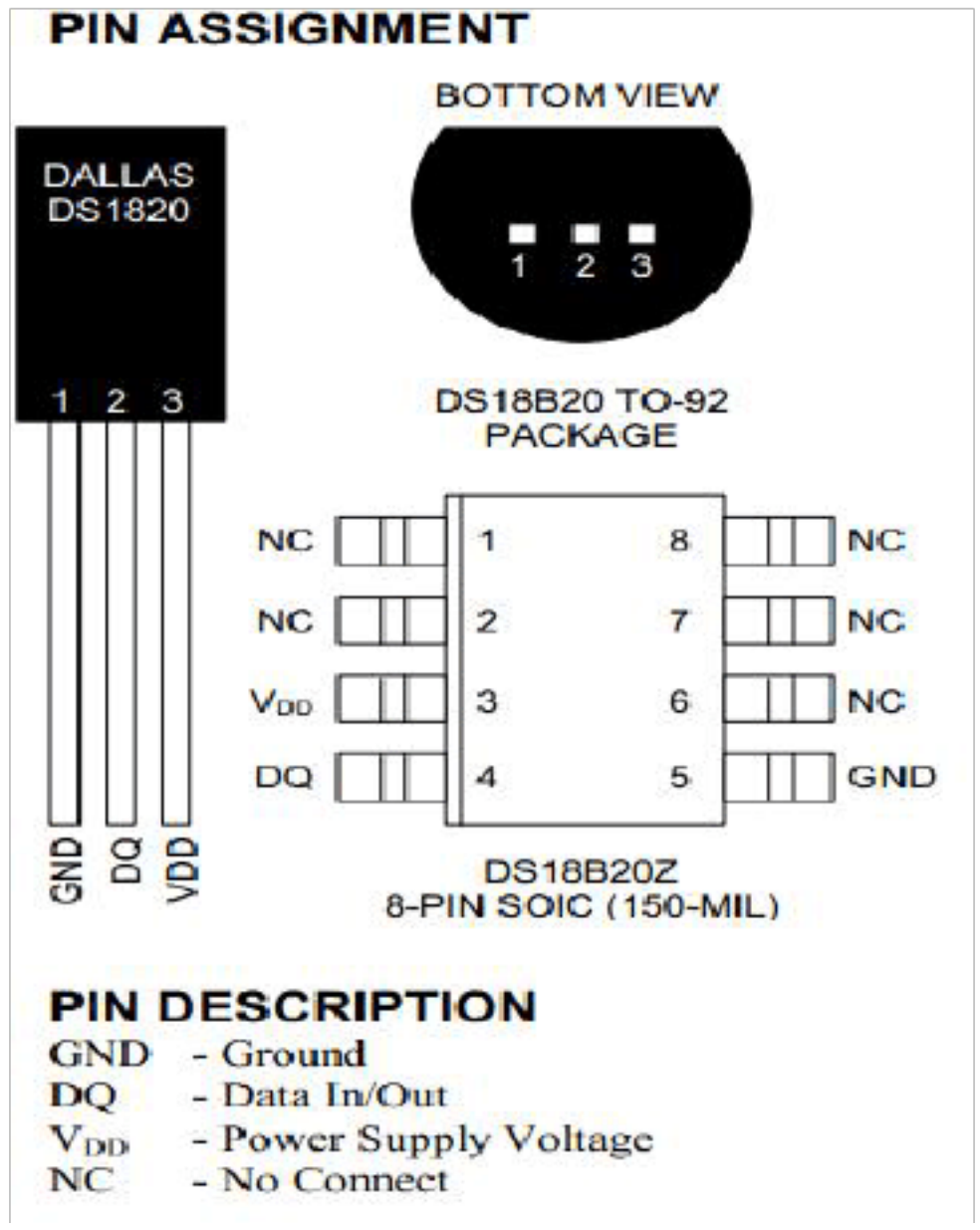


图 2-3 DS18B20 引脚及封装

DS18B20 的引脚：1.DQ 输入/输出端口；2. GND 为电源地；3. VDD +5V（寄生电源模式下该位接地即可）

DS18B20 的主要特性有一下方面，DS18B20 适应的温电压范围为 3.0~5.5v 可以直接与 3v 或 5v 单片机相连，如果使用寄生电源模式两个电源引脚可以直接接地，DS18B20 与单片机连接只需要一根数据线，通过一定的时序操作可以实现读和写的双向通信，另外 DS18B20 还有一大特色，一条数据线可以连接多个传感器 ，在不增加 I/O口的情况下实现多点测温的需求，DS18B20 设计精巧实用方便，整个功能电路集成在只有三个引脚的芯片内，形如三极管 ，其温度测量范围－55℃～＋125℃，温度测量分辨率为 9～12 位出厂设置为 12 位精度，这样在默认设置下只要把测得并转化后的温度值乘以 0.0625即可得实际温度值。

DS18B20 与单片机的连接有两种方法，外部电源供电和寄生电源供电方式，在实际使用时只需要在数据线上接上 4.7kΩ的上拉电阻即可。本设计采用外部电源供电方式如图 2-4 所示。

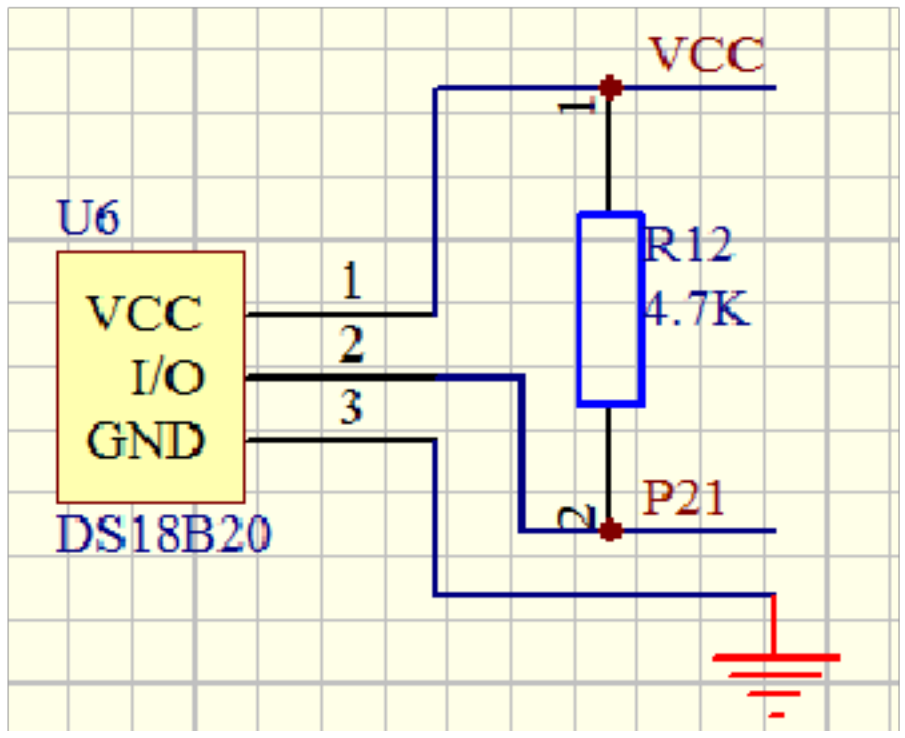


图 2-4 DS18B20 外部供电电路图

2.2.2DS18B20 内部结构及寄存器介绍

DS18B20 的内部结构如图 2-5所示。

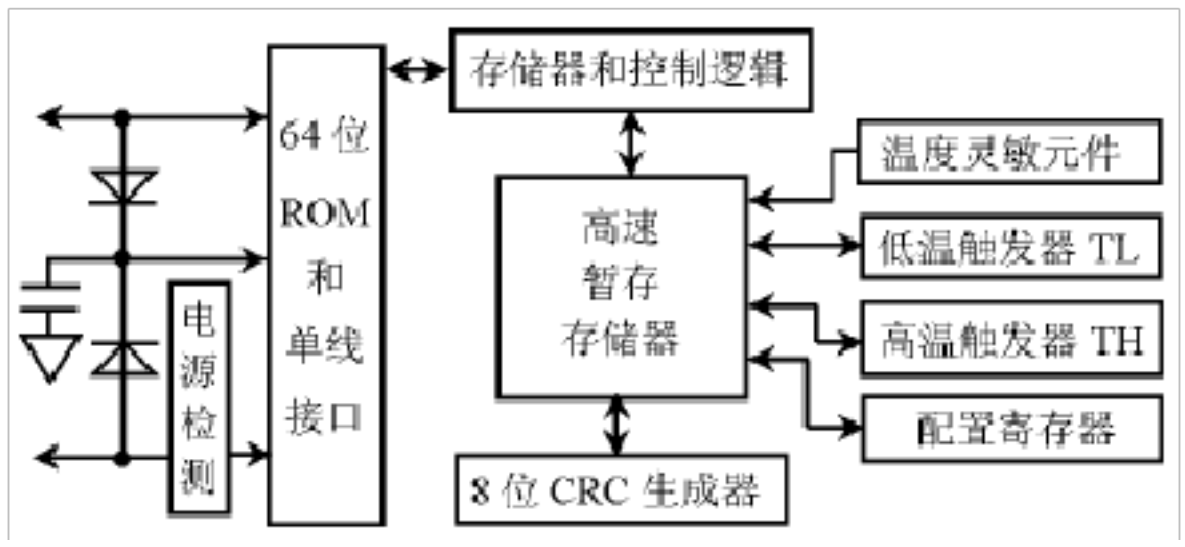


图 2-5 DS18B20 内部结构图

DS18B20 的四大数据组成部分：

光刻 ROM 中的 64 位序列号，用于实现一根总线上挂接多个 DS18B20 的目的。

温度灵敏原件，用于接收最初的温度模拟量，即电阻随温度变化。

非易失性温度报警(EEPROM)TH 和 TL 。可通过软件写入用户报警上下限值。

配置寄存器，通过 R0,R1 位设置，温度分辨率，出厂时 R0=1 ，R1=1 （即 12

位分辨率）具体设置方法如图 2-6所示。

TM	R1	R0	1	1	1	1	1

MSB LSB

R0	R1	温度计分辨率/bit	最大转换时间/ms
0	0	9	93.75
0	1	10	187.5
1	0	11	375
1	1	12	750

图 2-6 分辨率设置寄存器

下面是对 DS18B20 高速暂存存储器介绍：

高速 RAM 由 9 个字节组成如表 2-1所示。

表 2-1 DS18B20 存储器映像

温度低位	温度高位	TH	TL	配置位	保留	保留	保留	CRC 校验

温度转换命令（44H ）发出后在最长 750ms 的时间内，二进制温度值被存放于温度高位和温度低位（即高速 RAM 的第 0 和 1 字节）。这时单片机按照低位在前，高位在后的方式读取两字节数据，就可以通过一定的转换规则得到十进制温度。在转换时应注意，当 S=0(正温度)时，根据分辨率位数直接转换为十进制；当 S=1（负温度）时，应先将补码取反加一，再进行十进制转换。数据格式如图 2-7所示。

