1 课程设计目的

通过我们对 DSP控制器及其应用课程的学习和理解，综合运用课本中所学到的理论知识完成一个温度采集与显示的课程设计。通过这次实践锻炼我们查阅资料、方案比较、团结合作的能力。在这个过程我们必须掌握温度采集技术的硬件设计、熟悉 A/D转换技术和 DSP夜晶显示功能的软件设计。学会采用简单电路的实验调试和整机指标测试方法，增强我们的动手能力，为以后学习和工作 打下坚实基础

2 课程设计正文 2.1 系统分析

2.1.1 设计的任务及步骤

(1) 熟悉 MC1403芯片的应用；

(2) 根据实验测得热敏电阻和温度的一些数据，设计温度一一电阻公式；

设计外部硬件电路；

软件完成程序流程图设计和编程，其中包括 A/D转换和液晶显示部分；

编写上位机程序

(6) 软硬件联合调试；

(7) 书写设计说明书。

2.1.2 技术要求

此系统利用热敏电阻测得电阻一温度之间的关系，找到电阻和温度之间的代数关系，从而检测温度，设计硬件外扩电路，同时设计软件程序，包括 A/D程序设计，进行软硬件联系调试，能在液晶显示屏上显示温度。

2.1.3 设计思路

首先设计温度采集电路，由于考虑到使用的是非线性负温度系数的热敏电阻，因此采用了桥式电路尽量减小因外接不必要因素导致的误差，通过多次试验测得几个点，并拟合出一条合适的线性函数关系。其次进行软件设计，主要包括 AD转换模块、液晶显示模块、算法转换模块、主函数模块以及上位机模块。最后进行软硬件联系调试，并能在液晶上正常显示温度值。

2.2 总体设计 2.2.1 硬件设计

TMS320F2812作为本次课设使用的 DSP芯片。它包含 33 个电源引脚（为使器件正常运行，所有电源引脚必须正确连接且不能悬空）时钟源模块，DSP有六种信号可以使 DSP控制器复位，即电源复位、复位引脚~RS软件复位、非法地址复位、看门狗定时器溢出、欠压复位六种复位信号。

所以在设计的初期，把它分成了五个模块。其中复位采用电源复位的方式，由引脚 PCRESETI起。

为了可靠复位，其中低电平的有效时间至少 6 个 CPU时钟周期。DSP最小系统组成框图如图 1 所示。

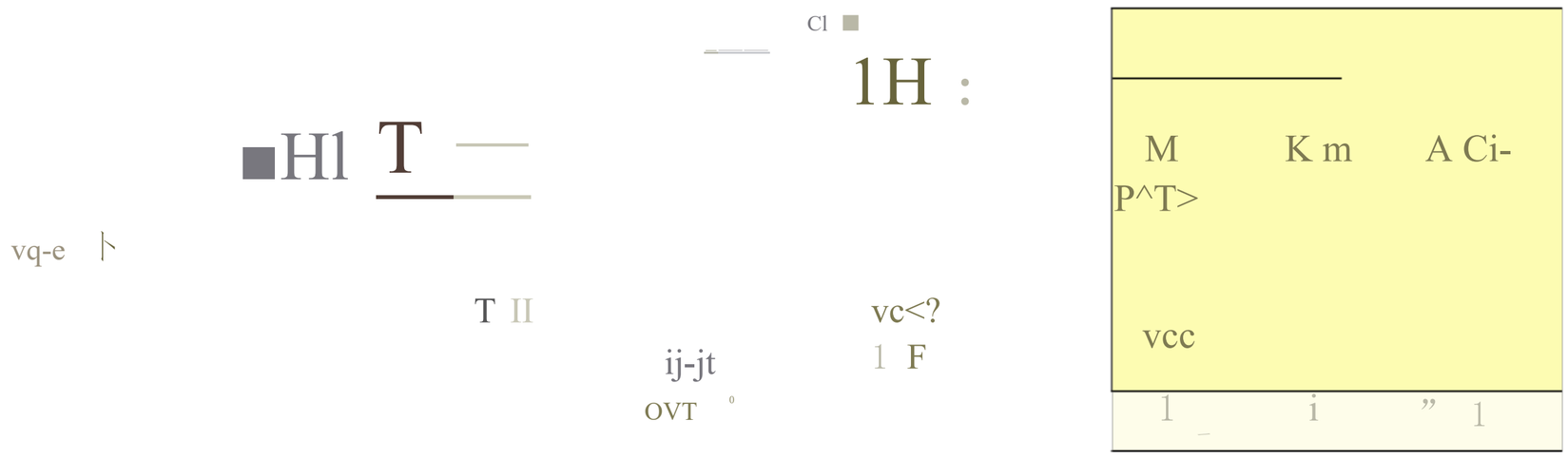

图 1 DSP 最小系统

其中：开关 SW-PB电容以及电阻构成一个简单的抚慰电路；晶振与两个并联电容接在 与 XTAL2为 XTAL1 2812 提供外部时钟；HT7133芯片将 5V直流电源转换为 3.3V 直流电源构成电源模块为 2812 供电。

