# 《考试报告》

目录

[1. 基本原理说明（附原理图及其他） 2](#_Toc18910)

[定时器中断基本原理说明 2](#_Toc24594)

[51单片机定时/计数器结构 2](#_Toc14012)

[工作方式寄存器TMOD 3](#_Toc19803)

[TCON 控制寄存器 3](#_Toc18010)

[定时器工作方式 3](#_Toc16053)

[代码 3](#_Toc7630)

[DS18B20温度传感器基本原理说明 4](#_Toc4464)

[电路原理图 4](#_Toc1907)

[温度寄存器格式 5](#_Toc26529)

[温度/数据关系 5](#_Toc8614)

[初始化时序方式 6](#_Toc16481)

[写时序 6](#_Toc10365)

[读时序 6](#_Toc25389)

[LCD1602液晶基本原理说明 7](#_Toc23110)

[原理图 7](#_Toc16112)

[引脚特性 7](#_Toc13507)

[指令介绍 8](#_Toc8901)

[读写时序 9](#_Toc31104)

[2. 程序设计原理（附流程图） 10](#_Toc9593)

[设计流程图 10](#_Toc6592)

[设计原理 10](#_Toc31304)

[硬件初始化 10](#_Toc28359)

[主循环 11](#_Toc9335)

[定时器中断服务 11](#_Toc2450)

[具体函数实现 11](#_Toc18607)

[总结 11](#_Toc32063)

[3. 程序代码及注释（main.c） 12](#_Toc4681)

[文字版 : 12](#_Toc25003)

[图片版 : 13](#_Toc5104)

[4. 运行效果截图 14](#_Toc30742)

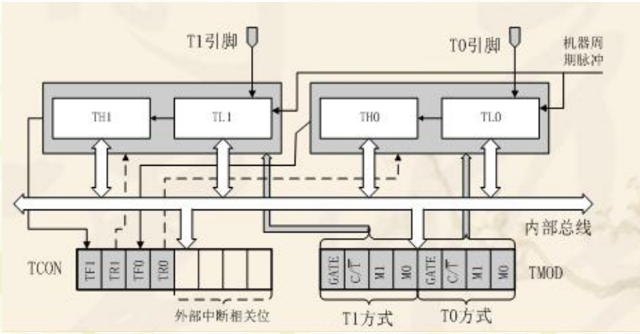
[编译运行 14](#_Toc1591)

[单片机显示效果 14](#_Toc27082)

# 基本原理说明（附原理图及其他）

## 定时器中断基本原理说明

### 51单片机定时/计数器结构



### 工作方式寄存器TMOD

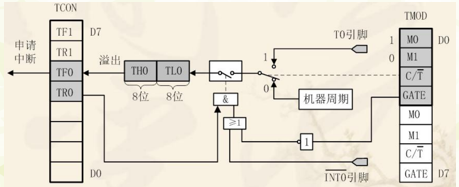


### TCON 控制寄存器



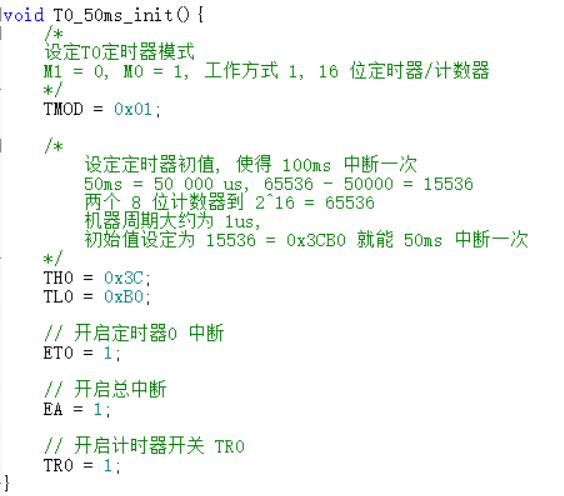
### 定时器工作方式

因为本程序中采用工作方式 1，所以只对工作方式 1 做介绍



## 代码

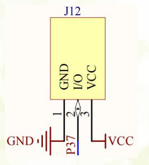
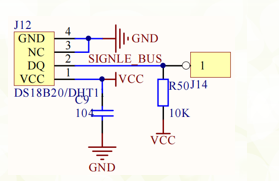
设定 T0 计时器 50 ms 中断



