

# 科技与艺术学院

## 单片机课程设计报告

项目名称： 电阻测量系统

系： 机电系

专 业： 17 电子信息工程 1

学生姓名： 江瀚林

学号： Xb17610107

## 目录

目录.....	I
图目录.....	II
表目录.....	III
摘要.....	1
Abstract .....	2
电阻测量系统设计.....	3
1 设计内容和要求 .....	3
2 系统总体结构 .....	3
3 硬件设计 .....	4
3.1 关键电子元件挑选.....	4
3.2 电路设计，基于 Protel 99 SE .....	6
3.3 仿真过程，基于 Proteus 8 Professional.....	10
4 软件设计 .....	12
5 电阻测量系统调试.....	15
6 遇到的问题与解决方法 .....	20
7 心得体会 .....	22
附件.....	24

## 图目录

图 2-1 系统总体结构 .....	3
图 3-1 OP07 俯视图 .....	4
图 3-2 OP07 内部结构透视 .....	4
图 3-3 元件图 .....	5
图 3-4 TL431 的实物图 .....	5
图 3-5 TL431 的使用方法 .....	5
图 3-6 恒流源电路的原理 .....	6
图 3-7 实际电路原理图 .....	6
图 3-8 差分比例放大原理 .....	7
图 3-9 差分比例放大电路 .....	8
图 3-10 ADC10 总体结构框图 .....	9
图 3-11 LCD 控制的信号控制图 .....	9
图 3-12 TAC6416A 的 IO .....	10
图 3-13 LCD 原理图 .....	10
图 3-14 同向放大电路实现恒流源 .....	11
图 3-15 差分比例放大电路 .....	11
图 3-1 主程序流程图 .....	12
图 4-2 通过 Grace 配置 ADC10 .....	12
图 4-3 .....	13
图 4-4 LCD_Display()流程图 .....	14
图 5-1 被测电阻与 LCD 显示电压的线性图像 .....	16
图 5-2 被测电阻收纳盒 .....	17
图 5-3 测量 120 $\Omega$ .....	17
图 5-4 测量 110 $\Omega$ .....	17
图 5-5 测量 470 $\Omega$ .....	18
图 5-6 测量 560 $\Omega$ .....	18
图 6-1 加入滤波电容 .....	21

## 表目录

表 1	电阻与电压数据.....	16
表 2	标称值、LCD 显示的电阻值、欧姆表测得值.....	19
表 3	焊接元件清单.....	29

## 摘要

电阻是任何电路中不可缺少的元件，它的作用很多，可以分压限流，可以进行能量转化，可以应用于传感，电阻阻值的大小直接关系到电路的性能。例如输入电阻是用来衡量放大器对信号源的影响的一个性能指标。输入电阻越大，表明放大器从信号源取的电流越小，放大器输入端得到的信号电压也越大，即信号源电压衰减的少。同时，电阻是产生热损耗与热噪声的重要原因，它的阻值大小直接决定了电路的好坏，因此围绕电阻测量产生了大量的测量方法。

常见的测量方法有伏安法，半偏法，电桥法等等，这些都是基本的方法，但普遍精度不高。当前范围内有许多种精确测量电阻的方法，比如对于低值电阻，有采用四线制电流倒向技术测量的方法，高值电阻而言，也有兆欧的欧姆表用于测量。本次探究是基于单片机 MSP430G2553 的电阻测量，用数字化的方法来实现对模拟电路值的间接测量计算得到我们要测量的电阻，可以很大程度上提高精度，并且方便简单。数字化测量方法是其一大特点，对于这一方法的探究很有价值，并且它拥有广阔应用前景。电路的相关原理会在本文中具体阐释，实验当中也不可避免的会遇见一些问题，本人也对这些问题进行了探究。

**关键词：**MSP430 电阻测量 单片机 恒流源 放大器

## Abstract

Resistor is an indispensable component in any circuit. It has many functions. It can be divided and limited, and energy conversion can be performed. It can be applied to sensing. The resistance value is directly related to the performance of the circuit. For example, an input resistor is a performance metric used to measure the effect of an amplifier on a signal source. The larger the input resistance, the smaller the current drawn by the amplifier from the source, and the greater the signal voltage at the input of the amplifier, ie, less attenuation of the source voltage. At the same time, the resistance is an important cause of heat loss and thermal noise. Its resistance directly determines the quality of the circuit. Therefore, a large number of measurement methods are generated around the resistance measurement.

Common measurement methods are voltammetry, semi-bias method, bridge method, etc. These are the basic methods, but the general accuracy is not high. There are many methods for accurately measuring resistance in the current range. For example, for low-value resistors, there are four-wire current reversal techniques. For high-value resistors, there are also megohmmeters for measurement. This inquiry is based on the resistance measurement of the single-chip MSP430G2553. The digital method is used to realize the indirect measurement of the analog circuit value. The resistance we want to measure can be greatly improved, and it is convenient and simple. The digital measurement method is a major feature, and it is valuable for the exploration of this method, and it has broad application prospects. The relevant principles of the circuit will be explained in detail in this article. In the experiment, some problems will inevitably be encountered. I also explored these issues.

**Keywords:** MSP430    resistance measurement    single chip    constant current source    amplifier

## 电阻测量系统设计

### 1 设计内容和要求

运用单片机 MSP430 进行数字化电阻测量这一方法非常方便,有广阔应用前景和探究价值,是未来的趋势。本次电阻测量系统设计的目的需要①掌握恒流源电路(例如:OP07)及放大电路的设计方法。②掌握 MSP430 单片机中 ADC 模块的特性及使用。③掌握 I2C 总线的原理及应用。④掌握 CCSv5 软件的应用。⑤掌握 C 语言编程。

要求实现的功能为用外部 PCB 电路板配合 MSP430 单片机构成电阻测量系统,实现对给定 0~1K 电阻的测量,用 MSP430 上的 LCD 段式液晶屏显示测量结果。

### 2 系统总体结构

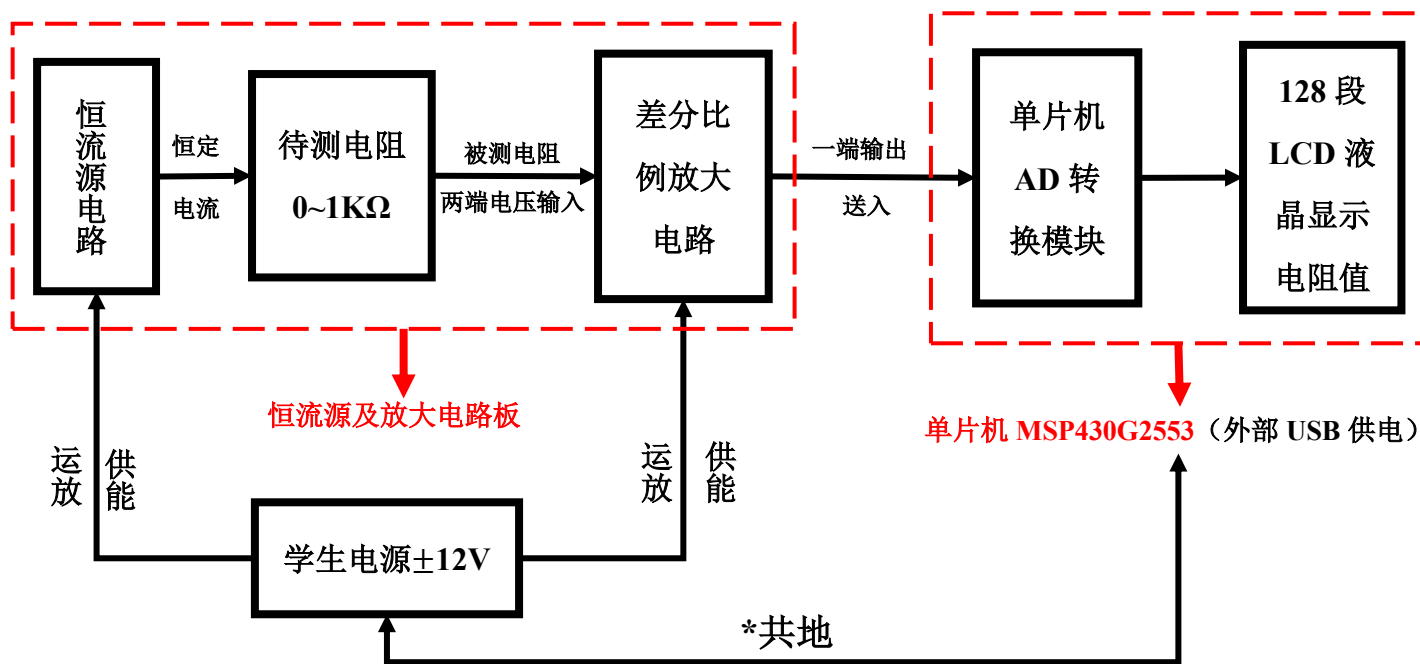


图 2-1 系统总体结构

如图 2-1 所示,本次电阻测量系统由恒流源及放大电路板和 MSP430G2553 构成。电路板由自己设计制作并由学生电源 $\pm 12\text{V}$  供能,恒流源模块输出恒定的电流,不受被测电阻阻值的影响。电流源恒定,被测电阻阻值的变化引起电阻两