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

图 2-7 温度值格式图

DS18B20 与单片机的通信遵循着一定的时序，其时序如图 2-8所示。

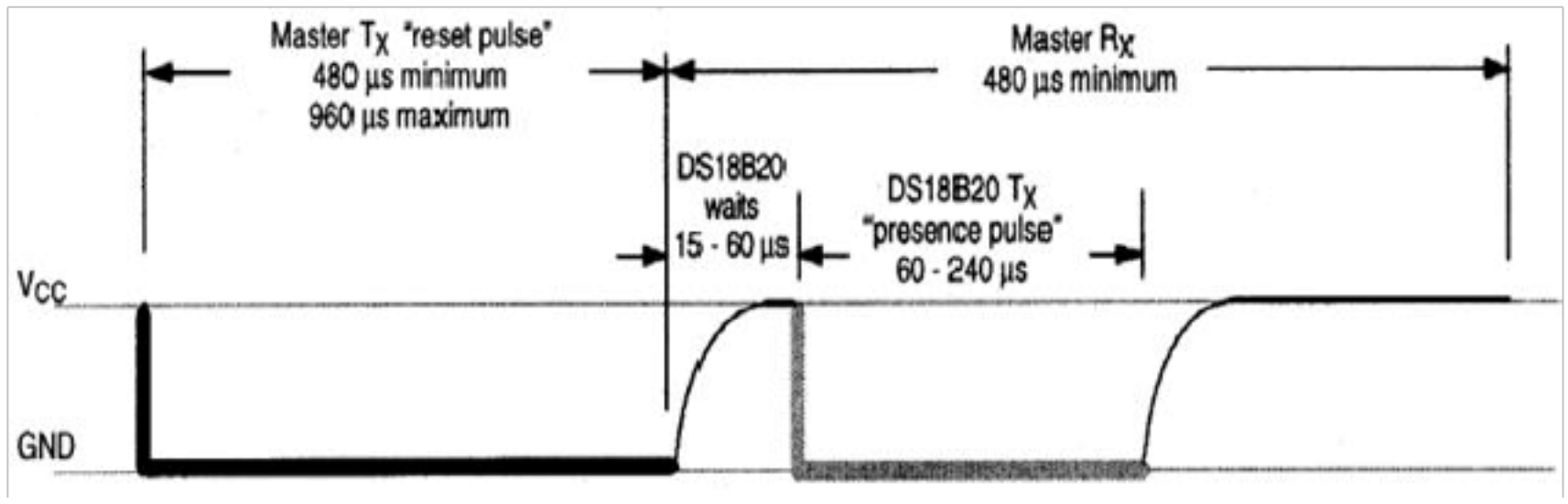


图 2-8 DS1302时序图

由图可知，使用 DS18B20 时首先对其复位，复位信号至少为 480us 的低电平，若 DS18B20 接收到复位信号则会在 15~60us 内发出一个回复脉冲，脉冲时间为 60~240us，此时控制器数据线应是高电平以便对其接收，至此，通信协议已经基本达成，接下来开始数据通信。

2.3 DS1302 时钟芯片介绍

2.3.1DS1302 引脚及内部寄存器简介

DS1302 是美国达拉斯公司推出的一款功耗低功能全的实时时钟芯片，RTC 寄存器可提供年、月、日、时、分、秒、及周的时间值，另有 31 字节静态 RAM ，采用串行三线接口与单片机进行通信，突发方式下，可以一次性传送多个字节的数据或时钟信号。工作电压宽度 2.5～5V ,用来保持片内数据时功率控制在 1 毫瓦内。具有备用电源引脚，也可以通过寄存器设置备用电源充电方式。DS1302 的引脚如图 2-9所示。

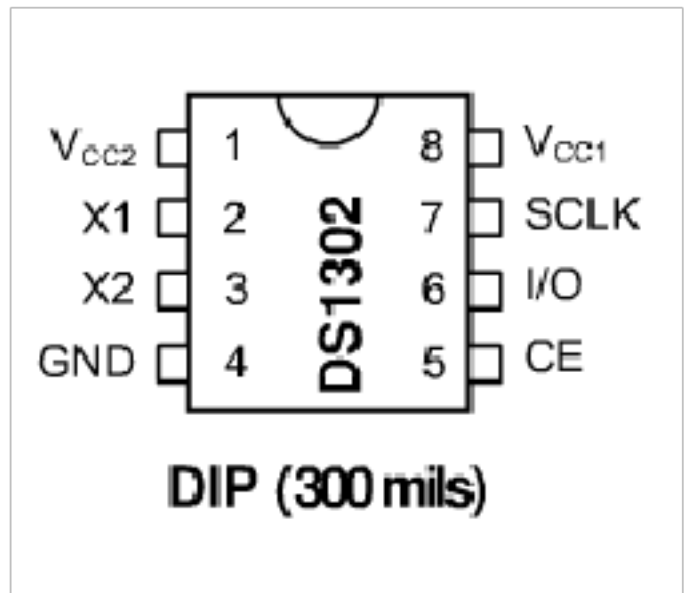


图 2-9 DS1302引脚图

其中 X1 、X2 外接 32.768KHz 晶振；GND 接地；CE 为信号使能端，数据传输时必须保持高电平，I/O 为数据输入/输出引脚；SCLK 为串行时钟，Vcc1,Vcc2 为电源供电管脚。

DS1302 关于时间和日期的寄存器如图 2-10所示，共有 12 个，编程时注意读和写的地址有所不同。

READ	WRITE	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	RANGE
81h	80h	CH	10 Seconds			Seconds				00-59
83h	82h		10 Minutes			Minutes				00-59
85h	84h	12/$\overline{24}$	0	10 $\overline{\text{AM/PM}}$	Hour	Hour				1-12/0-23
87h	86h	0	0	10 Date		Date				1-31
89h	88h	0	0	0	10 Month	Month				1-12
8Bh	8Ah	0	0	0	0	0	Day			1-7
8Dh	8Ch	10 Year				Year				00-99
8Fh	8Eh	WP	0	0	0	0	0	0	0	—
91h	90h	TCS	TCS	TCS	TCS	DS	DS	RS	RS	—

图 2-10 DS1302RTC 寄存器

DS1302 控制字，如表 2-2所示。

表 2-2 DS1302控制字

1	RAM	A4	A3	A2	A1	A0	RD/WR

控制字最高字节应该为 1，为 0 时不能向芯片写入数据，控制字从低位开始输出，第 6 位为 0 表示读取时钟数据，为 1 表示读 RAM 数据，A4 ——A0 为操作地址。在下一个时钟上升沿，写入数据，而在指令后的下一个时钟下降沿，读出数据。

2.3.2DS1302 读写时序

DS1302 读写时序如图 2-11所示。

CE 高电平开启数据传送功能，数据传输时，数据在时钟上升沿输入，时钟下降沿输出。

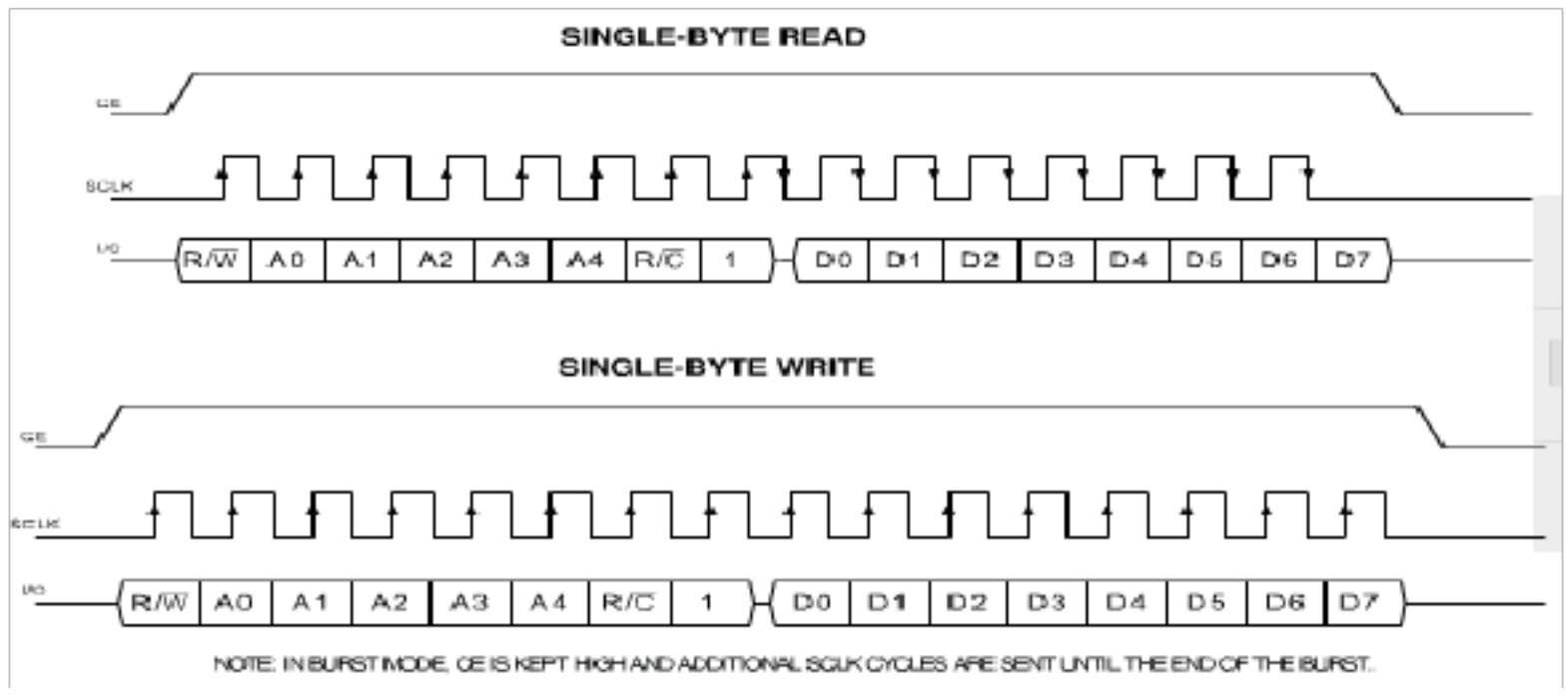


图 2-11 DS1302读写时序

2.4 HS0038 红外一体接收器介绍

2.4.1HS0038 特点介绍

红外通信有红外发射装置和接收装置组成，本设计的红外发射装置为载波 38KHz 的普通红外遥控器。HS0038 能够接受红外信号，并能对信号放大，检波，整形输出单片机可以识别的 TTL 电平。单片机经过一定的算法把接收到的数据进行解码。便可以得到遥控器键值。应当注意 HS0038 输出的数据正好和发射端的电平反向。