根据中断次数每 1s 读取一次温度值

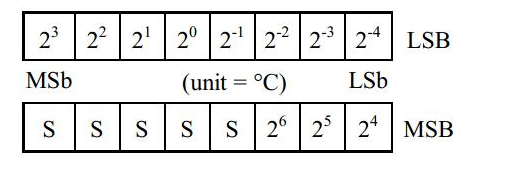
## DS18B20温度传感器基本原理说明

### 电路原理图



### 温度寄存器格式

DS18B20 可以实现最高 12 位温度存储，温度存储值以补码形式存储在寄存器中。



总共有2个字节，LSB是低字节，MSB是高字节。其中，MSb为字节的高位，LSb为字节的低位。对于二进制数，分别表示它们所代表的温度的含义。其中，S代表符号位，低11位均为2的幂，用于表示最终温度。

### 温度/数据关系

DS18B20的温度测量范围是-55度到+125度。温度数据的表现形式有正负温度之分。寄存器中的每个数字都像卡尺的刻度一样分布。

### 

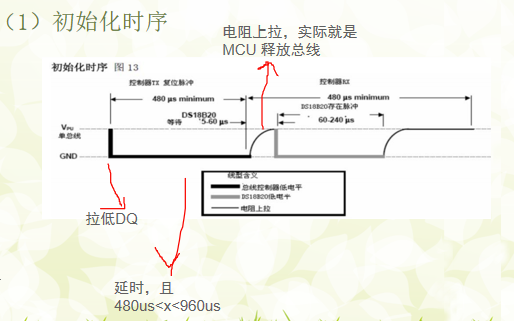
二进制数的最低位变1，表示温度变化0.0625度的映射关系。

**----** 当温度为0C时，对应的十六进制数为0x0000.

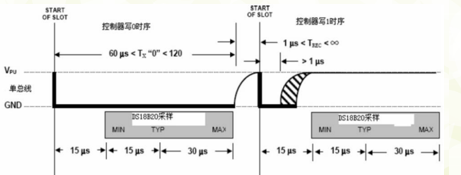
**----** 当温度为125“C时，对应的十六进制数为0xO7D0

**----** 当温度为-55°C时，对应的十六进制数为0xFC90。

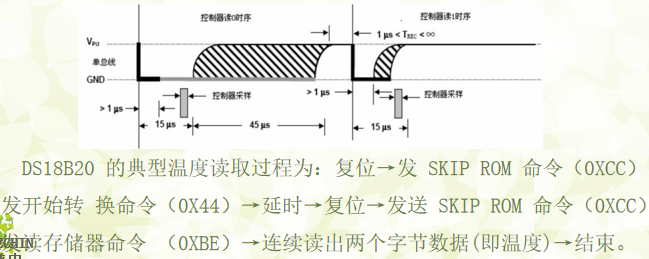