端压降的线性变化。将被测电阻两端的电压分两路输入送给后一级的差分比例放大电路，最终为单一输出，输出的电压已经过前级差分比例运算电路处理，得到的就是被测电阻两端的压降。将电路板的电压输出端送入单片机 MSP430G2553 的 AD 转换输入引脚，将模拟信号转换为数字信号，处理并显示在 128 段 LCD 液晶显示屏上，即为被测电阻的阻值。

### 3 硬件设计

本次电阻测量系统的硬件设计主要是在“恒流源及放大电路板”上，本人将从电子元件的挑选，基于 Protel 99 SE 的原理图，基于 Proteus 8 Professional 仿真来阐述。

#### 3.1 关键电子元件挑选

##### (一) OP07 芯片

OP07 芯片是一种低噪声，非斩波稳零的双极性运算放大器集成电路。由于 OP07 具有非常低的输入失调电压（对于 OP07A 最大为 25HV），所以 OP07 在很多应用场合不需要额外的调零措施。OP07 同时具有输入偏置电流低（OP07A 为  $\pm 2\text{ns}$ ）和开环增益高（对于 OP07A 为 3007/mV）的特点，这种低失调、高开环增益的特性使得 OP07 特别适用于高增益的测量设备和放大传感器的微弱信号等方面。

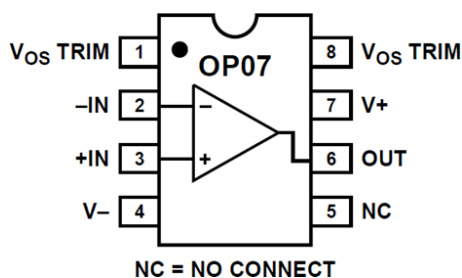


图 3-2 OP07 内部结构透视

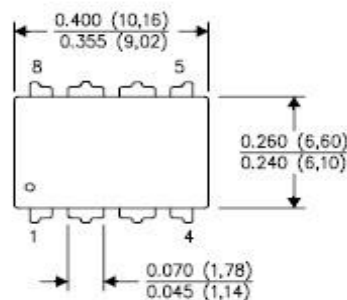


图 3-1 OP07 俯视图

OP07 芯片引脚功能说明：1 和 8 为偏置平衡(调零端)，2 为反向输入端，3 为正向输入端，4 接负电源，5 空脚 6 为输出，7 接正电源。

OP07 不管是简单的电路还是复杂的电路都会出现它的身影，它是运算放大的代表元件之一。此次电阻测量系统恒流源模块需要一颗 OP07 芯片配合电阻构



成同相比比例运算电路。放大电路模块需要另外一颗 OP07 芯片配合电阻构成差分比例运算电路获得被测电阻两端的压降，没有运放芯片无法完成期望的功能。

## (二) TL431

TL431 是可控精密稳压源。它的输出电压用两个电阻就可以任意的设置到从  $V_{ref}$  (2.5V) 到 36V 范围内的任何值。该器件的典型动态阻抗为  $0.2\Omega$ ，在很多应用中用它代替稳压二极管，例如，数字电压表，运放电路，可调压电源，开关电源等。

以下是分别是 TL431 的实物草图、软件中的元件图。



图 3-4 TL431 的实物图

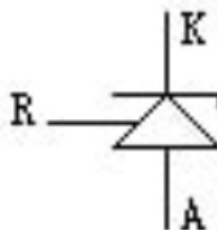


图 3-3 元件图

TL431 可等效为一只稳压二极管，其基本连接方法如下图所示。下图 a 可作 2.5V 基准源，下图 b 作可调基准源，电阻  $R_2$  和  $R_3$  与输出电压的关系为  $U_0 = (1 + R_2/R_3)2.5V$

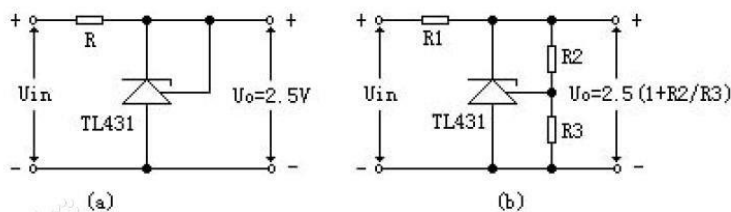


图 3-5 TL431 的使用方法

本次恒流源模块中 TL431 加在由 OP07 配合电阻构成同向比例放大电路的同相输入端，使得由学生电源输入的 12V 稳定在 2.5V。

### 3.2 电路设计，基于 Protel 99 SE

#### (一)恒流源

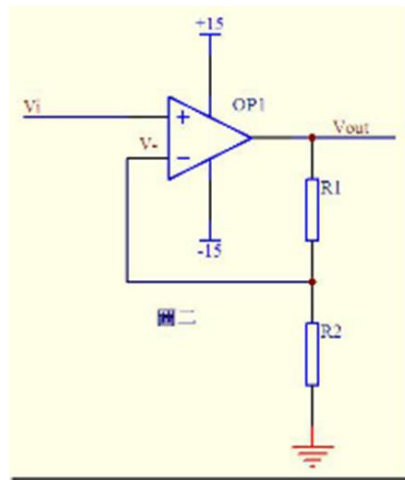


图 3-6 恒流源电路的原理

如上图 3-6 所示  $V_i$  与  $V_-$  虚短，则  $V_i = V_-$ ， $R_1$  为被测电阻， $R_2$  为定值电阻。

因为虚断，反向输入端没有电流输入输出，通过  $R_1$  和  $R_2$  的电流相等，设此电流为  $I$ ，由欧姆定律得： $I = V_{out}/(R_1+R_2)$