2.4.2HS0038 电路及应用

HS0038 引脚及封装如图 2-12所示。

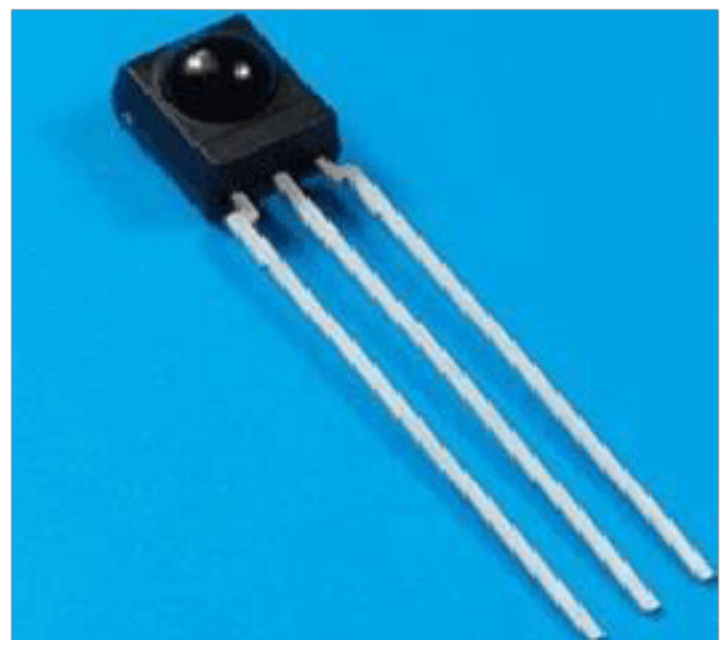


图 2-12 HS0038实物图

从左到右，引脚依次是：I/O数据端口、GND 接地、VCC 接+5V 电源。

HS0038 在单片机系统中常用的电路如图 2-13所示。

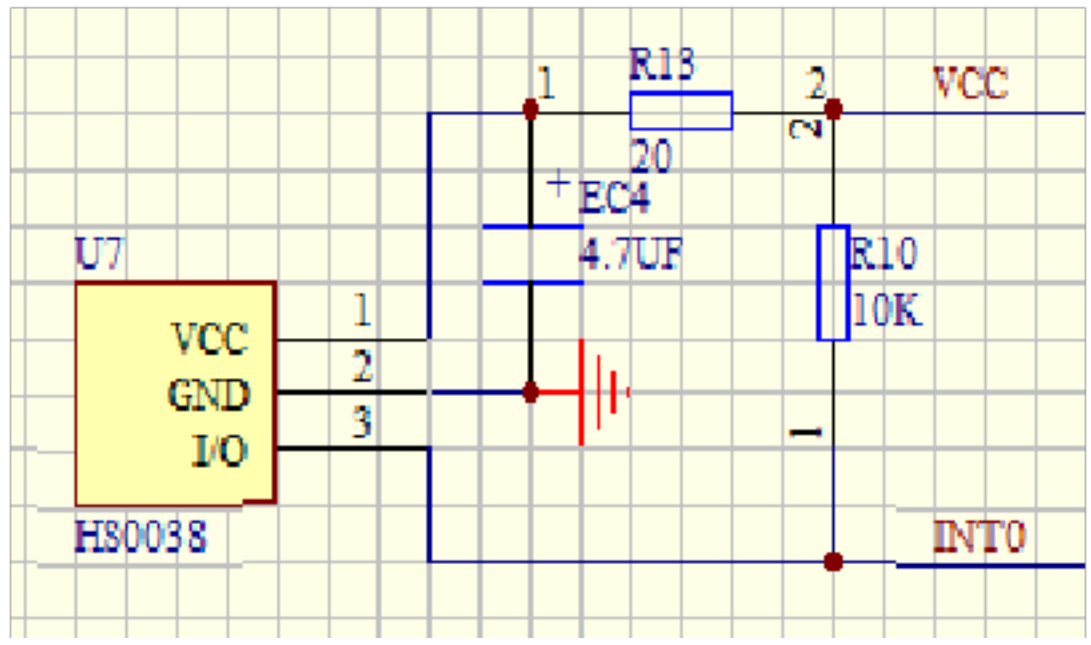


图 2-13 HS0038应用电路图

2.5 LCD12864 液晶介绍

2.5.1LCD12864 性能概述及引脚功能

本设计采用带汉字字库的液晶模块，内置 8192 个汉字 128 个字符还具有 64*256 的 GDRAM 可以用来绘制图形。液晶正常工作的电压宽度为 3.3~5V,显示分辨率为 128×64。应用中可以通过写入命令字来实现多种功能，如：开关光标，反白显示，屏幕移位，睡眠等。该液晶 20 个引脚与单片机的连接如图 2-14 所示。LCD12864 具有 20 个引脚其功能如表 2-3 所示。

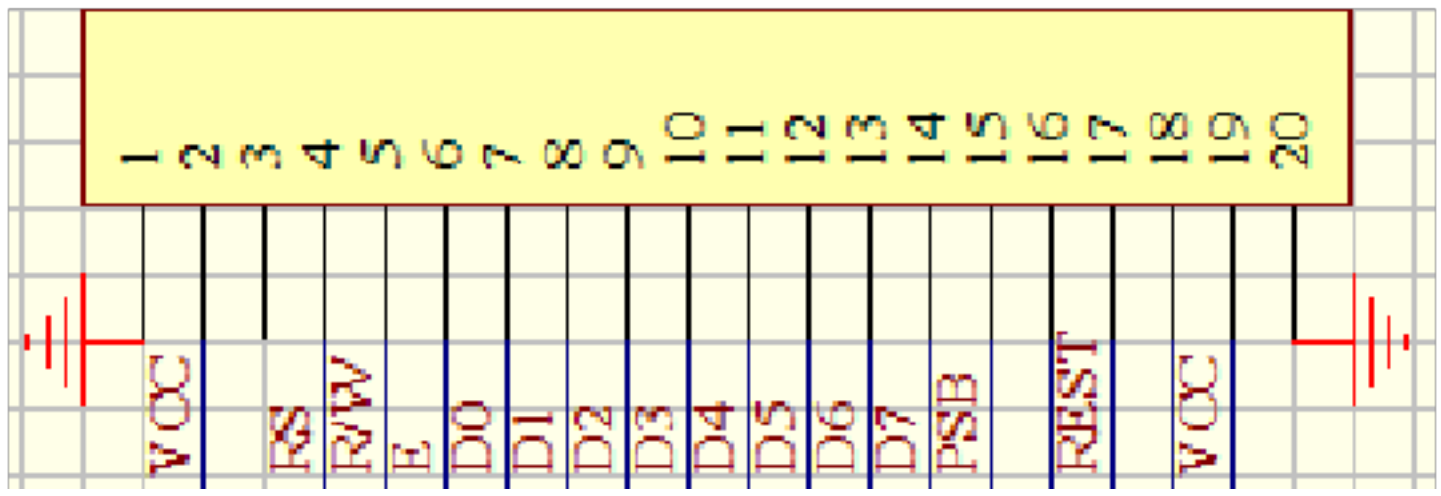


图 2-14 LCD12864 与单片机连接图

由于本设计采用 8 位并行数据传输方式，故其第 15 引脚 PSB 直接接高电平即可。此时，RS 引脚为指令/数据选择引脚。另外，18 脚和 19 脚为背光灯，可以根据需要接上可调电阻来实现对背光亮度的调节。

表 2-3 LCD12864 引脚功能图

引脚号	引脚名称	方向	功能说明
1	VSS	-	模块的电源地
2	VDD	-	模块的电源正端
3	V0	-	LCD 驱动电压输入端
4	RS(CS)	H/L	并行的指令/数据选择信号：串行的片选信号
5	R/W(SID)	H/L	并行的读写选择信号：串行的数据口
6	E(CLK)	H/L	并行的使能信号：串行的同步时钟
7	DB0	H/L	数据 0
8	DB1	H/L	数据 1
9	DB2	H/L	数据 2
10	DB3	H/L	数据 3
11	DB4	H/L	数据 4
12	DB5	H/L	数据 5
13	DB6	H/L	数据 6
14	DB7	H/L	数据 7
15	PSB	H/L	并/串行接口选择：H-并行；L-串行
16	NC		空脚
17	/RET	H/L	复位 低电平有效
18	NC		空脚
19	LED_A	-	背光源负极（LED-0V）
20	LED_K	-	背光源正极（LED+5V）