222 系统框图设计

该系统包括温度采集模块、F2812 芯片、AD转换模块以及液晶显示模块，首先通过热敏电阻对外界温度的变化，从而改变其电位的变化，F2812 芯片等待中断并接收数据，然后送入 AD转换模块，AD对模拟信号进行转换，并将转换结果存放在结果寄存器的高 12 位，为保证数据精确多次采样求平均值，最后通过算法转换，将结果显示在 LCD液晶上。系统设计流程如图 2 所

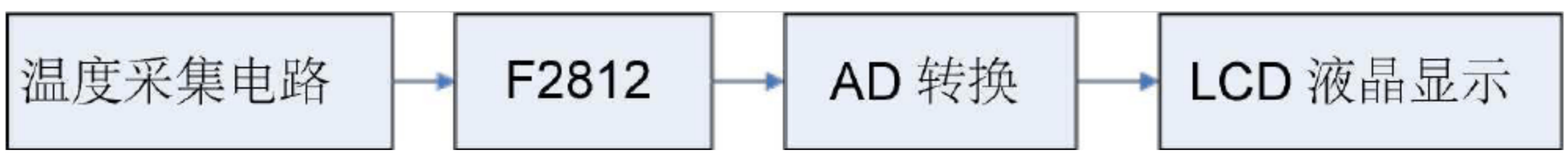

图 2 系统设计流程图

2.3 单元电路的设计

2.3.1 温度采集电路

考虑到此次使用的是非线性负温度系数的热敏电阻，因此采用桥式电路尽量确保其精度，并使用低压基准芯片 MC1403为电路提供基准电压，确保其尽量减少外界不必要因素的影响。温度采集电路如图 4 所示。