$V_i$  等于  $R_2$  上的分压，即：

$V_i = I \cdot R_2$        $I = V_i / R_2$ ，实现恒流，与  $R_1$  无关

由上述各式得  $V_{out} = V_i \cdot (R_1 + R_2) / R_2$

这就是传说中的同向放大器的公式了，应用在此次电阻测量系统中如下图 3-7 所示。

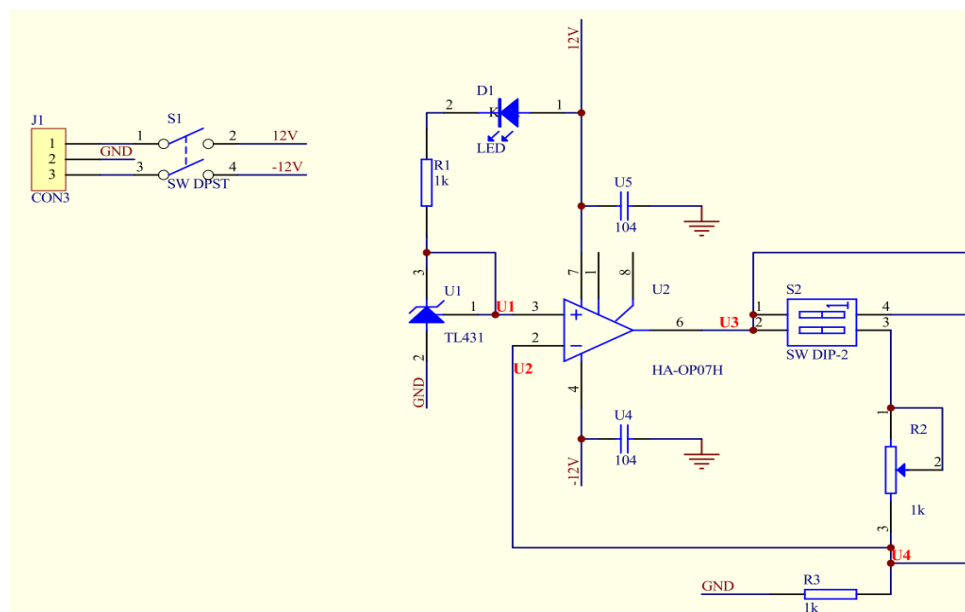


图 3-7 实际电路原理图

图 3-7 为实际恒流源模块电路图，电源输入端加入双刀开关可随时断电处理，引入 D1 二极管的作用是电源上电时亮起与人的视觉交互知道模块通电正常。TL431 将输入电压稳定在 2.5V，即  $U_1=2.5V$ 。由上文原理可得  $U_4=U_2=U_1=2.5V$ ，流过滑动变阻器  $R_2$  和定值电阻  $R_3$  的电流为定值  $I=2.5/1000=2.5mA$ ，可推得  $U_3=2.5+2.5\times 10^{-3}\times(\Delta R_2)$ ，所以  $(U_3)_{\max}=2.5+2.5\times 10^{-3}\times 1000=5V$ ，滑动变阻器  $R_2$  两端的电压变化范围在  $\Delta U_{R_2}=0\sim 2.5V$ 。恒流源功能说明到此结束。

## (二)差分比例运算电路

由于要利用电流源恒定，电压随被测电阻线性增大的规律，提取被测电阻两端的压降最为方便，但是需要引出两端才可得到压降。回忆之前模拟电子技术学过的运算电路中，确实有两端输入的放大电路结构，它就是差分比例运算电路。以下是具体原理。

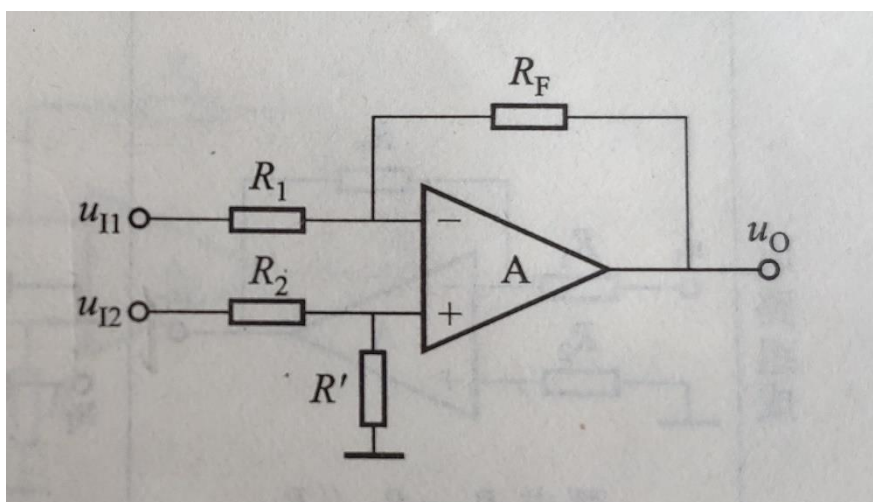


图 3-8 差分比例放大原理

如图 3-8 所示  $U_{I1}$  和  $U_{I2}$  为两个输入电压，当满足条件  $R_1=R_2$ ， $R_F=R'$  时，

$$U_0 = -\frac{R_F}{R_1}(U_{I1} - U_{I2})$$

以下为电阻测量系统工程实际使用时的差分比例放大电路原理图。

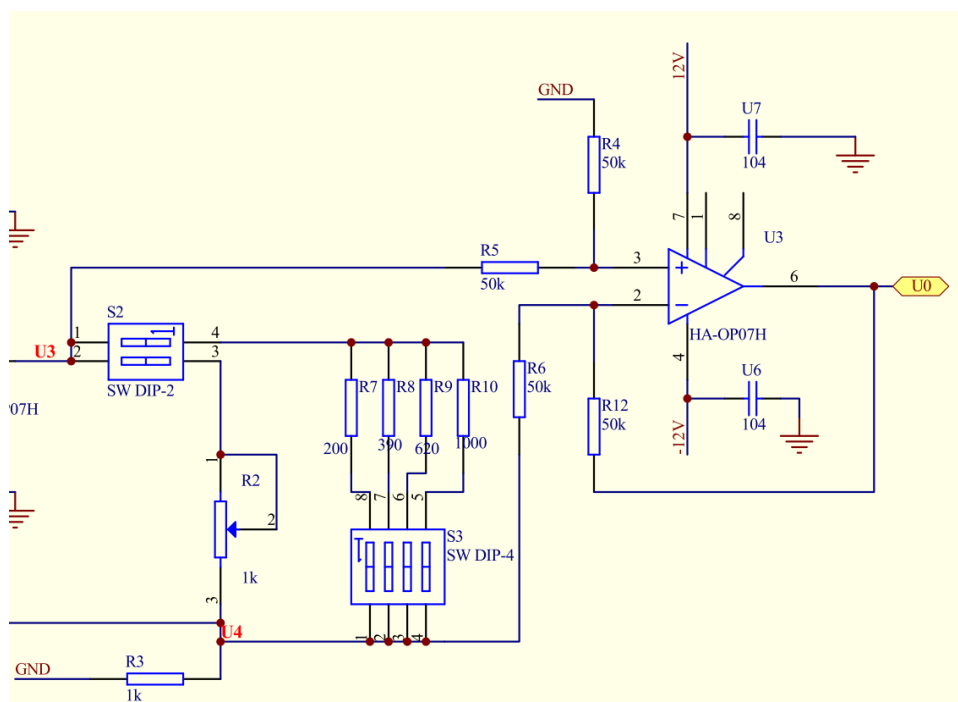


图 3-9 差分比例放大电路

如图 3-9 所示，根据之前的原理可知，图中

$$U_0 = -\frac{R_{12}}{R_6} (U_4 - U_3)$$

因为  $R_{12}=R_6=50K\Omega$  且  $U_4 \leq U_3$  恒成立

所以  $U_0=U_3 - U_4$ ，得到的就是被测电阻两端的压降。

为了考虑实际被测电阻取值的多种情况，本人在此次被测电阻处引入拨码开关 S2。23 导通 14 断开时， $1K\Omega$  滑动变阻器充当被测电阻接入，可进行无极调节。14 导通 23 断开时，由拨码开关 S3 决定由哪个定值电阻接入电路，只是在一定程度上方便了不少，不会破坏原有电路的规律。

### (三)ADC

ADC10 是 MSP430 单片机的片上模数转换器，其转换位数为 10bit，该模块内部是一个 SAR 型的 AD 内核，可以在片内产生参考电压，并且具有数据传输控制器。数据传输控制器能够在 CPU 不参与的情况下，完成 AD 数据向内存任意位置的传输。

ADC10 模块工作的核心是 ADC10 的核，即图中的 10-bitSAR。ADC10 的核将模拟量转换成 10 位数字量并储存在 ADC10MEM 寄存器里。这个核使用 VR+

和  $VR_-$  来决定转换模拟值的门限。模拟电压  $V_{in}$  的输入范围： $VR_- \leq V_{in} \leq VR_+$ 。  
当输入  $V_{in} \geq VR_+$ ,  $N_{ADC}=1023$ 。当输入  $V_{in} \leq VR_-$ ,  $N_{ADC}=0$ 。采样值的计算公式为：

$$N_{ADC} = 1023 \times \frac{V_{in} - V_{R-}}{V_{R+} - V_{R-}}$$

反推模拟量的大小  $V_{in}$

$$V_{in} = \frac{VR_+ \times N_{ADC}}{1023}$$

本次被测电阻两端的电压可由  $V_{in}$  得到，以下是 ADC10 总体结构框图。

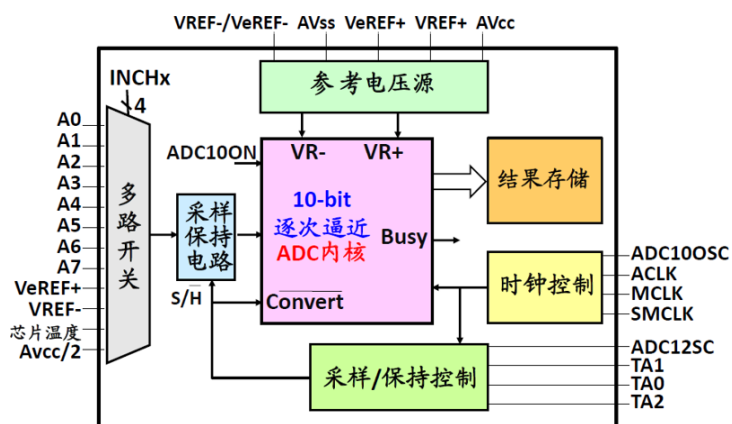


图 3-10 ADC10 总体结构框图

#### (四)128 段 LCD 液晶屏

单片机必须用 I2C 协议去控制 TCA6416A 输出 4 个控制信号 CS、WR、RD、DATA，它们对应的引脚分别为 P1.4、P1.6、P1.5、P1.7，用于控制 LCD 驱动芯片 HT1621，来实现 128 段 LCD 的显示。下图 3-11 和为 LCD 的控制流程和 LCD 原理图。