2.5.2LCD12864 读写时序及用户指令集

对 LCD12864 液晶的操作不外乎通过单片机不断进行读写，单片机写数据到液晶模块的时序图如图 2-15所示。

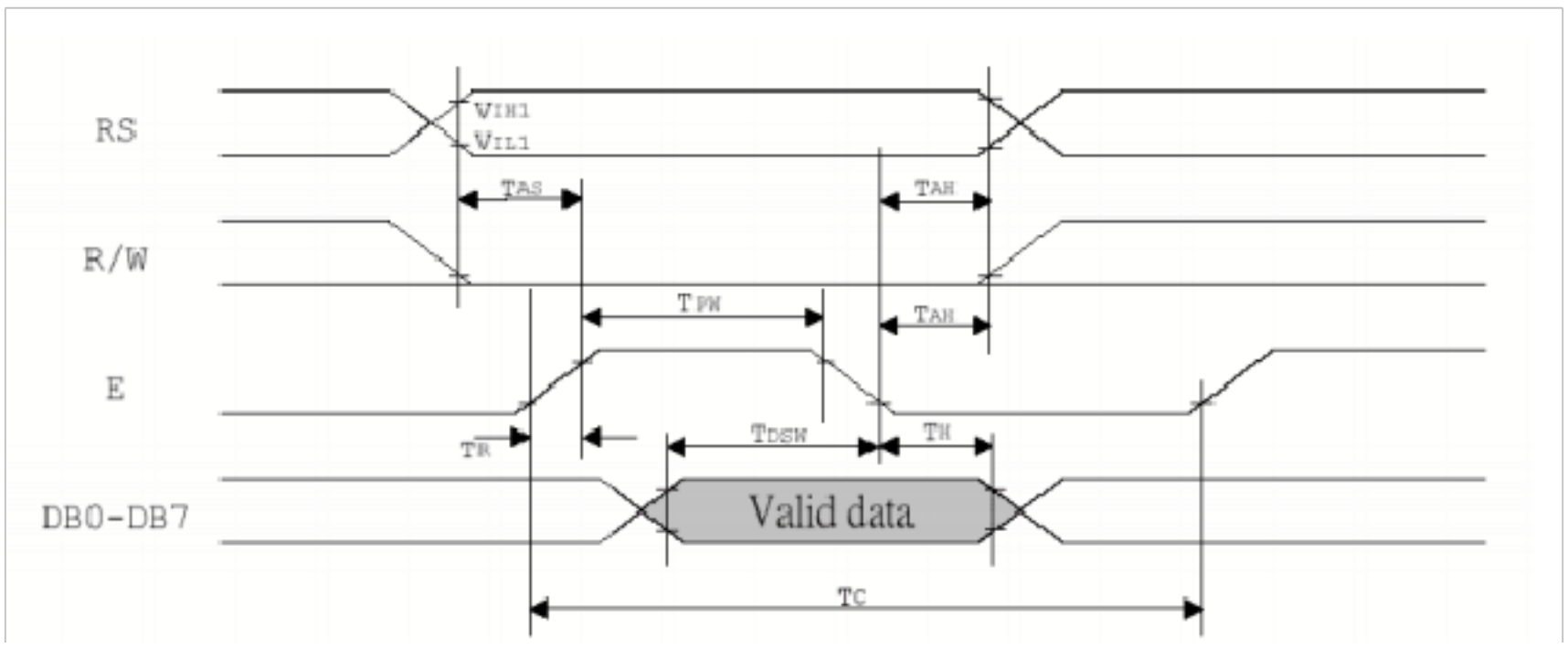


图 2-15 单片机写数据到液晶模块

单片机读取液晶模块数据的时序图如图 2-16所示。

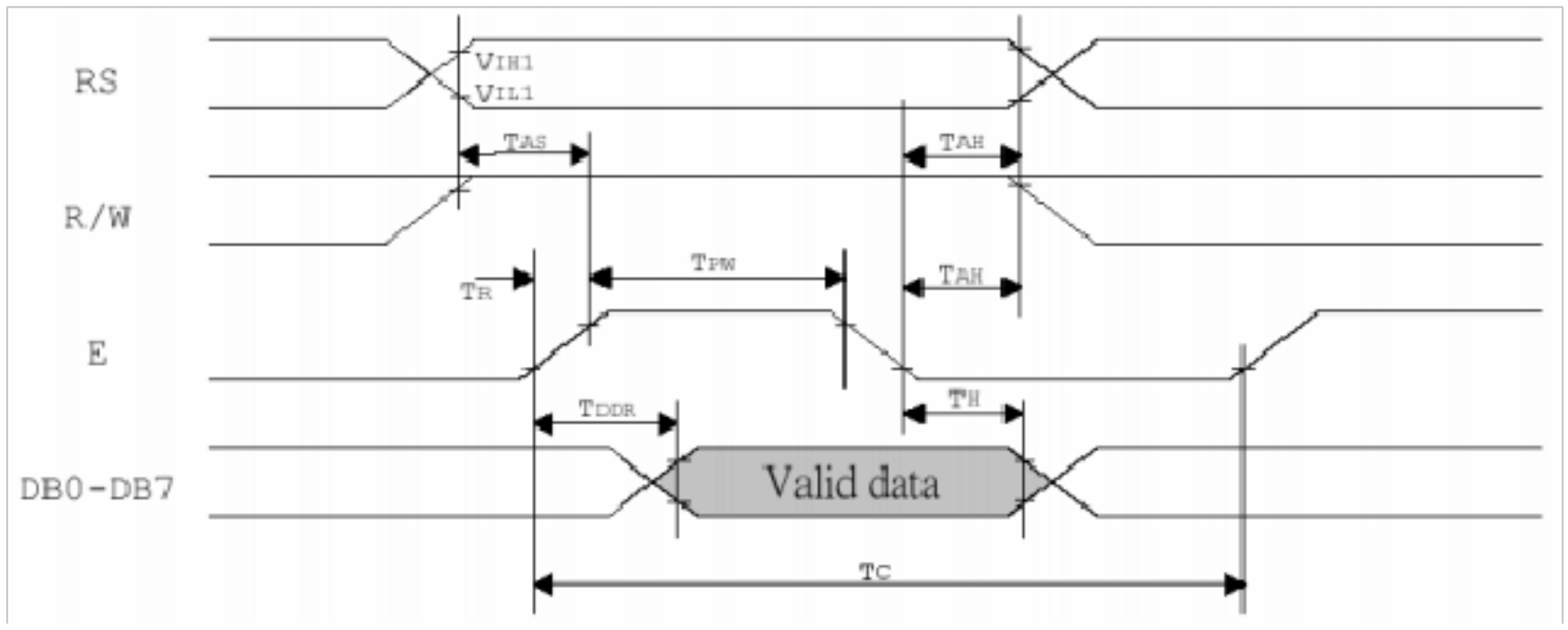


图 2-16单片机读取液晶模块数据

由图 3-14可知写数据或命令的操作大致如下，RS 为高时写数据，RS 为低时写指令；RW 引脚为低电平进行写操作； E 引脚设置为高允许数据传送，接着便把所传数据放于数据引脚，E 引脚再次拉低便完成了写操作。

LCD12864 有两种指令集，基本指令集（RE=0 ）和扩充指令集（RE=1 ），写指令时 RS=0,RW=0 。通过正确的时序写指令可以设置 LCD12864 的很多功能，如写入 0x01 指令即可清屏并把 DDRAM 地址计数器调整为“00H ”,再如写入指令“0xc”即可打开显示关闭游标。

2.6 其他硬件电路

2.6.1STC89C52RC 最小系统的设计

本设计的单片机最小系统主要包括，STC89C52 单片机，12M 外接晶振，上电复位电路三个部分。

最小系统晶振电路如图 2-17所示。

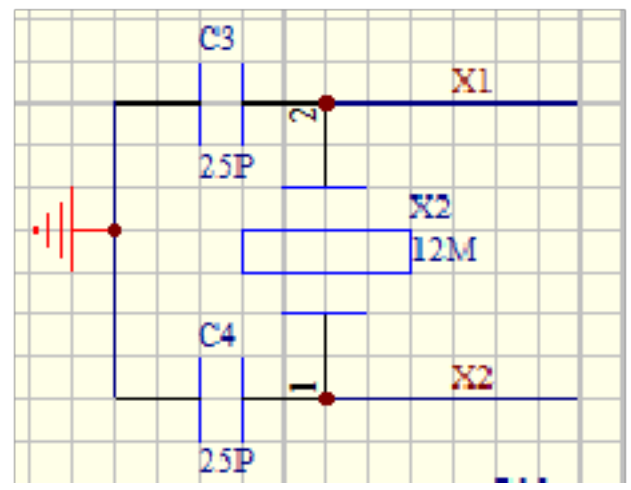


图 2-17 晶振电路

晶振两个引脚分别接于单片机的 XTAL0　和 XTAL1　引脚，电容值的选择范围为 5~30PF，电容对晶振频率具有微调作用。

最小系统复位电路如图 2-18所示。

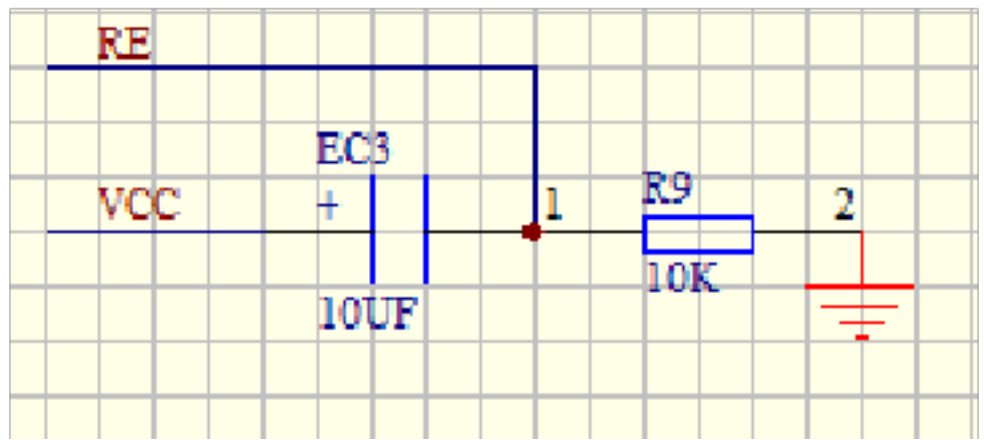


图 2-18 上电复位电路

单片机复位电路的设计必须能满足上电震荡稳定后，保持至少两个周期的高电平。本最小系统采用 10uf的电容，经验证完全满足要求。由于本电路上电即可以使单片机复位，又具有电源开关，故省去了按键复位开关。

另外，由于本设计采用单片机内部存储器存储程序，故单片机的第 31 引脚（EA ）接高电平，用来访问片内程序。

2.6.2系统电源电路的设计

由于所选用单片机为 5V　单片机，为得到较稳定的 5V　电压，本设计采用了 L7805cv 三端集成稳压管。输出电压 4.75-5.25V 最大输入电压 35v，压差 3-5v时工作在理想状态。输出电流可达 1.5A（需做好散热），同时 L7805cv　内部含有限流保护电路和过热保护电路，防止负载过大温度过高烧坏器件。本设计中 L7805cv 的应用电路如图 2-19 所示。

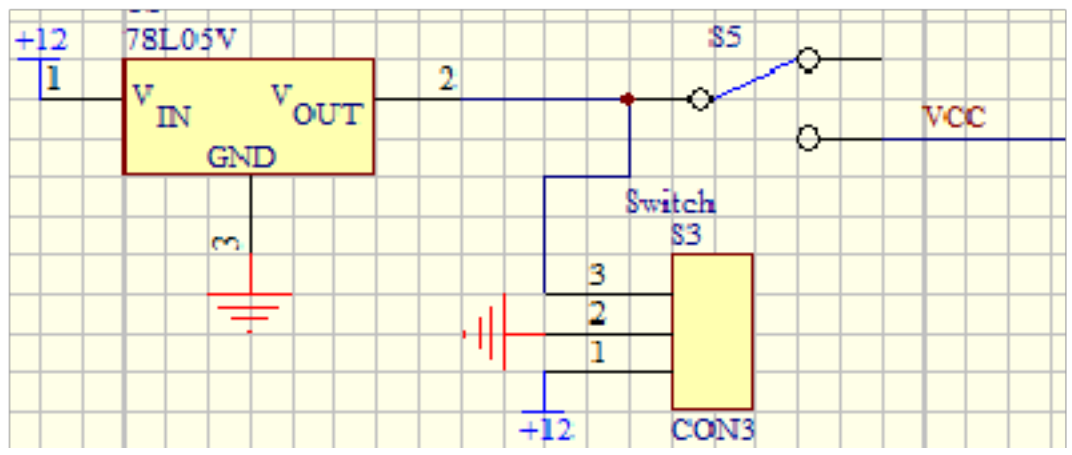


图 2-19 L7805cv稳压电路图

3 软件开发平台及软件

3.1 开发环境介绍及程序总体结构

3.1.1开发环境介绍

本设计采用 KEIL C51 集成开发环境，它具良好用户的界面，使用极为方便。支持汇编语言，C 语言及其混合编程，能够兼容绝大多数 51 系列单片机的程序设计和仿真。在本设计中，所有的程序采用 C 语言编写，最后把调试通过的程序编译成二进制文件并通过串口下载到目标板。

3.1.2程序总体结构

由于整个系统涉及到的外设模块较多，故采用了多文件模块化编程，这样大大提高了程序的可读性，也为以后可能遇到的程序移植和升级提供了方便。整个设计的程序结构如图 3-1所示。