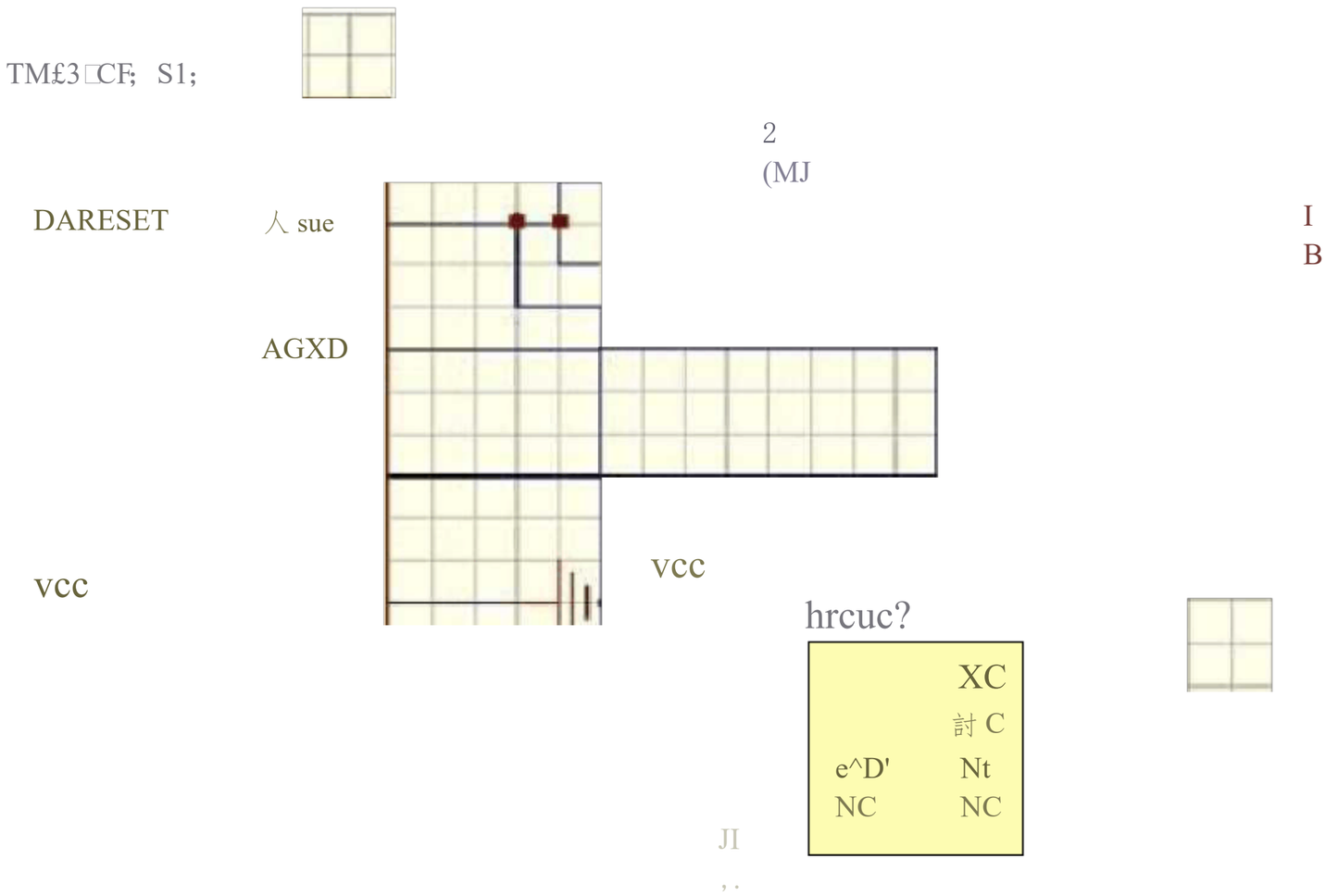

图 3 温度采集电路

2.3.2 单元模块的分析

（1） MC1403[氐压基准芯片引脚如图 4 所示。

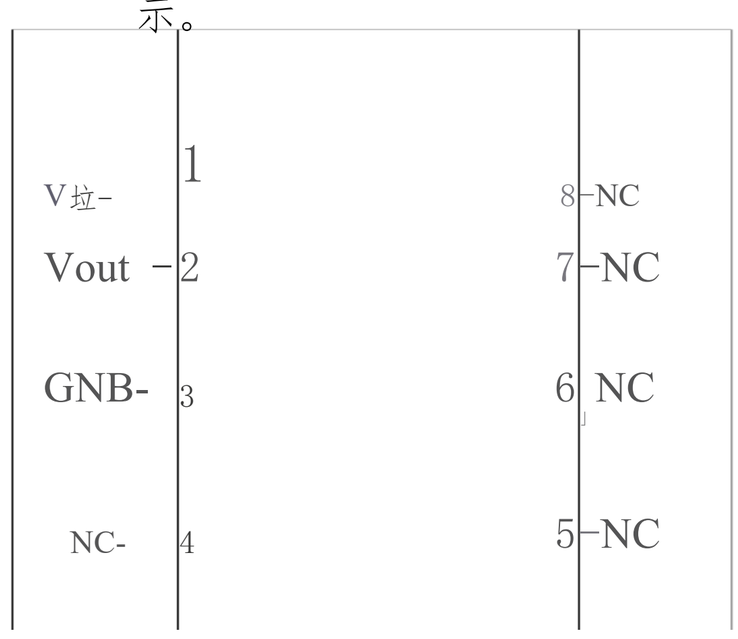

图 4 MC1403低压基准芯片引脚图

Vin 端口输入一个 4.5~40V 的模拟电压值（接 P2 口的 引脚），在 Vout 端输出了一个稳定的电压值（接 P2 口的 1 引脚，为 AD模块提供参考模拟高电平），GND端接模拟地（接 P2 口的 33 引脚）为 AD提供参考模拟低电平，并与数字地短接。

（2） AD转换模块

当模/数转换完成后，读取结果寄存器前，最好先读取模/数转换控制寄存器 ADCRL2勺 ADCRESULT 或 ADCRESULL 以确定当前结果寄存器的状态，保证读取的结果是正确。另外，要注意 12 位的转换结果放在结果寄存器中的高 12 位上,要进行位移转换，该 12 位数据与外部模拟输入电压的关系为：