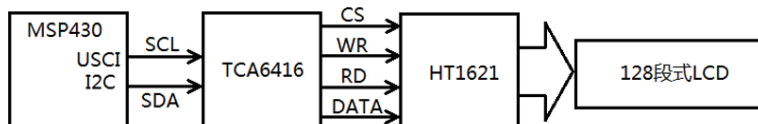


图 3-11 LCD 控制的信号控制图



讲完硬件原理图之后，本人将从仿真的角度验证电路设计的正确性。由于 MSP430 单片机电路板集成有 ADC 功能和 128 段 LCD 液晶显示屏，只仿真外部需个人焊接的恒流源及放大电路模块。

10

### (一)恒流源模块的仿真

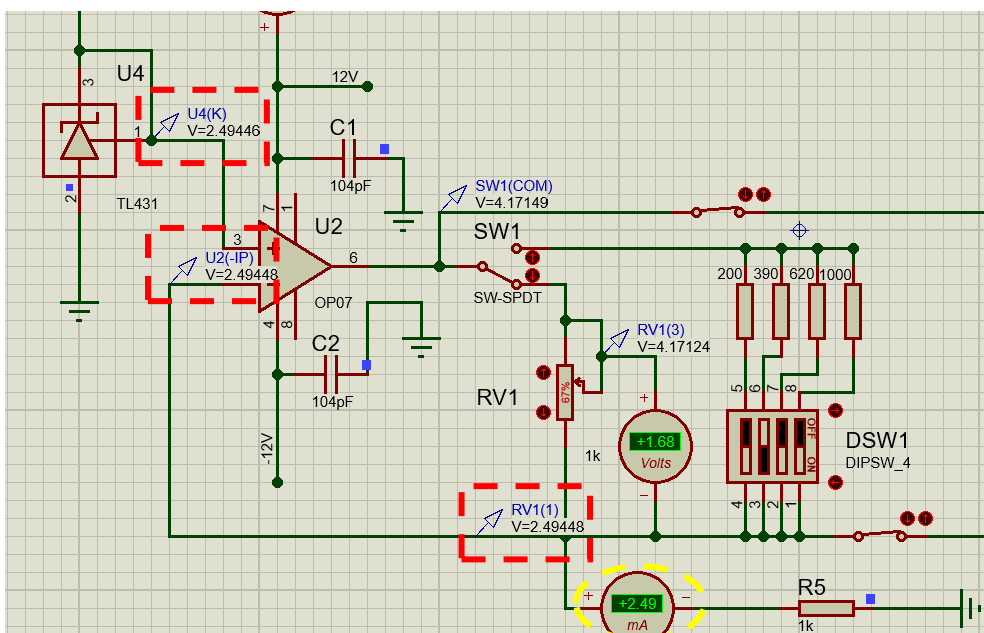


图 3-14 同向放大电路实现恒流源

观察图 3-14，电压探针 U4、U3 处的电压恒定 2.5V，调节电位器 RV1，下方直流毫安表的电流值示数不变， $I=2.5/R5=2.5/1000=2.5\text{mA}$ ，电位器 RV1 上的压降值始终为 2.5mA 乘以电位器接进电路的阻值。探针 SW1 的电压变化范围是 2.5~5.0V，仿真成功，达到预期要求。

### (二)差分比例运算电路仿真

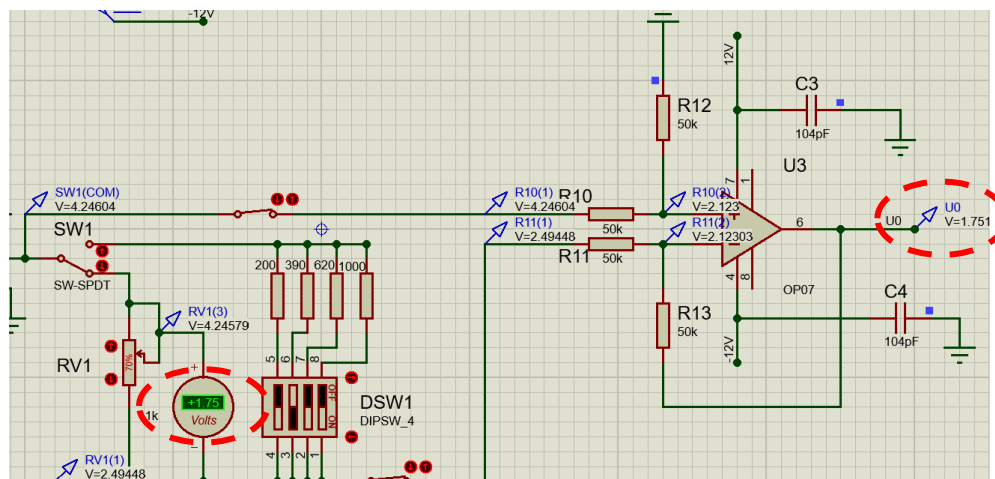


图 3-15 差分比例放大电路

观察图 3-15 电路输出端 U0 的电压始终是被测电阻两端的压降，变化范围为 0~2.5V，仿真成功，达到预期要求。

## 4 软件设计

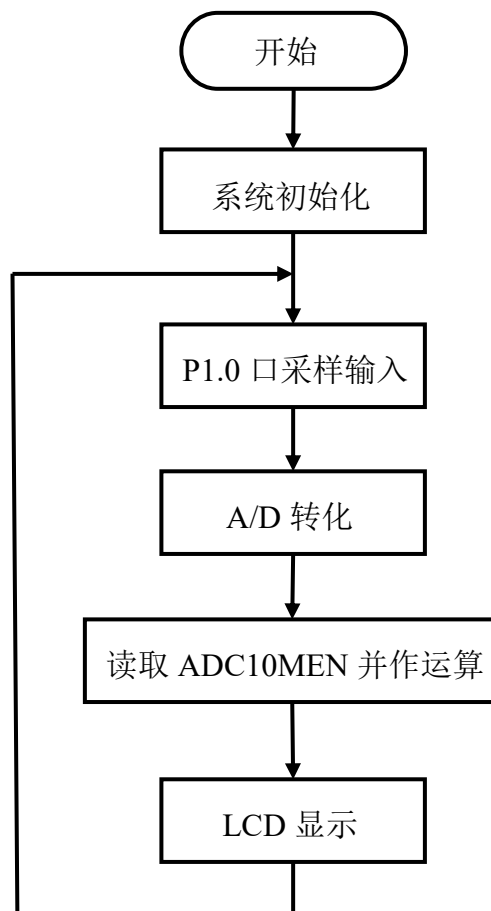


图 3-1 主程序流程图

如图 3-1 所示，为此次主程序的流程图，系统初始化完成后，由 P1.0 复用作为外部模拟电压的输入端口，经过 A/D 转化后，将转化后的值保存在 ADC10MEN 寄存器中，要做的只需用一个变量去保存 ADC10MEN 的值，将变量进行代数运算，最后调用自定义函数 LCD\_Display()即可。由于本人的能力有限，无法徒手打出 ADC10\_init()语句块，借助 Grace 来进行配置。