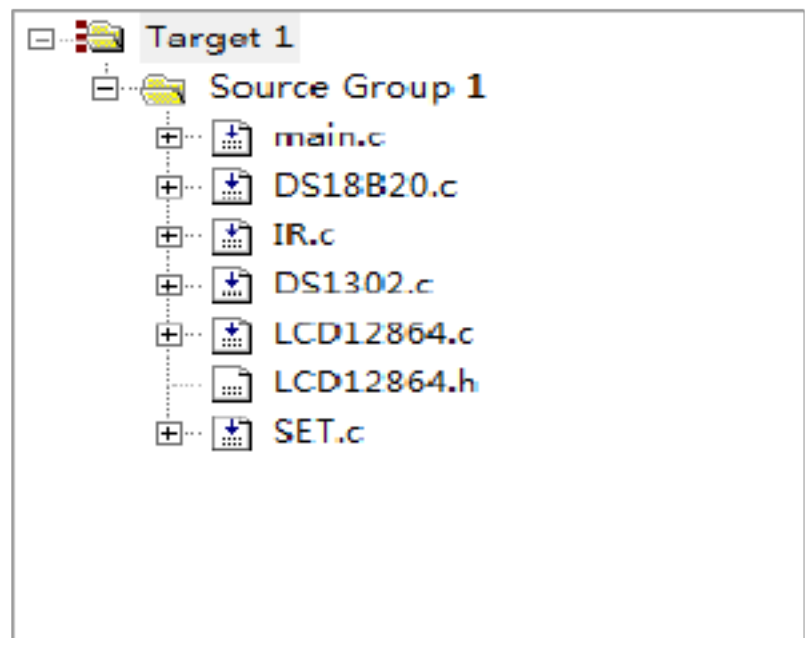


图 3-1 程序结构图

其中，main.c是设计主函数包含了各种外设的初始化及其功能函数的调用，DS18B20.c、IR.c、DS1302.c、LCD12864.c 分别为温度传感器、红外接收器、时钟芯片和液晶的驱动程序，SET.c 是用来定时、定温的功能模块。

3.2 DS18B20 温度程序模块

3.2.1DS18B20 操作流程

使用 DS18B20 时首先应遵循图 2-8所示的时序图，然后按照流程进行具体操作。DS18B20 操作流程如图 3-2所示。

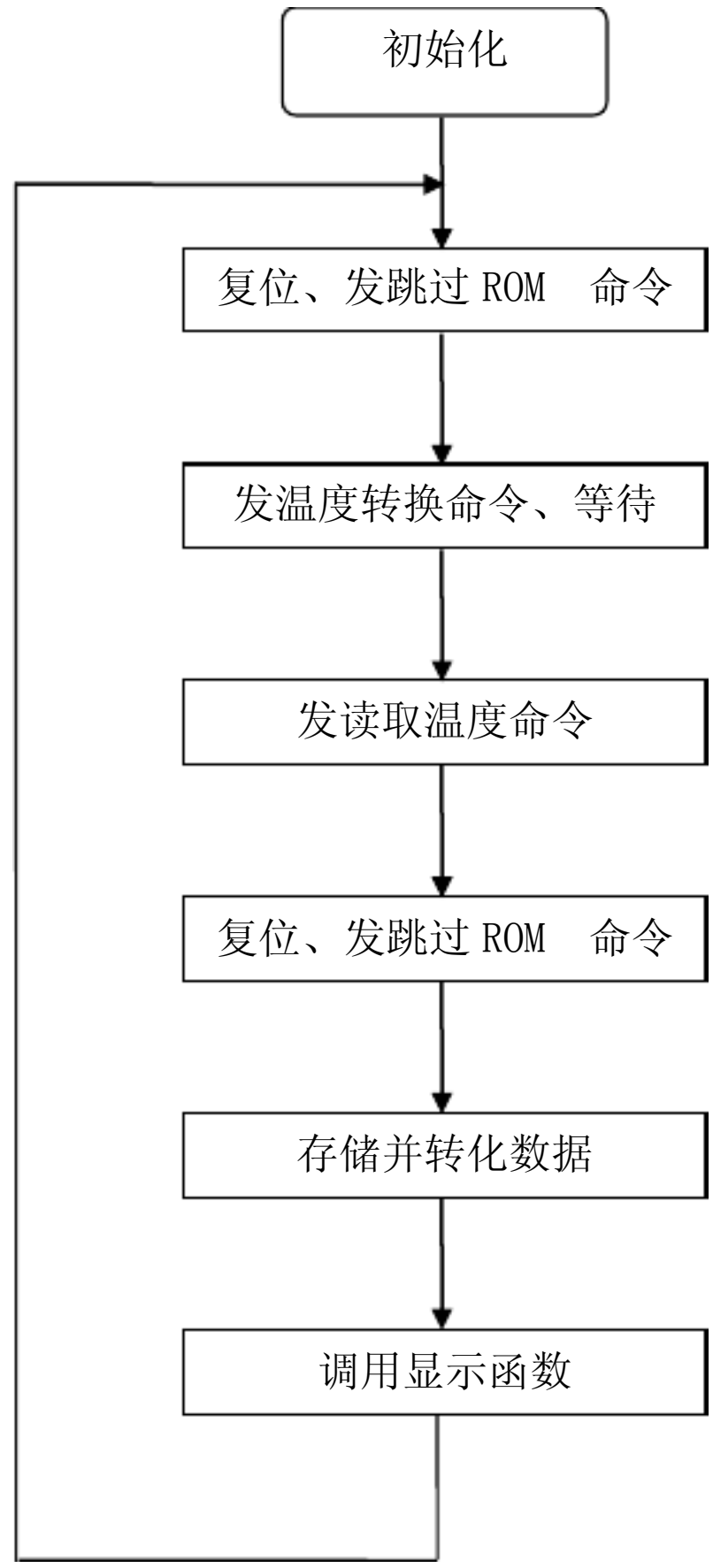


图 3-2 DS18B20 操作流程

3.2.2DS18B20 重点代码及其分析

DS18B20 程序的核心部分是利用单片机读取芯片内部的温度值，其读函数如下，

```
/*************************************************************
*从 DS1820 中读出数据
*************************************************************/
uchar DS1820_RData()
{
    uchar i,j,Tmep;
    for(i=8;i>0;i--)
{
    Tmep>>=1;
```

```
    DS1820_bit=0;          /低电平,产生读信号
    for(j=2;j>0;j--);      /延时 4us
    DS1820_bit=1;          /释放总线,准备读数据
    for(j=4;j>0;j--);     /延时
    if(DS1820_bit==1)
{Tmep|=0x80;}
    for(j=30;j>0;j--);  /延时
    DS1820_bit=1;          /拉高数据线，准备读下一位
}
    return(Tmep);         /返回数据
}
```

通过上面的读函数，温度值被保存到 Tmep 变量内，接下来便可以按照一定的规则对 Tmep 的值进行转化，最后显示在液晶屏上。另外，应当注意在每次获取温度值时都要对 DS18B20 进行复位操作。