### 初始化时序方式



### 写时序

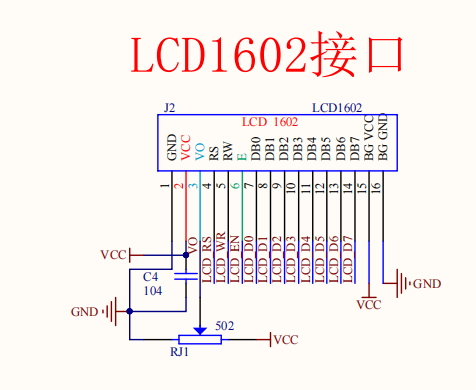


### 读时序



## LCD1602液晶基本原理说明

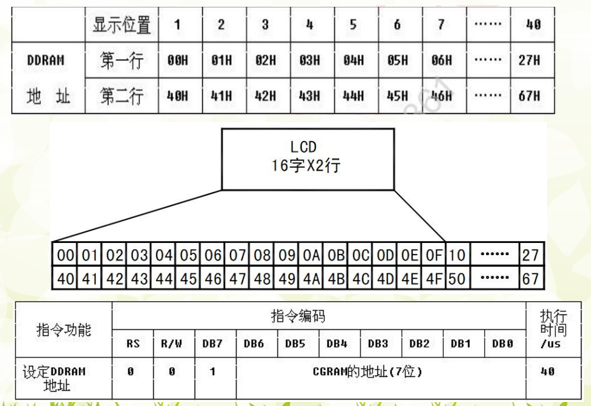
### 原理图



### 引脚特性



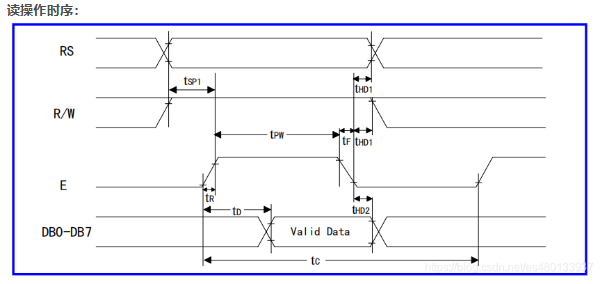
### 指令介绍

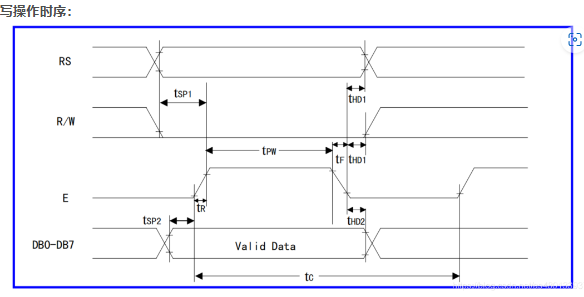






### 读写时序



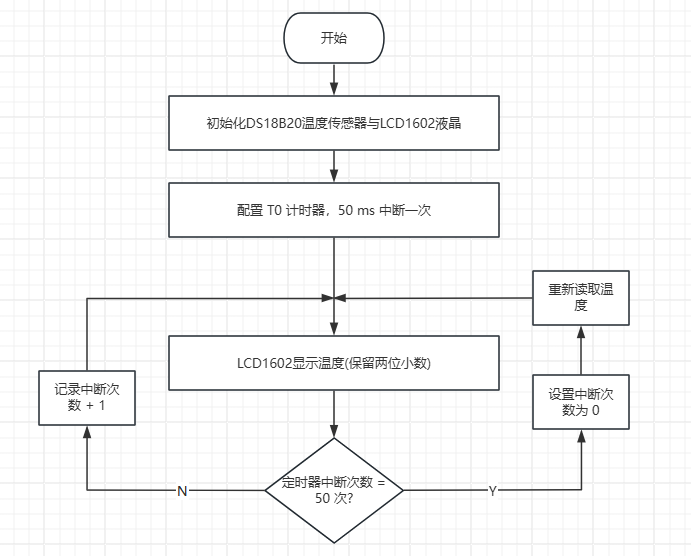


我们来分析一下时序图，当我们要写指令的时候，RS置为低电平，RW置为低电平，EN置为低电平，然后将指令数据送到数据口D0~D7，延时tsp1，让1602准备接收数据，这时候将EN拉高，产生一个上升沿，这时候指令就开始写入LCD，延时一段时间，将EN置低电平。

当我们要写数据的时候，RS置为高电平，RW置为低电平，EN置为低电平，然后将指令数据送到数据口D0~D7，延时tsp1，让1602准备接收数据，这时候将EN拉高，产生一个上升沿，这时候数据就开始写入LCD，延时一段时间，将EN置低电平。