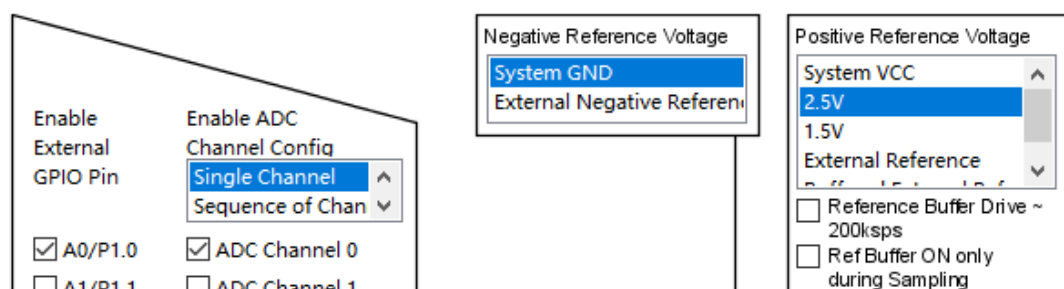


图 4-2 通过 Grace 配置 ADC10



ADC10 一共有 8 个外部输入，且全部与 IO 复用，所以首先需要使能 IO 的复用功能。如图 4-2 所示，我将 P1.0 复用作为外部模拟电压的输入，选择单通道采样，通道选择 ADC Channel 0。由于外部输入电压的最大值为 2.5V，在参考电压配置处选择内部基准电压 2.5V。

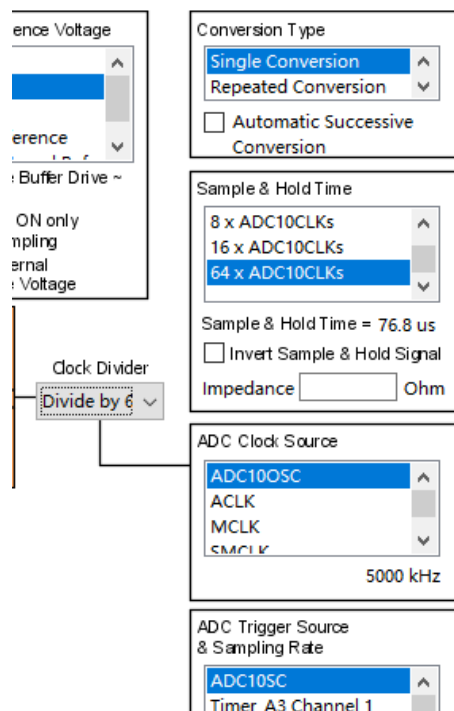


图 4-3

如图 4-3 所示。

选择转换模式为单次转换，“吱一声”动一下。

人工设定采样保持时间为 64 倍 ADC10CLK。

ADC 的时钟选择 5MHz 专用振荡器，8 分频。

采样起始信号选择软件写 ADC10SC 位，通过写 ADC10SC 位来开始一次 AD 转换。

至此，ADC10\_init()配置到此结束，写程序时只需找到将 Grace 工程目录下的 ADC10\_init.c，复制其中的语句即可。下图 4-4 为子函数 LCD\_Display()的流程图，在此声明，RES 变量用来保存被测电阻的阻值并且被测电阻的阻值在原来的基础上扩大了 100 倍，使得显示时保留小数点后两位。根据 RES 的大小不同，显示时采用的取位语句也不同，最终都可以准确地显示在 128 段 LCD 液晶显示屏上。

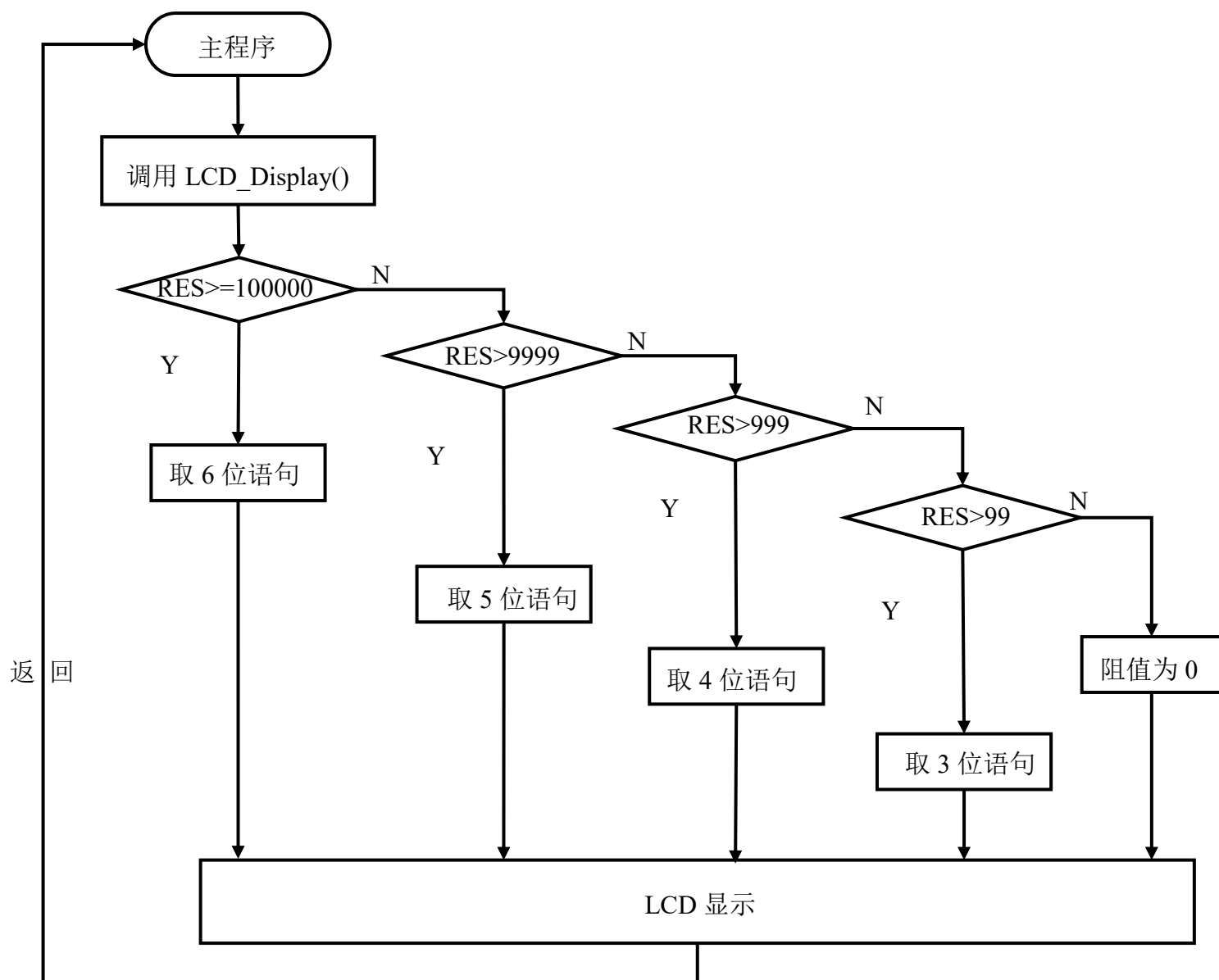


图 4-4 LCD\_Display()流程图

## 5 电阻测量系统调试

硬件与软件设计过后，就是本次电阻测量系统的最后调试阶段。由前面可知，单片机 AD 转化后的结果根据公式

$$V_{in} = \frac{VR_+ \times N_{ADC}}{1023}$$

始终是外部采集的电压值，又由理论可得电流不变，电阻和电压呈现线性的关系。至于这个比例理论已经告诉我们是 2.5mA，即  $U_0 \times 400 = RES$ 。我将表达式直接带入程序进行测试，结果还可以，但是误差能够达到  $10\Omega$  左右，不符合要求。

老师告诉我理论只是理想情况，实际调试时会有许多干扰，要想 LCD 最终显示电阻值，必须得到电阻与电压的线性表达式。要想得到表达式，就必须通过试验、记录大量的数据，最终绘制线性图像来得到。

在老师的指导下，在软件方面，我将程序进行改造，让 LCD 始终先显示采集的电压值。硬件方面，我挑选了 70 多种不同电阻来获得大量的数据，以下是换上不同的电阻值记录的不同的 LCD 显示的电压值，并于万用表测的数据对比。

电阻值 $\Omega$	LCD 显示电压值 (mV)	实际用万用表测得 (mV)
3	0	0
4.7	0	0
5.1	0	0
9.1	0	3
11	4	8
12	7	10
13	9	12
18	21	25.2
20	26	30
30	53	56
36	68	72
51	105	109
51	105	109
62	131	136
75	163	169
91	207	213
110	251	257
110	254	260
120	276	280