3.3 DS1302 时间程序模块

3.3.1DS1302 总体操作及基本读写功能函数

本设计中 DS1302 操作主要包括读取内部时间值和修改时间值两部分，DS1302 模块总体操作如图 3-3所示。

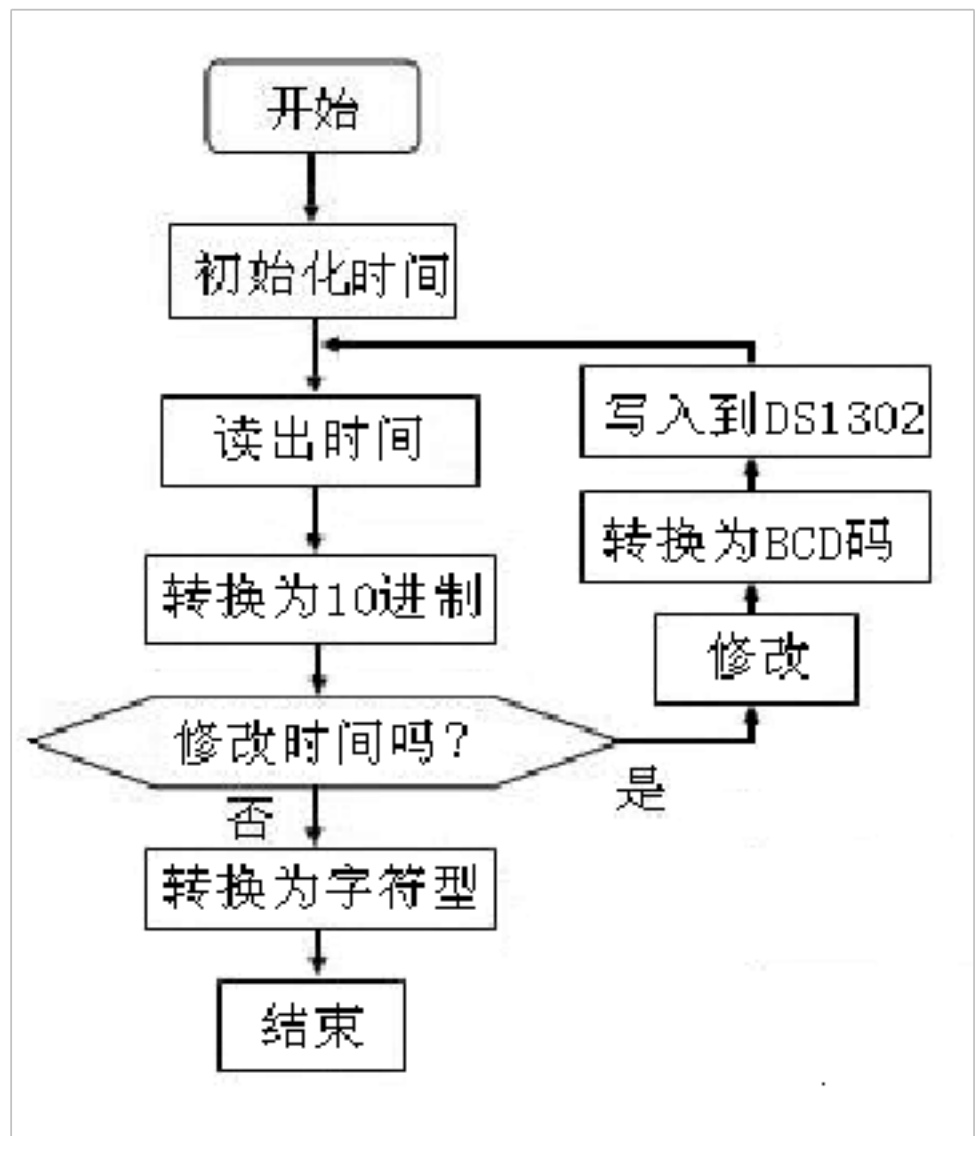


图 3-3 DS1302操作总流程

DS1302 读写操作流程如图 3-4所示。

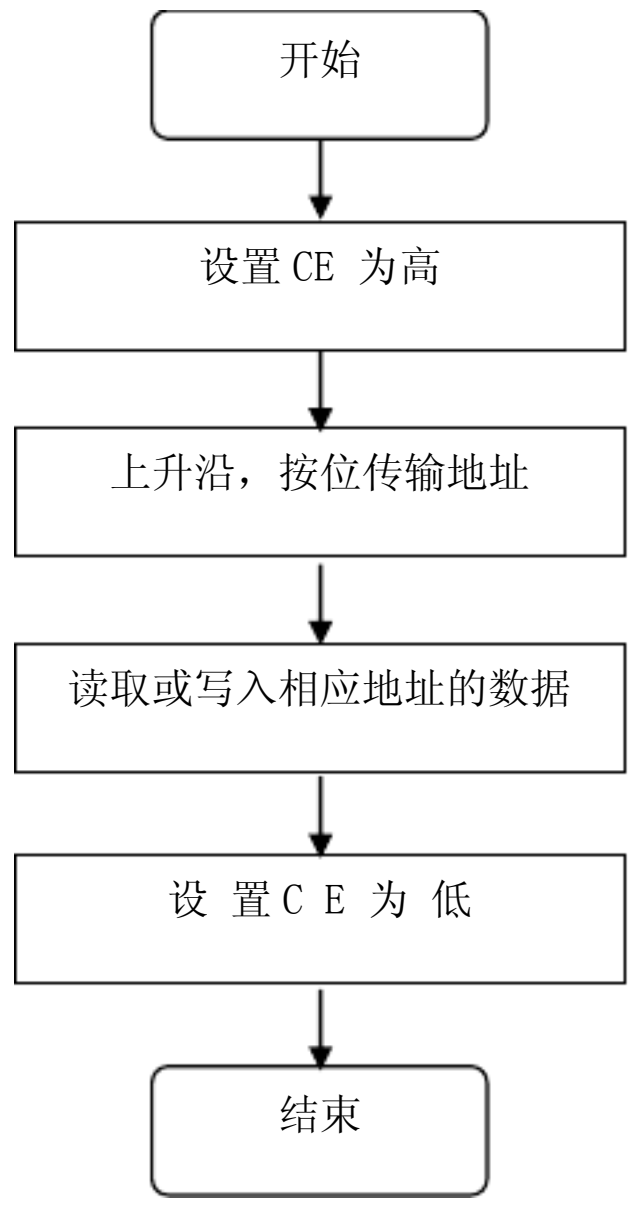


图 3-4 DS1302读写操作流程

在对 DS1302 操作时，只要调用 Write1302_Cmd_Dat(),和 Read1302()就可以实现其与单片机之间的数据交流。下面是读写函数的实现。

```
/*********************************************************************
/向 DS1302 写地址和数据， Addr 地址，Dat 数据
*********************************************************************/
void Write1302_Cmd_Dat(unsigned char Addr, unsigned char Dat)
{   DS1302_RST = 0;
    DS1302_CLK = 0;
    DS1302_RST = 1;
    DS1302_WriteByte(Addr);          //地址，命令
    DS1302_WriteByte(Dat);        //写 1Byte数据
    DS1302_CLK = 1;
    DS1302_RST = 0;
}
/*********************************************************************
/读取 DS1302 某地址的数据,
*********************************************************************/
 uchar Read1302(unsigned char Addr)
{   uchar Dat;
```

```
    DS1302_RST = 0;
    DS1302_CLK = 0;
    DS1302_RST = 1;
    DS1302_WriteByte(Addr|0x01); //以写数据时的地址为参考，读时应加 1
    Dat = DS1302_ReadByte();     //读取数据到变量 Dat
    DS1302_CLK = 1;
    DS1302_RST = 0;
    return(Dat);
}
```

3.3.2读取时间数据并转化成字符串

由于时间数据包含年、月、日、时、分、秒、周，为了方便可以定义一个时间类型结构体，把转换前后的数据都作为这个结构体的成员。

```
typedef struct
{   unsigned char Week;
    unsigned char Second;
    unsigned char Hour;
    unsigned char Minute;
    unsigned char Hour;
    unsigned char Month;
    unsigned char Day;
    unsigned charYear;
    unsigned char Str_Time[9];
    unsigned char Str_Year[5];
    unsigned char Str_Month[3];
    unsigned char Str_Day[3];
}
    struct_TIM;
```

然后，通过 Read1302()函数读取各个地址处的时间数据分别保存于 Second、Minute 等变量中并转换成十进制，如读取并转换秒值可以用以下两个语句：

```
    temp = Read1302(0x80);低四位是秒的个位，高三位是秒的十位（原来是 BCD 码）
    Time->Second = ((temp &0x70)>>4)*10 + (temp &0x0F);
```

接着就可以把数值转换成字符串了，仍以秒值为例转换语句如下：

```
    Time->Str_Time[0] = Time->Second/10 + '0';十位//
    Time->Str_Time[1] = Time->Second%10 + '0';个位/
```

至此，所得数据便可以方便的显示在液晶屏上了。

3.4 LCD12864 液晶驱动程序

3.4.1LCD12864 初始化

LCD12864 初始化流程如图 3-5所示。

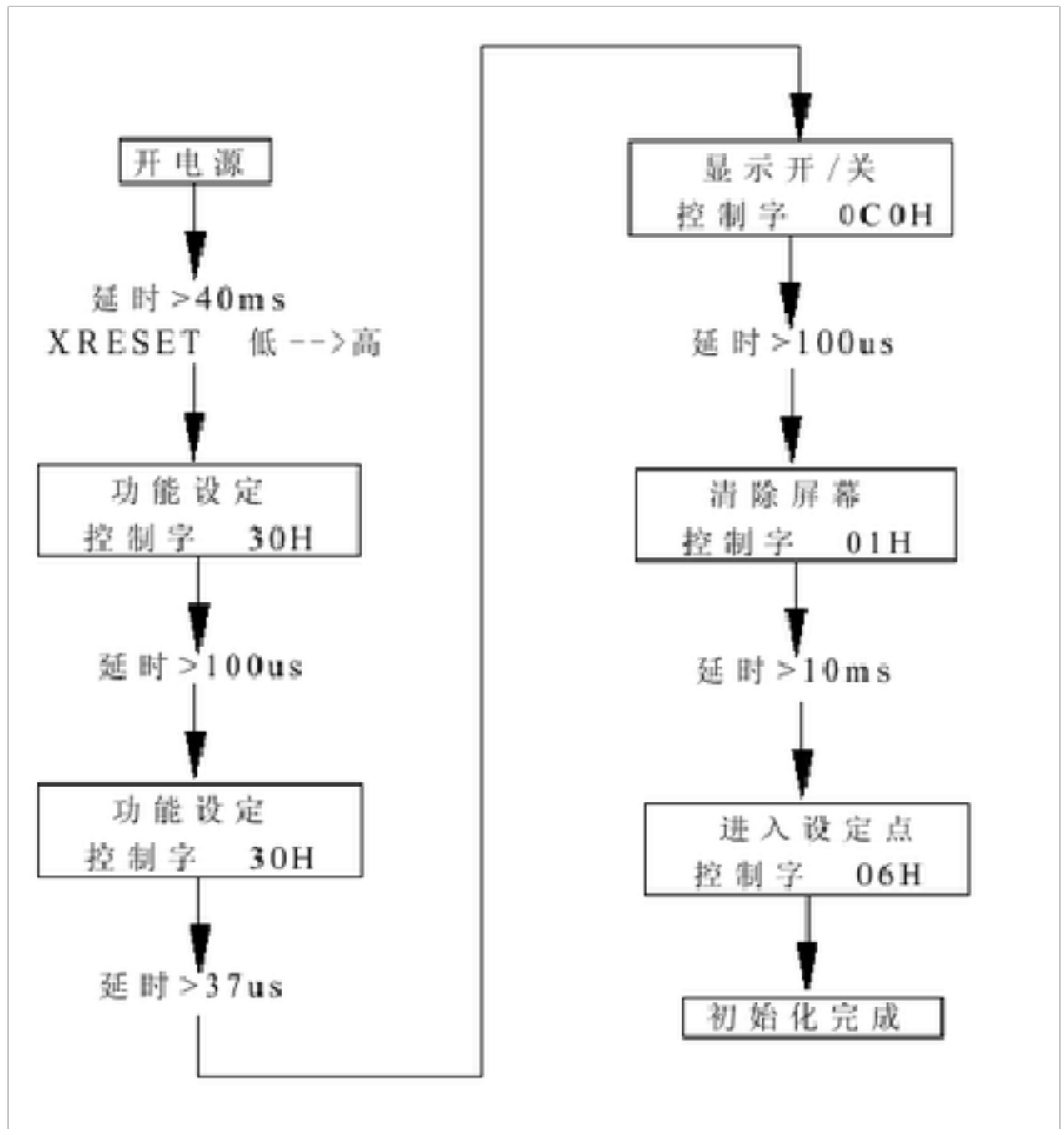


图 3-5 1286液晶初始化

LCD12864 液晶屏使用前先要进行初始化，已设定不同的功能和模式，本设计中液晶初始化为基本指令集，开显示关闭游标，光标从左向右加 1 移动，初始化通过 Init_12864()函数完成。

3.4.2写数据到 LCD12864 功能函数

LCD12864 的功能函数主要是显示从其他模块采集到的并转化后的数据，这里主要提供了两种显示函数，writestring()和 writedat()两个函数，writestring()用来显示从一个地址开始的字符串，writedat()函数用来在某个地址显示一个字符。下面是相应的实现代码。

```
/从地址 addr_12864开始，写字符串到 LCD
void writestring(uchar addr_12864 ,uchar * str)
{
uchar i = 0;
Write12864_Cmd_Dat(addr_12864,0);
while(str[i] != '\0')
 {
   Write12864_Cmd_Dat(str[i++],1);
 }
}
/在地址 addr，写一个字符到 LCD
void writedat(uchar addr ,uchar dat)
{
    Write12864_Cmd_Dat(addr,0);
    Write12864_Cmd_Dat(dat,1);
  }
```

3.5 HS0038 红外解码程序模块

3.5.1红外遥控器编码介绍

在进行红外解码之前首先要对本设计采用的红外遥控器编码方式进行简要介绍。遥控器发送一组完整数据（如图 3-6所示），首先是 9ms 高电平和 4.5ms 低电平组成的前导码，紧接着是 8 位地址码及其反码，最后便是 8 位操作码及其反码。具体的编码方式为脉宽调制的串行码，用 0.56ms 的高电平和 0.56ms 的低电平表示二进制 0，以 0.56ms 的高电平和 1.68ms 的低电平表示二进制 1，编码形式如图 3-7所示。