12 位数字结果=4095* （输入电压/基准电压）

基准电压：3V

输入电压：A/D采集的电压

12 位数字结果：12 位的转换结果是放在结果寄存器的高 12 位上。

(3) LCD液晶显示模块

DSP经常会对读写周期较慢的输入/输出设备(如液晶显示模块、打印机、键盘等)进行访问，通常以下两种方法来解决 DSP与这些慢速设备之间的输入 /输出时序匹配问题。直接访问方式是将 DSP的读写信号限于慢速设备接口控制板引出的读写信号线直接相连，时序由 DSP内部读写逻辑控制。由于慢速外设的读写周期相对 DSP较慢，是两者的时序匹配，还必须进行一些时序方面的控制处理。一中处理方法是软件编程等待状态发生器，将外部总线周期扩展到数个机器周期。由于受硬件条件的限制，这种扩展通常也是有限的。

液晶显示为 128 行和 64 列的矩阵，它分左右两个平面，而一个字是 16 行和 16 列的矩阵，在按页的显示中每一页是一个 8 行和 128 列的矩阵，所以每一行字是分两页显示的

系统设计总电路图如图 5 所示。

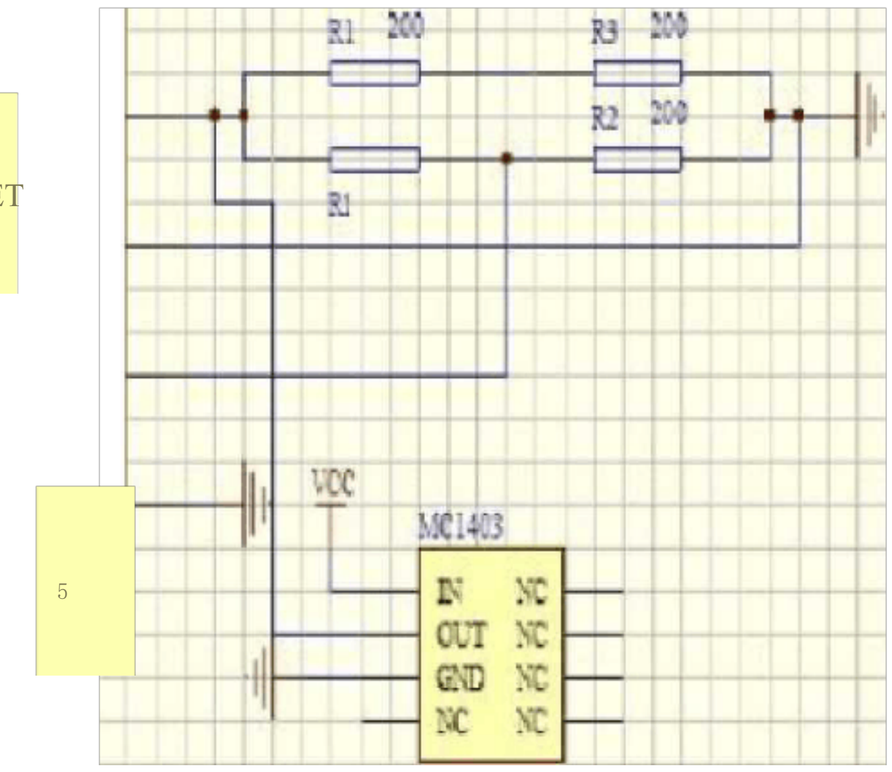

图.5 系统设计总电路图

(4) 单片机与 PC机串口通讯

DSP有一个全双工的串行通讯口，所以DSP和电脑之间可以方便地进行串口通讯。进行串行通讯时要满足一定的条件，比如电脑的串口是RS232电平的，而DSP的串口是TTL电平的，两者之间必须有一个电平转换电路，我们采用了专用芯片MAX232进行转换，虽然也可以用几个三极管进行模拟转换，但是还是用专用芯片更简单可靠。我们采用了三线制连接串口，也就是说和电脑的串口只连接其中的3根线：第5脚的GND第2脚的RXD第3脚的TXD这是最简单的连接方法，但是对我们来说已经足够使用了，电路如下图所示，MAX232的第10脚和DSP的11脚连接，第9脚和单片机的10脚连接,第15脚和单片机的20脚连接。全双工的串行通讯口原理图如图1所示:

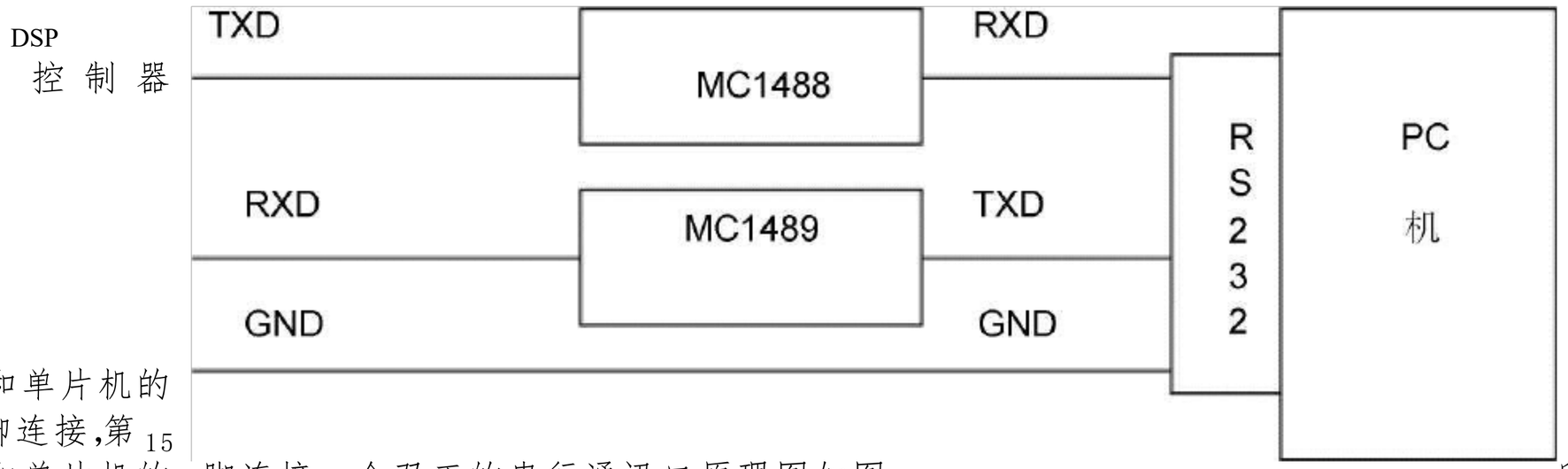

图1异步串行通讯原理图

DSP控制器串行通信接口（SCI）是一个标准的通用异步接收/发送（UART通信接口。它的接 收器和发送器都是双缓冲的。DSP控制器串行通信接口（SCI）有自已的使能和中断位，它们可以半双工或全双工方式工作。为了保证数据的完整性，串行通信接口对接收的数据进行间断检测、奇 偶性、超时和帧错误的检查。位速率（波特率）可通过一个16位的比特率选择寄存器进行编程，因此，可以有超过65000种不同的速率。