150	356	362
160	376	383
200	478	486
300	723	734
360	874	880
390	955	969
470	1163	1177
510	1251	1268
560	1375	1390
620	1554	1572
680	1688	1708
750	1867	1880
820	2013	2030
910	2260	2280
1000	2480	2500

表 1 电阻与电压数据

可以发现万用表与单片机采样得到的电压值十分接近，于是我进行下一步，以 LCD 显示的电压值为横坐标，被测电阻值为纵坐标，绘制线性图像图 5-1。

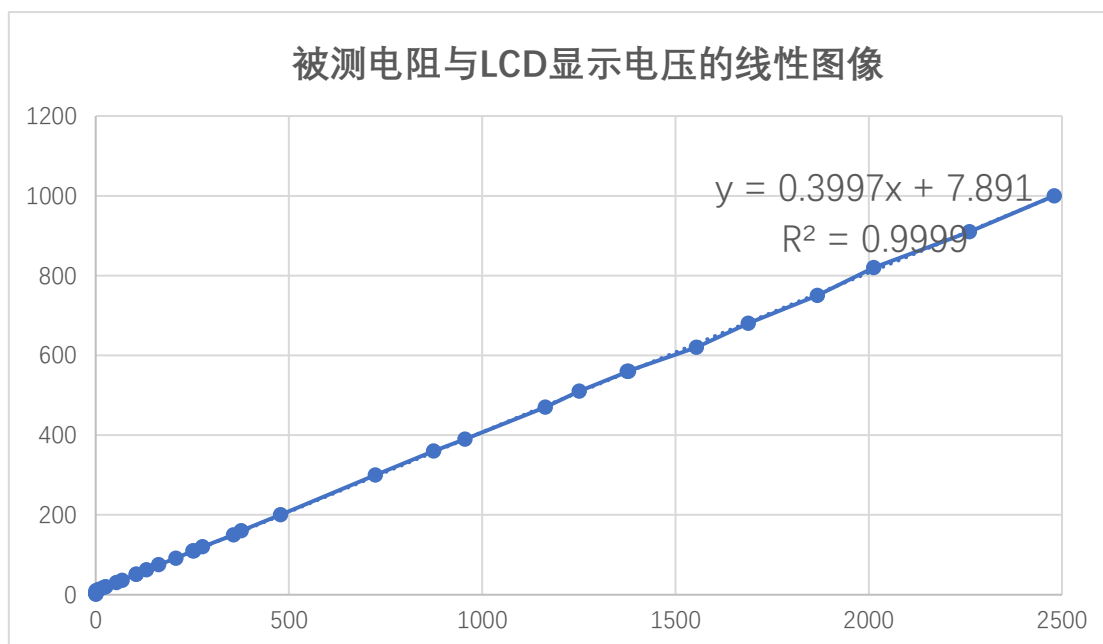


图 5-1 被测电阻与 LCD 显示电压的线性图像

观察到被测电阻与 LCD 显示的电压值线性度非常好，利用 Excel 的图像功能能够显示出函数表达式。此处的 X 即为  $V_{in}$ ，Y 即为被测电阻 RES。

所以  $RES = 0.3997 * V_{in} + 7.891$  即为要写入程序主函数的表达式。

在 Code Composer Studio 软件中编写好代码，烧录入 MSP430G2553 之后，

以下是实验现象图片。为被测电阻收纳盒，为测量  $110\Omega$ ，为测量  $120\Omega$ ，为测量  $470\Omega$ ，为测量  $560\Omega$ 。