# 程序设计原理（附流程图）

## 设计流程图



**使用 ProcessOn 绘制**

## 设计原理

使用DS18B20温度传感器读取温度数据，并通过配置定时器使得LCD 1602显示器每一秒刷新一次温度值。设计原理包括以下几个方面：

### 硬件初始化

在 main 函数中，首先初始化了三个硬件模块：

DS18B20温度传感器：调用 ds18b20\_init() 函数进行初始化。

LCD 1602显示器：调用 lcd1602\_init() 函数进行初始化。

定时器T0：调用 T0\_50ms\_init() 函数进行配置，使其每50毫秒产生一次中断。

### 主循环

主循环通过以下步骤更新LCD上的温度显示：

使用 lcd1602\_write\_cmd(0x80) 命令设置LCD光标到第一行的起始位置。

检查温度值 temp\_value 是否为负值，如果是负值，则将其转为正数并在LCD上显示负号。

依次输出字符 "TEMP:"，表示温度信息的标签。

将温度值 temp\_value 分解成各个位显示，其中温度值被放大了100倍以保留小数点后两位：

千位：temp\_value / 1000

百位：(temp\_value % 1000) / 100

小数点：显示 '.'

十位：(temp\_value % 100) / 10

个位：temp\_value % 10

显示摄氏度符号 'C'。

在第二行显示个人信息 "CREATED BY WPC"。

### 定时器中断服务

定时器T0的中断服务函数 T0\_isr 主要负责定时读取温度数据：

每50毫秒进入一次中断服务，使用计数器 counter 记录中断次数。

当计数器达到20（即经过1000毫秒或1秒）时，重置计数器并读取一次温度数据，更新全局变量 temp\_value。

读取的温度数据 ds18b20\_read\_temperture() 被放大100倍以保留两位小数，并存储在 temp\_value 中。

重新加载定时器的高8位和低8位以确保50毫秒的中断周期，使用值 0x3C 和 0xB0。

### 具体函数实现

ds18b20\_init(): 初始化DS18B20温度传感器。

lcd1602\_init(): 初始化LCD 1602显示器。

T0\_50ms\_init(): 配置定时器T0使其每50毫秒产生一次中断。

ds18b20\_read\_temperture(): 读取并返回DS18B20传感器的温度值。

### 总结

通过定时器每秒读取一次DS18B20温度传感器的数据，并通过LCD 1602显示实时温度。通过合理的硬件初始化、主循环中的数据处理以及定时器中断的使用，达到了对温度数据的准确读取和显示。

# 程序代码及注释（main.c）

## 文字版 :

#include "public.h" **// 多文件结构**

int temp\_value; // 全部变量记录温度

void main(){

ds18b20\_init();// 初始化DS18B20温度传感器

lcd1602\_init(); // LCD 1602 初始化

T0\_50ms\_init(); // T0 50 ms 中断配置

while(1){

lcd1602\_write\_cmd(0x80); // 第一行开始写

if(temp\_value<0)//处理负温度

{

temp\_value=-temp\_value; // 转化成正数

lcd1602\_write\_data('-'); // 输出 - 号

}

// 输出显示温度信息

// 输出显示 Temperature 的缩写 temp

lcd1602\_write\_data('T');

lcd1602\_write\_data('E');

lcd1602\_write\_data('M');

lcd1602\_write\_data('P');

lcd1602\_write\_data(':');

lcd1602\_write\_data('0' + temp\_value/1000); // 温度的十位

lcd1602\_write\_data('0' + temp\_value%1000/100); // 温度个位

lcd1602\_write\_data('.'); // 小数点

lcd1602\_write\_data('0' + temp\_value%100/10); // 保留小数点一位

lcd1602\_write\_data('0' + temp\_value%10); // 保留小数点两位

// \' 转义字符与下面的句子结合 输出 'C 表示摄氏度

lcd1602\_write\_data('\'');

lcd1602\_write\_data('C');

// 显示个人信息

lcd1602\_show\_string(0, 1, "CREATED BY WPC"); // 显示个人信息

}

}

// 定时器配置 : 1s 读取一次温度

void T0\_isr() interrupt 1{

static unsigned int counter = 0; // 记录中断次数

counter ++; // 中断次数 + 1

if(counter == 20){ // 50ms \* 20 = 1000 ms = 1 s

counter = 0; // 清空记录变量

temp\_value=ds18b20\_read\_temperture()\*100;// 全局变量 temp\_value 读取温度, 保留温度值小数后两位