SCI模块是一个8位片内外设，通过DSP的16位外部数据总线的低8位与外部设备通信。SCI的物理描述：

（1） 两个I/O引脚。

*SCIRXD（SCI接收数据输入）

*SCITXD（SCI发送数据输出）

（2）

通过一个16位波特率选择寄存器的编程，可得到超过65000种不同的可编程位速率。

1~8的可编程数据字长。

长度为1位或2位的可编程停止位。

内部产生的串行时钟。 四个错误检测标志。

（3）

（4）

（5）

（6）

*奇偶性错误 * 超限错误

*帧错误 * 间断检验

（7） 两种唤醒多处理器模式，可与任意通信格式一起使用。

*空闲线路唤醒 * 位寻址唤醒

（8） 半双工或全双工操作。

（9） 双缓冲接收和发送功能。

（10） 接收和发送操作均可通过中断或查询操作进行，相应的状态标志如下：

发送器：TXRDY标志（发送器缓冲寄存器准备好接收另一个字符）和TXEMPTY标志（发送移位寄存器空）。

接收器：RXRDY标志（准备从接收缓冲寄存器中读出一个新字符）、BRKDT标志（间断条件发生）

和 RXERRQR 可用来监视四个中断条件。

（11） 发送器和接收器中断的分立的使能位（间断除 。
外）

NRZ(非归零）格式的。

（12） SCI 数据，不管是接收的、还是发送的，都是

2.4 软件程序设计

241 软件系统流程图

软件系统流程如图 6 所示。

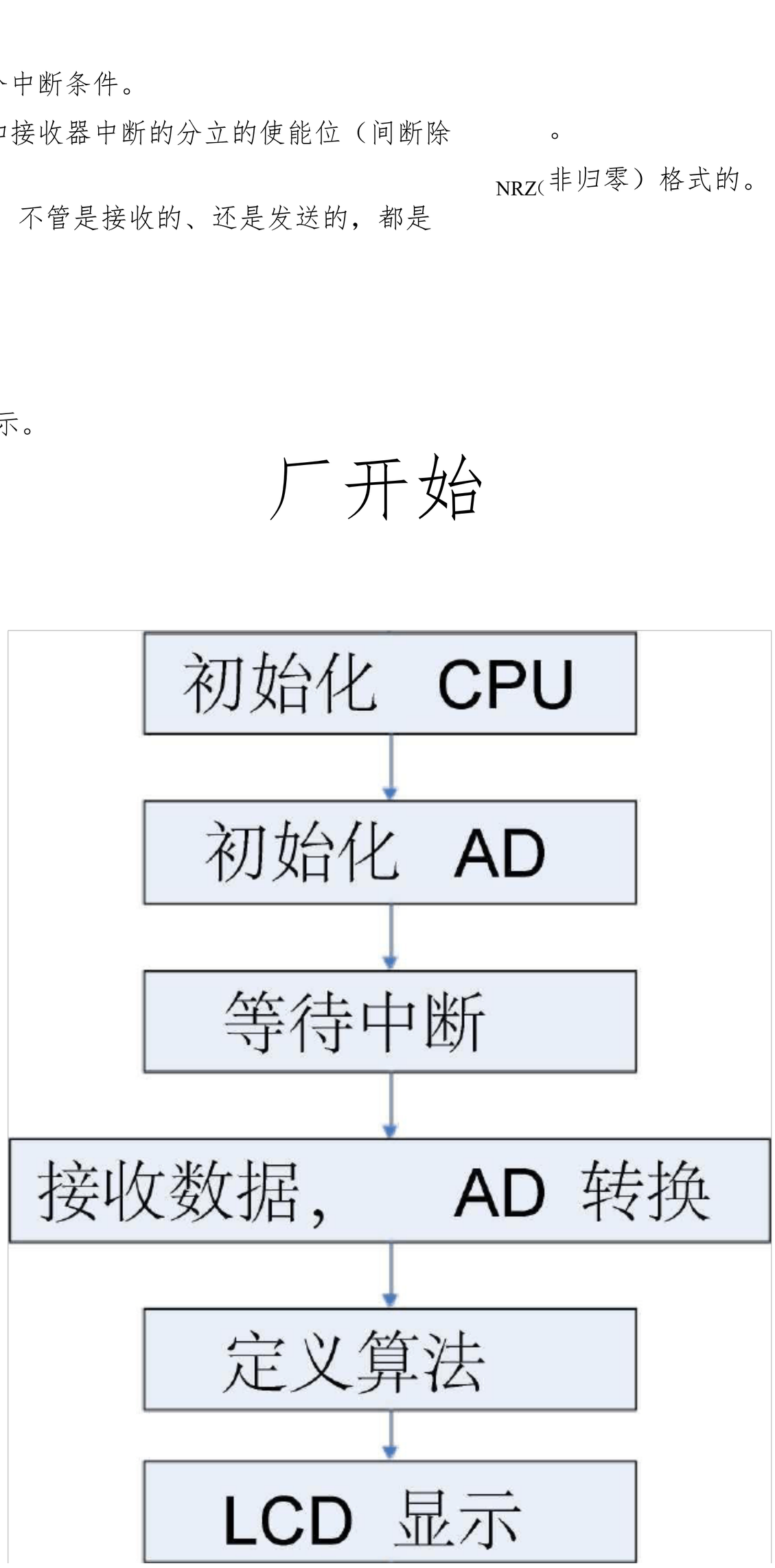

图 6 软件系统流程图

2.4.2 软件程序设计

程序主要包括主程序、中断、延时、 A/D 转换、算法转换、液晶显示等子程序，实现温度采集与显示的功能，程序详见附录。此次课设我主要负责 AD 转换、算法转换和主程序部分。

AD 转换部分我采用单通道采样方式对温度采集电路发送的值进行转换，并使用了第一组的号中断对电路进行定时采样， 由于考虑到热敏电阻比较灵敏， 我采用了多次采样求平均值的方法确 保数据精确。

算法转换部分主要是对 AD 采集并转换的值进行进一步的转换，主要是将桥式电路的电压与热

敏电阻的关系和拟合的温度与电阻的关系整合在一起，使其转换为我们能看懂的温度，并将其一位一位拆分开来复制到 LCD液晶显示子程序，将结果显示字液晶屏上。

(1) NTC热敏电阻原理及应用

NTC热敏电阻器在室温下的变化范围在 100~1000000 欧姆，温度系数-2%~-6.5%o NTC热敏电阻器可广泛应用于温度测量、温度补偿、抑制浪涌电流等场合。