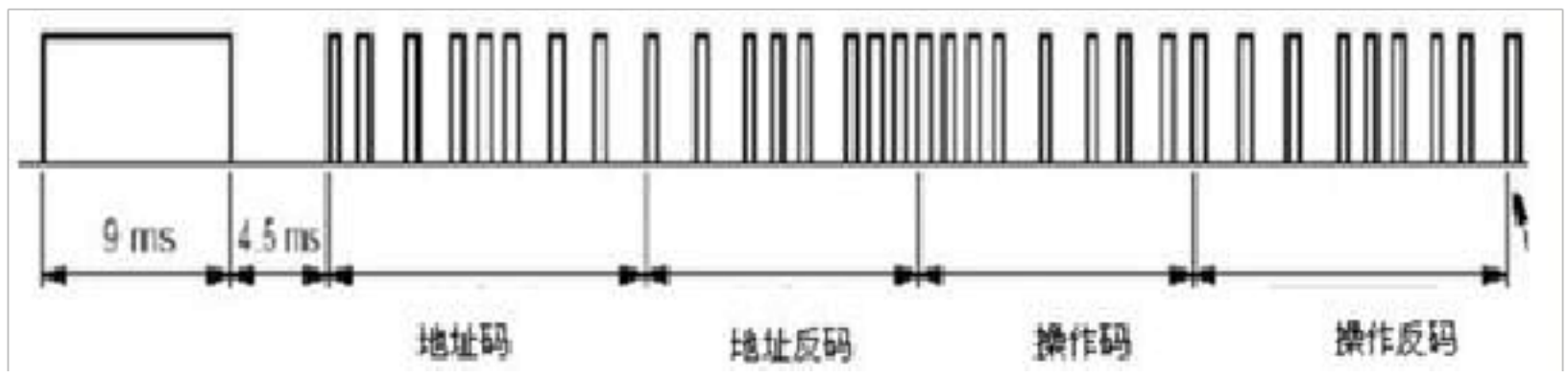


图 3-6遥控器发送一组完整数据



图 3-7遥控器编码的 0 和 1

3.5.2红外解码程序流程及重要代码分析

本设计中红外遥控器作为控制单片机运行的按键，为了更好的实现按键功能红外接收端接在了单片机的外部中断 0 引脚，另外遥控器编码采用脉宽调制方式，为了更精确的记录脉宽时间需要启用定时器功能，所以程序开始要进行中断和定时器的初始化。为了保证精度，定时器初始化为工作方式 2 即 8 位自动重装模式。

中断触发方式采用下降沿触发，中断程序只需要记录两个中断之间时间值并保存，然后对相应脉冲的时间值与标准的脉冲时间进行比较，这样得出了是 0 还是 1。由于中断函数尽量要短，进行解码的工作尽量不要在中断函数中进行。下面是中断函数。

```
/*********************************************************
晶振 12M ，一个机械周期为 12/12000000 为 1 微秒，则每一次中断
256*0.001ms=0.256ms
*********************************************************/
void timer0() interrupt 1
{
  timer++;
}
/*********************************************************
startflag初始值位 0，第一次进入中断标记为 1，同时时间清零，如果
经判断有引导码出现，就开始记录了。
*********************************************************/
void int0() interrupt 0
{
    if(startflag)
{
    if( timer>45) //9+4.5ms的引导码 13.5/0.256 =52   取低于 52 的数即可
{   ir_num=0;}
    ir_time_data[ir_num]=timer;
```

```
    timer=0;
    irnum++;
    if(ir_num==33)
        {
            Ir_num=0;
            Ir_end=1; //一组接收完成
        }
    }
    else
{
    startflag=1;/第一次进来时被标记为 1，timer清 0
    timer=0;
}
}
```

另外，根据通过实际应用发现，外部光源特别是太阳光对红外通信干扰较大，为了避免光源干扰可能带来的麻烦，这里采用了软件解决的方式。按键按下后把得到的数据分别与操作码和操作反码比较，每次比较通过后要立即清零。下面是一段示例代码：

```
if(ircode[2]==0x09)
{   ircode[2]=0x00;
    if (ircode[3]==0xf6)
    ircode[3]=0x00;
    具体功能代码
    ……
}
```

这样就保证了即使干扰数据能进入第一层，但进不了第二层，由于进入第一层后数据被清除，下一次连第一层也进不去了，从而避免了由于干扰出现随机数对系统功能的影响。

3.6 定时定温功能的程序实现

本系统具有随时设定时间值和温度值的功能，要达到这样的目的需要满足这样几个条件，第一、能进入设置模式，第二、能通过按键选择要修改的项并能对其数值增大和减小，第三、能顺利返回。该流程可以通过图 3-8表示。

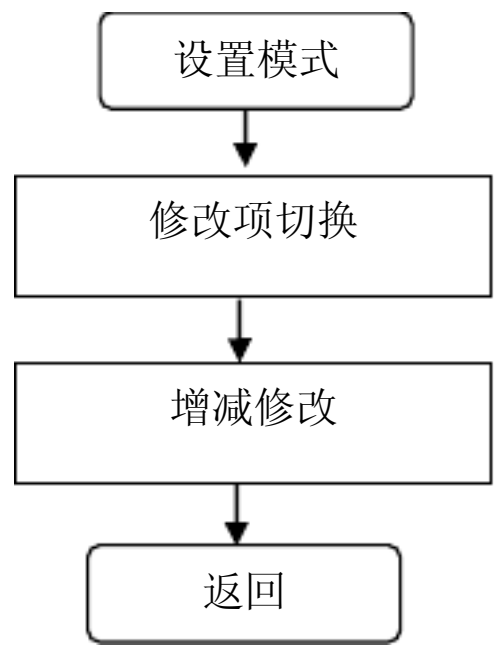


图 3-8 时间温度设置流程

以上工作完成后，设置功能已经能够实现，只需要在主函数中调用pare_temp_time()函数就可以通过比较设定值与实际值，来执行继电器开关灯和开关空调动作了。

4 系统调试及设计总结

4.1 系统调试

4.1.1最小系统调试

本设计的电路图由 Protel99SE绘制，各功能模块的实际电路由万用板人工焊接。最小系统的正常运行是整个系统得以正常运行的前提。焊接完成最小系统后，首先用软件平台编写一个能使 P2.0口一秒间隔高低电平变化的 C 程序，然后把编译生成的二进制文件通过串口下载到单片机。这时再在 P2.0引脚连接发光二极管，上电复位后，发现二级管能够一秒间隔出现亮灭循环。

通过以上实验能够证明单片机最小系统已经正常运行，下面就可以焊接其他功能模块了。

4.1.2其他各模块调试过程

由于本设计的所有数据都以可见方式显示在液晶上，所以在焊接其他模块前需要使液晶屏能够正常显示。焊接好液晶模块后，通过上面的方式下载程序使液晶显示出几个汉字和字符，液晶调试通过。接着焊接温度传感器和时钟芯片，并通过各自的程序在液晶上显示出数据。红外模块的调试是通过遥控继电器开关来

完成的，焊接好红外和继电器模块后，通过编程使遥控器某按键每次键按下时 P2.0 电平高低翻转，把继电器连接在 P2.0引脚，上电复位后按下遥控器，继电器发出啪啪的响声，调试通过。

在实际调试时钟模块时，时间数据经常发生抖动和随机跳变，但在下一次刷新时又变正常，这说明时钟芯片与单片机的通信是正常的。经过多次对电路的检查怀疑是由于 DS1302 的 I/O口驱动能力不足造成的，而且在焊接的电路中从 DS1302 到单片机之间的连线又很长，于是在 DS1302 的 I/O口和 VCC 之间加了一个 10K 的上拉电阻，问题得到圆满解决。 由此得出了一个结论 DS1302 的驱动能力很是有限的在和单片机的通信中，要考虑到加上拉电阻的可能。

各个模块调试通过后就可以把各模块程序加以组合，实现整体功能了，由于采用了多文件编程要注意声明全局变量。图 5-1即为调试用电路实物及显示效果，各项功能均已正常。

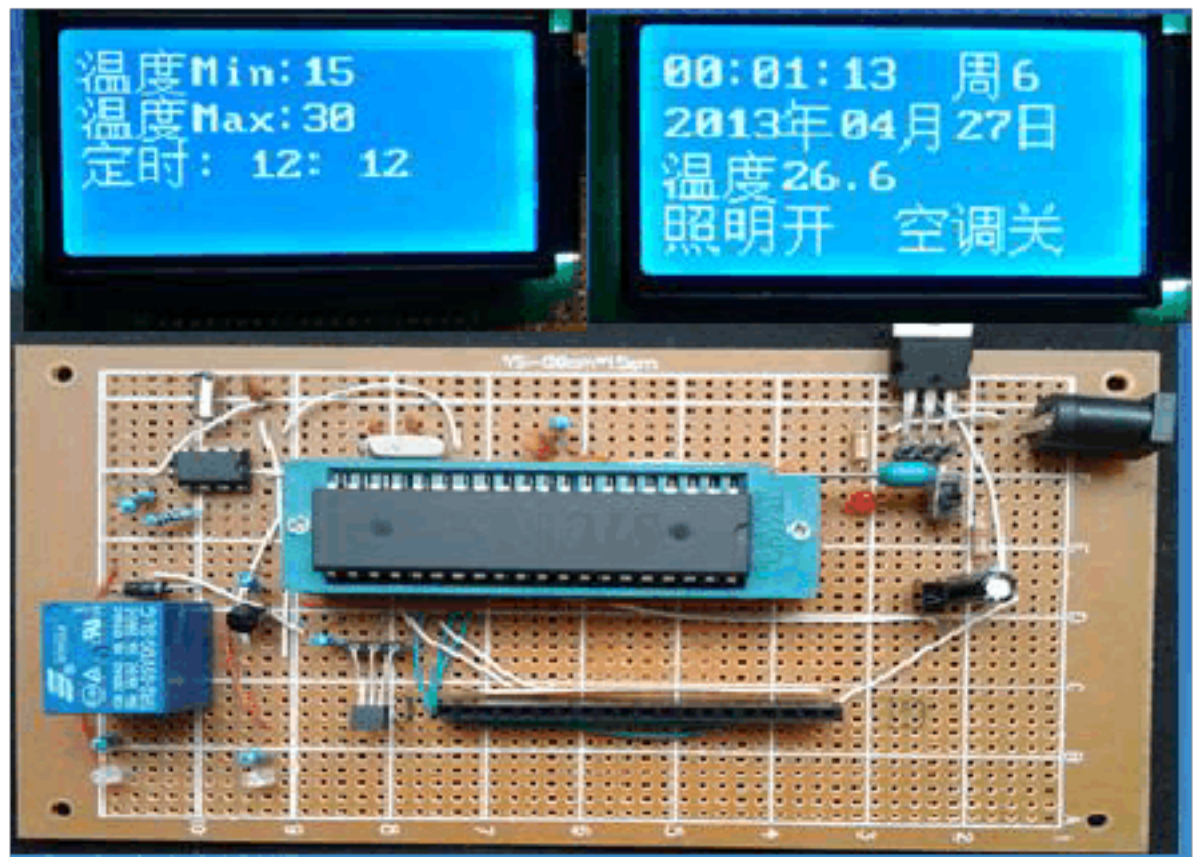


图 4-1实物电路及显示效果

4.2 总结

综合上面的介绍，本设计的硬件共有，电源模块，最小系统，红外模块，温度模块，时钟模块，液晶模块和继电器模块组成。软件部分共有，main.c（主函数模块文件），DS18B20.c（温度模块文件），DS1302.c（时钟模块文件），IR.c（红外模块文件），LCD12864.c（液晶模块文件），SET.c（设置功能文件）以及相应的头

文件组成。整个系统实现的功能有：

(1) 实时温度的测量和显示。

(2) 年月日时分秒时间信息的实时显示以及调整。

(3) 通过定时功能控制继电器开关灯，也可以随时遥控开关灯。

(4) 通过设定温度值利用继电器开关空调。

本设计还有以下有待升级之处：

(1) 可以加装移动电源，而不是通过电源适配器供电，这样可以使产品移动更方便，使用更加灵活。

(2) 由于液晶耗电量比较大，可以增加光敏电阻以及相应电路达到根据外接光线变化自动调节背光亮度的效果，以节省电能。

(3) 可以利用液晶的绘图功能为时钟模块增加一个图形时钟界面，这样可以使设计更加人性化。

参考文献

[1]桑胜举,沈丁.单片机原理及应用[M].北京：中国铁道出版社,20XX.8:11-20.