}

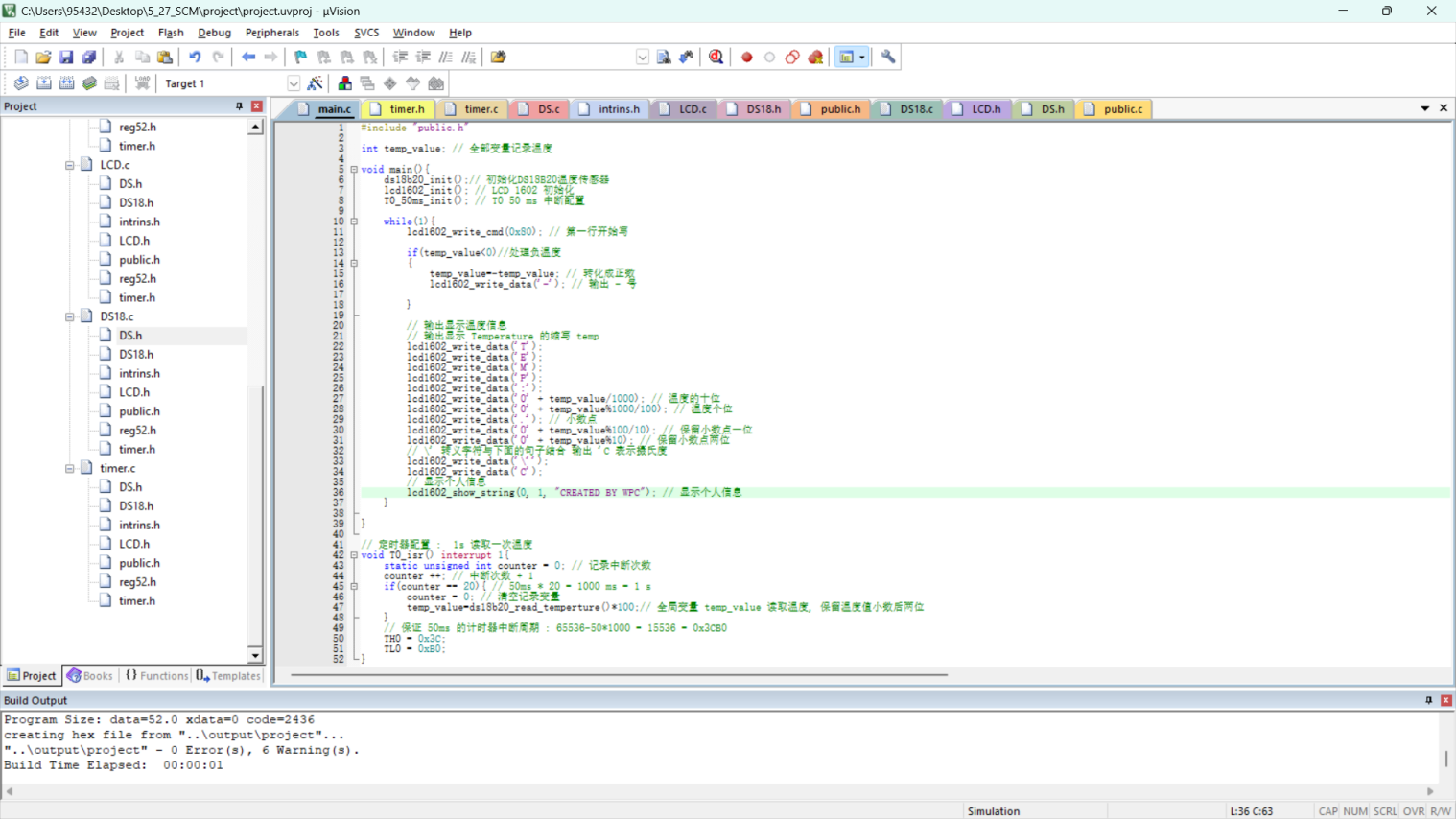
// 保证 50ms 的计时器中断周期 : 65536-50\*1000 = 15536 = 0x3CB0

TH0 = 0x3C;

TL0 = 0xB0;

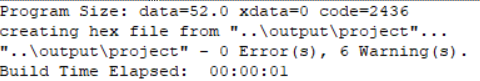
}

## 图片版 :



# 运行效果截图

## 编译运行



## 单片机显示效果