NTC热敏电阻为负温度系数热敏电阻。

零功率电阻值 RT (Q)：RT指在规定温度 T时，采用引起电阻值变化相对于总的测量误差来说可以忽略不计的测量功率测得的电阻值。

电阻值和温度变化的关系式为:

$$RT = RN \exp B(1/T - 1/TN)$$

RT：在温度 T (K)时的 NTC热敏电阻阻值。

RN：在额定温度 TN（K）时的 NTC热敏电阻阻值。

规定温度（ K）。

NTC热敏电阻的材料常数，又叫热敏指数。

exp ：以自然数 e 为底的指数（e = 2.71828 …）o

该关系式是经验公式，只在额定温度 TN或额定电阻阻值 RN的有限范围内才具有一定的精确度，因为材料常数 B本身也是温度 T的函数。

额定零功率电阻值 R25 (Q)：根据国标规定，额定零功率电阻值是 NTC热敏电阻在基准温度 25 C 时测得的电阻值 R25，这个电阻值就是 NTC热敏电阻的标称电阻值。通常所说 NTC热敏电阻多少阻值，亦指该值。

材料常数(热敏指数)

$$B = \frac{T_i T_i}{T_3 - T_1}$$

RT1 ：温度 T1（K）时的零功率电阻值。

RT2 ：温度 T2（K）时的零功率电阻值。

T1， T2：两个被指定的温度(K)o

对于常用的 NTC热敏电阻，B值范围一般在 2000K □6000K 之间。已知 150Q 时为 25 C, 即 R25 = 150 Q, 120Q 时为 45C，经计算，B约为 2121.7K。代入得:

r=150*ex p(2121.7*((1/t)-1/298))

取曲线上的任意几点，再用 Matlab 进行曲线拟合，得到几条直线，经筛选后得到如图 7 的图形。经变换单位后，得到温度与电阻值之间的关系式:

T = 125 ¯ 2/3*R

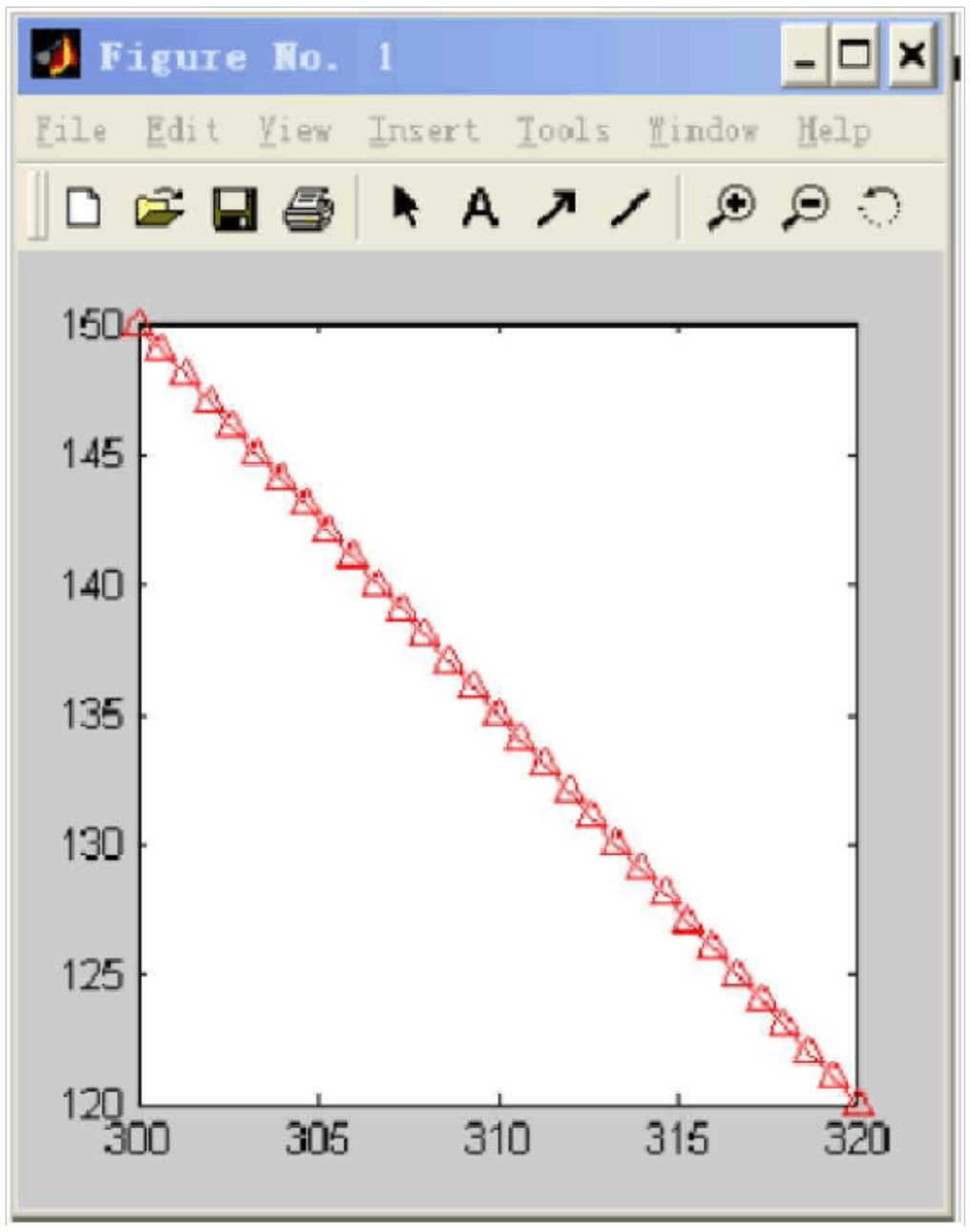

图 7 拟合后的直线

其中横坐标为温度，单位为 K；纵坐标为电阻值。

由硬件部分(电桥电路)得到电压与电阻值之间的关系：V = 2.5*R1/(R1+R), 其中 R1=200Q,

与上式相结合并经调试修改后，得到温度值与电压之间的关系：

T=140.0-0.67*(500.0/V-200.0)

//温度计算子程序

```
void Wen du(float c)

    temp erature=140.0-0.67*(500.0/c-200.0);

    T_te mp e=(i nt)te mp erature;

    T_te mp e1=T_te mp e/10;

    T_tem pe0=T_tem pe%10;

    D_te mp e=(i nt)(10*(te mp erature-T_te mp e));
```

(2) 上位机

SCI 波特率的设置：

16 位的波特率数值用 BRF表示。

使用 16 位的波特率选择寄存器值来选择 64K种不同的串行时钟频率中的一种。

选择可以通过以下方法来确定:

当 BRR =1~65535时（BRR波特率选择寄存器中的 16 位值）：

$$\text{SCI 异步波特率} = \frac{\text{SYSCLK}}{\text{RR}}$$

波特率是按位计算的，而每位需要 8 个 SCICLK周期。

当 BRR=0时

$$\text{SCI 异步波特率} = \frac{\text{SYSCLK}}{}$$

串行通信过程:

① 在进行串行通信之前，先对 SCI 通信控制寄存器 SCICCR进行设置。

信息块的起始位始终是 1 位，停止位可选择 1 位或 2 位。

数据位的长度是可编程的，可设置 1~8 位，由 SCICCR的 SCICHAR2 0设置。数据位长度小于 8 位时，写到发送缓冲寄存器 SCITXBUF的数据只有低 L 位被发送（这些数据位必须右对齐，因为对于少于 8 位的字符，左边的位是被忽略的）。

校验位可设，由 SCICCR的 PARITY ENABLE（ 5 位）设置。校验方式有两种，奇、偶校验，由 SCICCR的 PARITY（ 6 位）设置。

空闲线方式与地址位方式的选择，由 SCICCR的 ADDR/IDLE MOD 3 位）设置。

SCICCR的 SCIENA(4 位）是一个总开关位。

②在信息块的格式确定后就要确定串行通信的波特率。

串行通信的时钟 SCICLK是由时钟源模块的 SYSCLK按照 SCI 波特率寄存器规定的分频系数分频后得到的。

16 位的波特率寄存器 SCIHBAUD和 SCILBAUD分别决定了波特率寄存器的高、低 8 位，二者共同构成了 16 位的波特率数值。

③ 设置好串行通信的波特率后，就要确定接收、发送部分的控制位，这些由 SCI 控制寄存器 SCICTL1、2 来完成，

④ 当数据写入 SCITXBUF寸，SCICTL2的 TXRDY（ 7 位）被清 0 ；当 SCITXBUF的数据全部移出 后，贝 U SCICTL2的 TXRDY被置 1，并且产生发送中断 TXINTP

（3）SCI 下位机程序

```
#inelude ”  DSP281x_Device.h" //  使用的函数原型声明  void Gpi o_select(void);
void Sp eedU pRevA(void);
void SCI_I nit(void);
   in terru pt void SCI_TX_isr(void); in
          terru pt void SCI_RX_isr(void);
//全局变量
char message[]=te mp erature; void main (void)
```

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/028117023127007010