[2]凌志浩.AT89C52 单片机原理与接口技术[M]. 北京：高等教育出版社,20XX:8-21.

[3]郭循钊,邝帆,邵平,许宏科.[J].公路交通技术, 20XX:5-29.

[4]林凌,李刚,丁茹,李小霞.新型单片机接口器件与技术[M].西安:西安电子科技大学出版社,20XX:30-34.

[5]边春远,王志强.MCS-51 单片机应用开发实用子程序[M].人民邮电出版社,20XX:43-47.

[6]张友德,赵志英,徐时亮.单片微机原理应用与实验[M].复旦大学出版社,2000:32-34.

[7]李朝青.单片机原理及接口技术(简明修订版)[M].北京航空航天大学出版社,1998:23-43.

[8]黄河,李晓,刘学瑞,黄艳秋.基于 DS18B20 单总线数字温度计的设计与实现[J].中国矿业大学信息与电气工程学院.20XX.9.15:3-5.

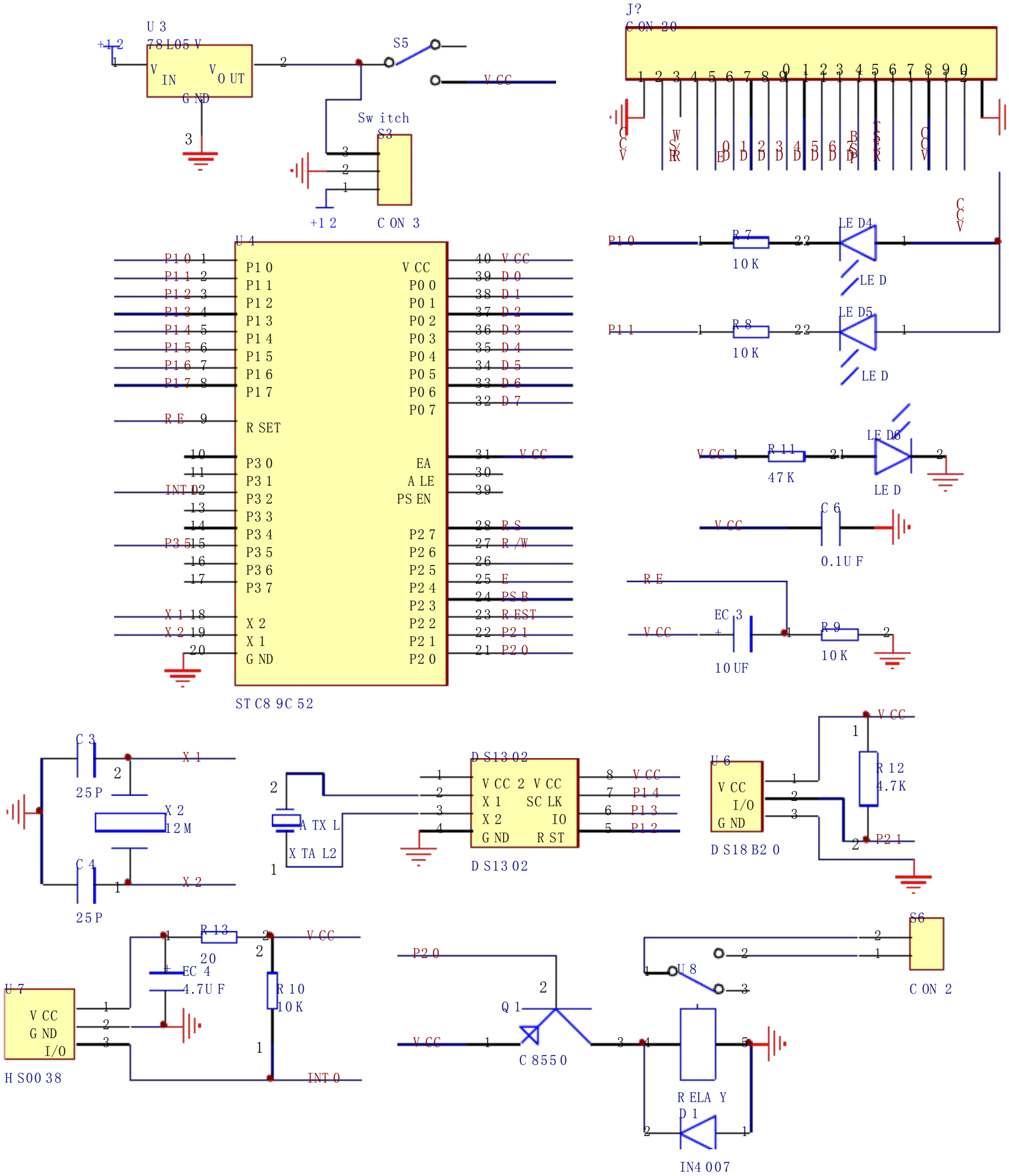
[9]郝文化.Protel Dxp 电路原理图与 PCB 设计[M].机械工业出版社.20XX.3:23-25.

[10]袁旭军,庄松林.单片机复位电路的可靠性分析[J].电子技术用.20XX.11:19-21.

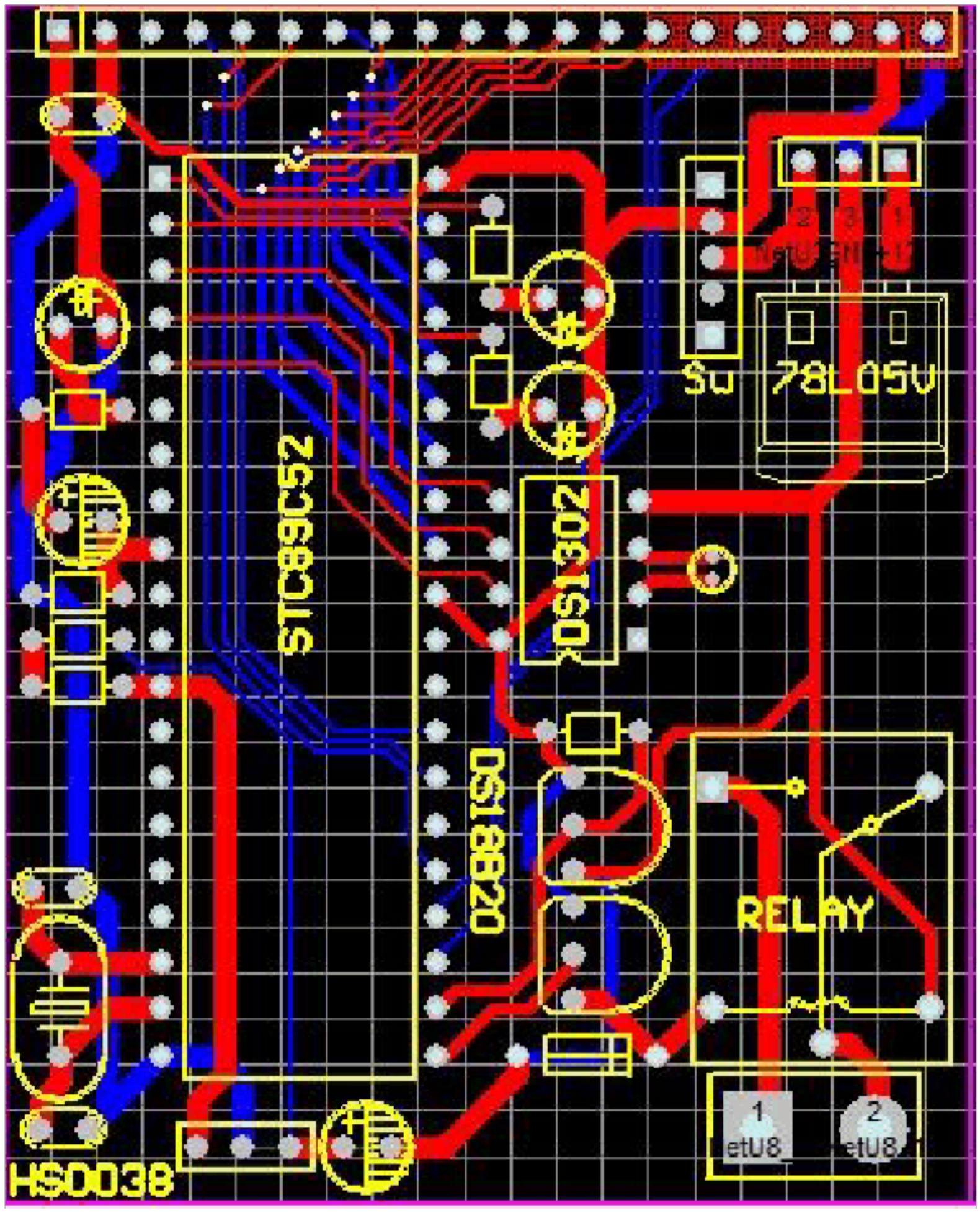
致 谢

本篇能够得以顺利完成，首先要感谢**老师对我的悉心指导。**老师在我着手做这个设计之初就给我分析了可能遇到的困难，这使我在做设计和撰写时更加有针对性，少走了不少弯路。宗栋老师带人宽厚做事认真，在这段时间里老师的教导使我明白了，做事要严谨细致，要有克服困难的耐心和勇气，当遇到技术瓶颈时没有这点素质是万万不行的，同时，还要宽厚带人，成功很大程度上来自他人直接或间接的帮助，只有与他人相处融洽我们才能得到别人更多的帮助才能更好地与他人合作。另外，在这里还要感谢一下从事编辑工作的我的舍友王坤同学，在编写过程中他教会了我不少使用 Office软件的小技巧，带着这些意外的收获对他表示衷心的感谢。

附录 1 整体电路原理图


U 3
78L05V
+12
VIN
VOUT
GND
S5
VCC
Switch
S3
CON 3
U 4
STC89C52
P10
P11
P12
P13
P14
P15
P16
P17
RE
RSET
P30
P31
INT0
P32
P33
P34
P35
P36
P37
X1
X2
VCC
EA
ALE
PSEN
P27
P26
P25
P24
P23
P22
P21
P20
RS
R/W
E
PSB
REST
P21
P20
J?
CON 20
LED D4
LED D5
LED D6
R7
R8
R11
10K
47K
LED D
C6
0.1UF
EC3
10UF
R9
C3
C4
25P
12M
DS1302
ATXL
XTAL2
R12
4.7K
DS18B20
R13
20
EC4
4.7UF
R10
10K
HS0038
S6
CON 2
U8
Q1
C8550
RELAY
D1
IN4007

附录 2 PCB 版图


STC89C52
DS1302
DS18B20
RELAY
78L05V
5v
HS0038
1
etU8
2
etU8

附录 3 Protel99SE生成的 3D 效果图

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：https://d.book118.com/035104301032012011