图 5-2 被测电阻收纳盒

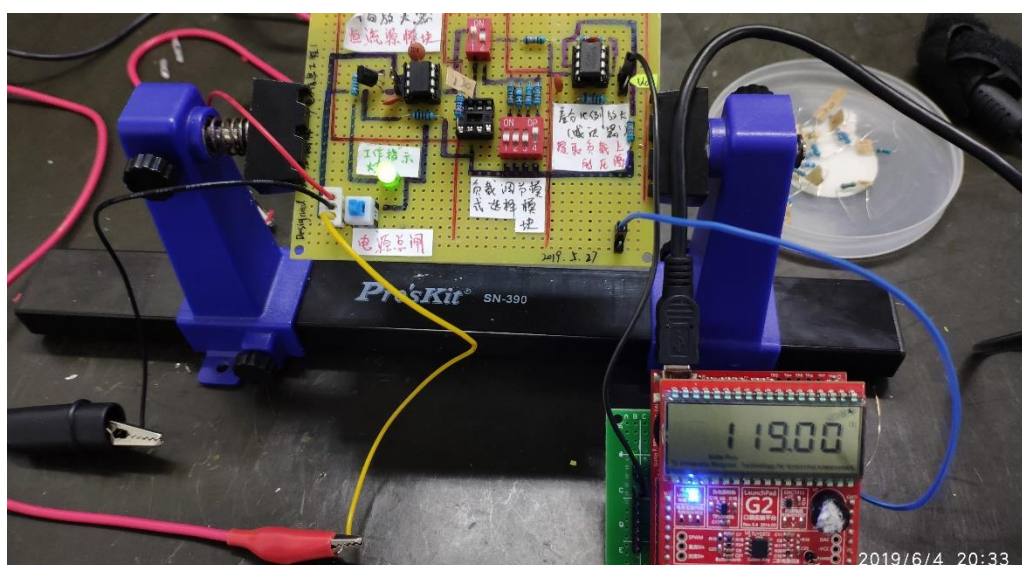


图 5-3 测量  $120\Omega$

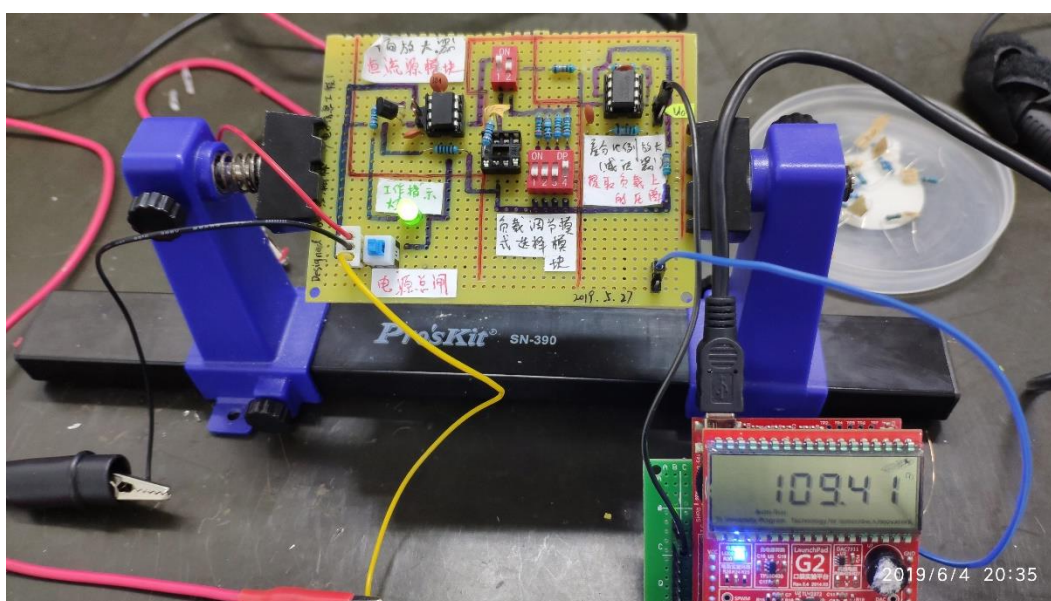


图 5-4 测量  $110\Omega$



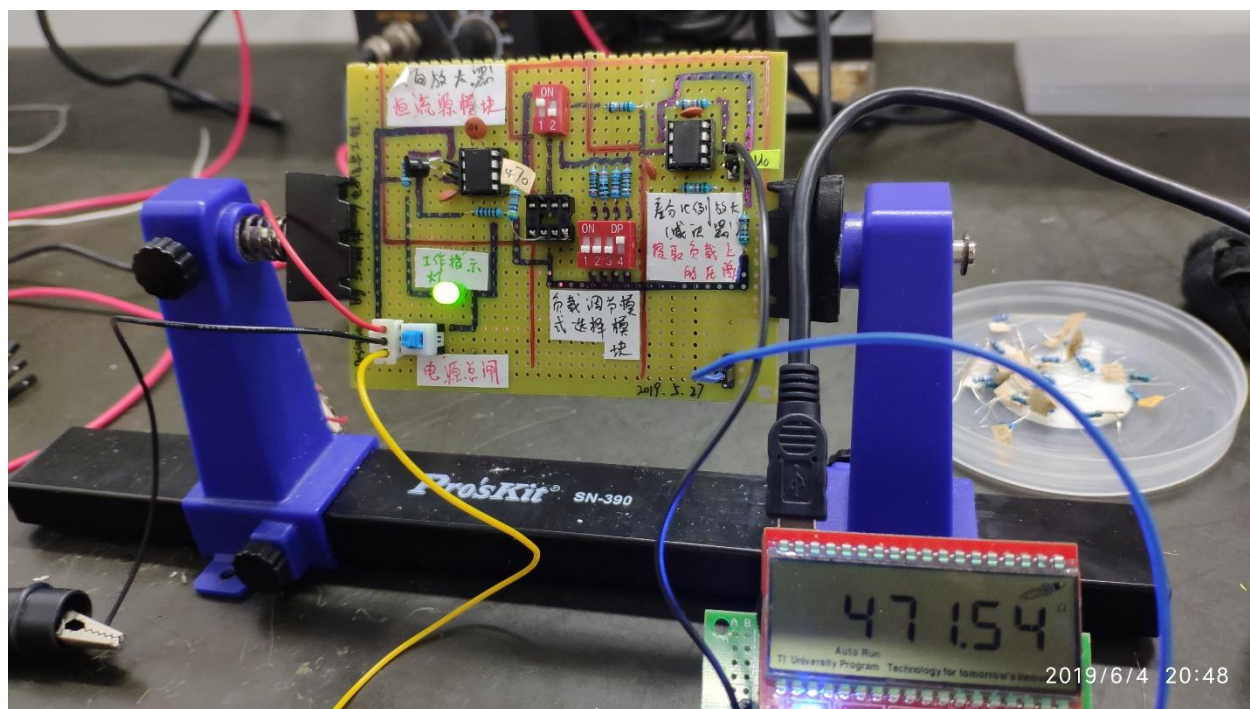


图 5-5 测量  $470\Omega$

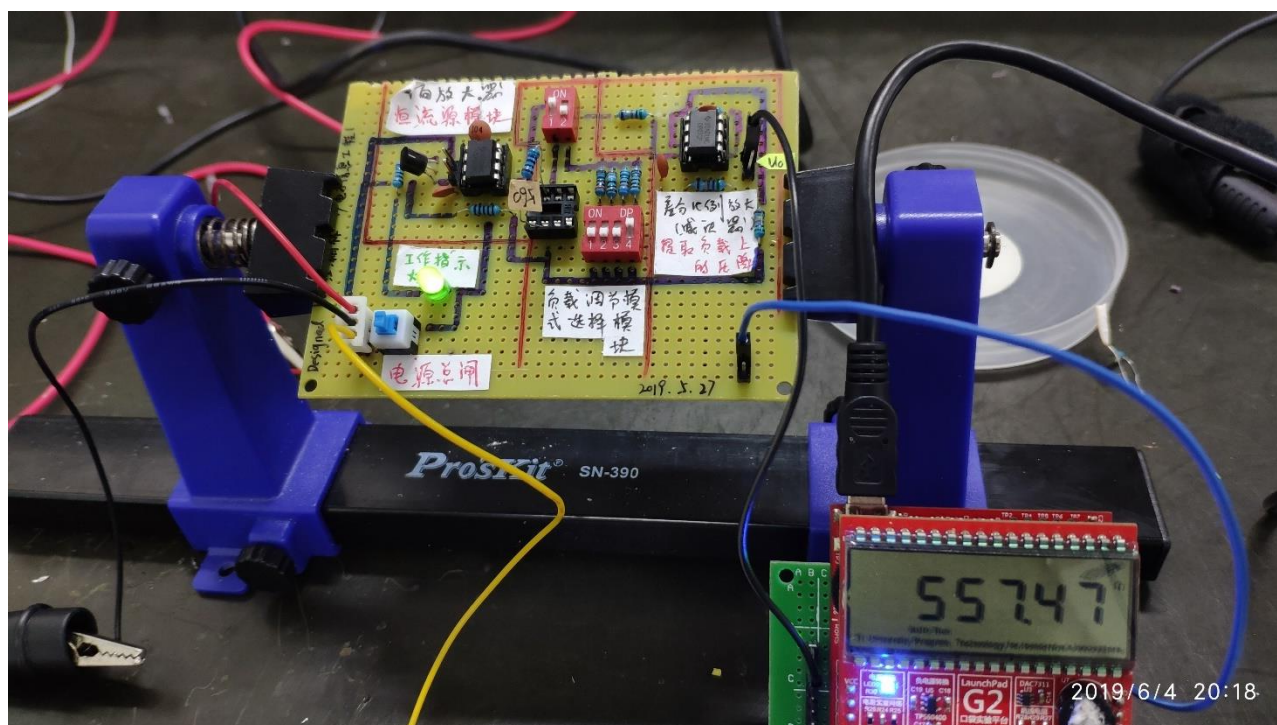


图 5-6 测量  $560\Omega$

根据实际 LCD 显示的电阻值，我又记录了一组数据并与欧姆表测得的数据进行对比，下所示。

标称电阻值 $\Omega$	万用表测得 $\Omega$	LCD 显示 $\Omega$
4.7	5	7.89
5.1	6.6	7.89
9.1	9.7	7.89
11	11.7	9.48
12	12.49	10.68
13	14.8	11.48
18	18	16.28
20	20	18.28
30	30.2	29.07
36	44	35.07
51	51	49.85
62	63.5	60.25
75	76.5	73.04
91	92.7	90.62
110	109.4	109.41
120	123.6	118
150	149	150.18
160	156.8	158.17
200	196	198.14
300	295	296.87
360	360	357.22
390	387	389.6
470	469	471.54
510	504	507.91
560	553	557.47
620	625	627
680	678	682.58
750	751	752.93
820	809	812.48
910	908	911.21
1000	995	998.34

表 2 标称值、LCD 显示的电阻值、欧姆表测得值

通过观察可以发现 LCD 显示的电阻值与实际很接近，能够达到此次课程设计的要求。

## 6 遇到的问题与解决方法

### (一) ADC 采样十分不稳定，跳变剧烈

上文提到，要想 LCD 最终显示电阻值，必须得到电阻与电压的线性表达式。要想得到表达式，就必须通过试验不同的被测电阻、记录大量的电阻电压数据，最终绘制线性图像来得到。记录数据的前提是数据稳定。在实际调试的过程中，LCD 显示的电压值都无法保持稳定，跳变非常剧烈，又怎么能进行下一步操作呢！

- 1、首先我从软件代码入手，将 ADC 采样 20 次再求平均值的方法显示在 LCD 上，以下是我写的多次采样求平均值部分代码：

```
while(1)
{
    ADC10CTL0 |= ENC + ADC10SC; //在主循环中开启 ADC10 转换
    _bis_SR_register(CPUOFF + GIE); //休眠开总中断，等待 ADC10 转换完成后，进入中断运行唤醒 CPU，之后再执行以下代码
    ADC = ADC10MEM; //用 ADC 读取 AD 采样值
    sum=sum+ADC; //sum 变量保存累加 20 次采样的值
    count++; //count 用来记录累加的次数
    if(count==20) //count 累加一旦达到 20
    {
        Average=sum/20; //Average 变量保存累加 20 次采样的平均值
        Vin=(Average*2.5*1000)/1023; //电压表达式，扩大 1000 倍，以 mV 为单位
        RES=(0.3997*Vin + 7.891)*100; //转换为电阻值，并 100 倍处理，精确到小数点后两位
        LCD_Display(); //调用 LCD 显示函数
        count=0; //count 清零，重新记录累加次数
        sum=0; //sum 清零，重新记录累加值
    }
}
```

烧录 MSP430G2553 后，现象还是不见好转。

我想到用 Grace 配置 ADC10 有一个人工设定采样保持时间，或许将它改大一点可以使得示数稳定下来，于是我将采样保持时间改为  $64 \times \text{ADC10CLK}$ ，烧录后还是没用，跳变不定。

2、既然软件没法解决的问题，只好从最原始的硬件着手，我归结为我的电路板输出电压不稳定，但是用万用表检测时十分稳定，搞得我是一头雾水。最后没办法只好求助老师。老师建议我在靠近运放芯片的正负电源引脚上都加一个



104pF 的电容，于是我在 2 颗 OP07 的电源引脚都焊上了 104pF 的电容，共计 4 个。图 6-1 为电容放置位置。

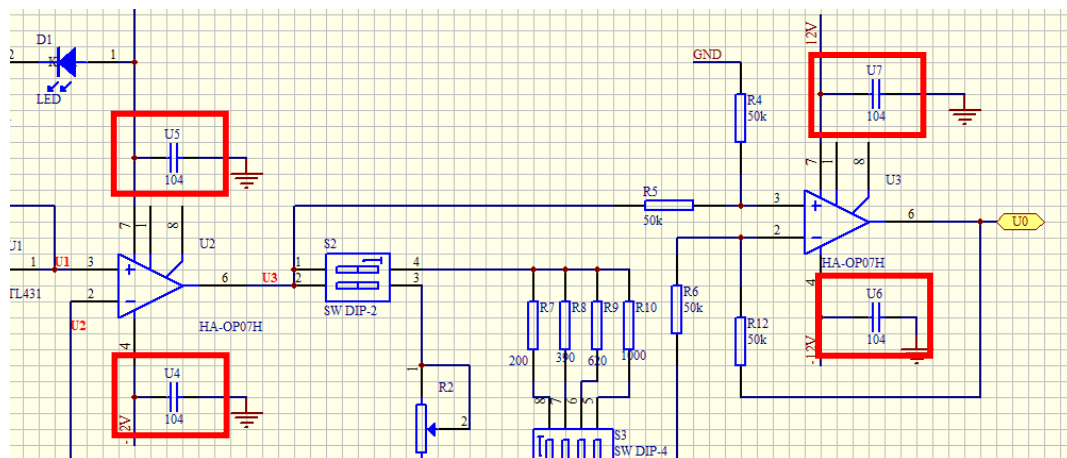


图 6-1 加入滤波电容

加入电容元件以后，电路板上电，将电路板的地线 GND 探针和单片机的 GND 地线探针接起来；将电路板上的电压输出端 U0 引脚与单片机的 P1.0 引脚接起来，观察到此时 LCD 终于稳定下来，可以正常记录数据。

上网查阅资料我了解到，加在靠近运放电源提供引脚处的电容成为旁路电容，作用是把前级或电源携带的高频杂波或信号滤除，使得输出电压稳定，学到了！

## 7 心得体会

我始终认为电子信息工程是一个非常有潜力的专业，并且我对该专业充满了浓厚的兴趣。虽然说专业课让我学得喘不过气来，难度颇大，但我相信这只是一个阶段性的过渡期。

属于工科的电子信息工程需要我有缜密的分析思维，强大的动手能力和学习能力，这些素养还需后期慢慢培养。时光荏苒，如今我已是一名快要读大三的人了，在大学度过的两年里，我收获了许多许多宝贵的经验，积累了一定的专业知识与素养，回首看看两年前的自己，我发现自己已不再是刚进大学的是那个对专业一无所知的小白了。我清楚地知道目前我学到的只是专业的冰山一角，剩下的还需要我跟随老师的脚步，课外自己慢慢摸索。

此次的单片机课程设计【电阻测量系统】中：

被测电阻压降提取功能电路的设计运用到了很多之前专业课的知识，尤其是模拟电子技术。在恒流源模块，涉及到了理想运放的“虚短虚断”；放大电路模块，用到了差分比例放大电路。落实到具体的课程设计我才恍然大悟专业知识并不是白学的，原来是这么运用的，也难怪很久以前专业课老师每次上课都会给我们埋下伏笔。以前自己还没有这么深的感触，但越到后来的专业课程设计我渐渐感觉到专业课知识起到对课程设计的支配与指导作用。

在电压采集模块，需要我对单片机 MSP430G2553 的各个硬件模块，各个引脚的连线熟记于心。本次课程设计主要接触到了 ADC10，需明白如何将外部的电压采集进单片机，内部又如何将模拟量转化为数字量显示在 LCD 屏幕上。Code Composer Studio 的编程很考验我的 C 语言功底，学到现在我只能遗憾地说我只会照搬前人的工程，要达到徒手写代码还有一段长的路要走。

本学期的单片机课结束了，我很遗憾地说好多东西没学懂，课外的时间还得投入大量的时间去钻研这个小东西。

2 年下来，算算做过的课程实验加课程设计大概有 30 个左右，之前都是老师带着我们做着课程设计与实验。这次单片机课程设计真正让我体会到了徒手从零开始做一个项目的感触。

现在我逐渐摸清了要想做出一个能实现具体功能的成品需要哪几个步骤，常

用哪几个模块，怎么把这几个模块联系起来。以我现在的能力，我能够自己设计出能够实现简易功能的成品，当然是建立在查阅资料的基础上。以我自己现在的知识储备量，我知道做出某种功能的成品，电路中运放（OP07 逃不掉）是一定要用的，运放电路常有电压比较器、差分比例放大电路（需要两个输入信号）、同向比例运算电路（其中一个特例可实现电压跟随器）；有运放就会涉及到正负供电，供电的话就需要电源模块，电源模块有两类，恒流源和恒压源。另外电路复杂一点会引入单片机，译码器。译码器的话就要去查它的功能表，筛选或组合输出信号实现功能。

我相信之后我还会遇到更大的困难，更大的挑战，不过本身也不就是一路解决问题，克服困难，提升自我的过程么。未来是未知的，我只有不断进步，不断去适应困苦的环境，才能超越自我，达到新的高度。

感谢此次能够耐心指导我的课程设计老师与同学们的帮助。

## 附件

Code Composer Studio 电阻测量 main.c 代码程序

焊接元件清单一张

Proteus 8 Professional 仿真布局图一张

Protel 99 SE 原理图一张

Protel 99 SE PCB 板布局连线图一张

恒流源及放大电路 PCB 正反面照片一张

任务书一张

```
#include <msp430.h>
#include "LCD_128.h"
#include "HT1621.h"
#include "TCA6416A.h"

long RES;//电阻变量
long Vin;//输入电压变量
void ADC10_ISR(void);
void ADC10_init(void);
void LCD_Init();
void LCD_Display();

void main()
{
    long sum=0;
    long Average; //Average变量保存累加20次采样的平均值
    long ADC; //用ADC读取AD采样值
    int count=0;
    WDTCTL=WDTPW+WDTHOLD;
    ADC10_init();
    LCD_Init();
    while(1)
    {
        ADC10CTL0 |= ENC + ADC10SC; //在主循环中开启 ADC10 转换
        _bis_SR_register(CPUOFF + GIE);
        //休眠开总中断，等待ADC10转换完成后，进入中断运行唤醒 CPU，再执行之后的代码
        ADC = ADC10MEM;
        sum=sum+ADC;//sum变量保存累加20次采样的值
        count++;//count用来记录累加的次数
        if(count==20)//count累加一旦达到20
        {
            Average=sum/20;
            Vin=(Average*2.5*1000)/1023;//电压表达式，扩大1000倍，以mV为单位
            RES=(0.3997*Vin + 7.891)*100; //转换为电阻值，并100倍处理，精确到小数点后两位
            LCD_Display(); //调用LCD显示函数
            count=0;//count清零，重新记录累加次数
            sum=0;//sum清零，重新记录累加值
        }
    }
}
```

```

/*****
* 名      称: LCD_Init()
* 功      能: 初始化LCD显示相关的硬件, 并预设固定不变的显示内容
* 入口参数: 无
* 出口参数: 无
* 说      明: 无
* 范      例: 无
*****/

void LCD_Init()    //-----显示固定不变的LCD段-----
{
    TCA6416A_Init(); //I2C 模块的初始化由 TCA6416A 初始化函数在内部完成了
    HT1621_init();   //LCD_128 库函数由 HT1621 初始化函数在内部引用了
    LCD_DisplaySeg(_LCD_AUTO);
    LCD_DisplaySeg(_LCD_RUN);
    LCD_DisplaySeg(_LCD_TI_logo);
    LCD_DisplaySeg(_LCD_QDU_logo);
    LCD_DisplaySeg(_LCD_OHOM);
    LCD_DisplaySeg(_LCD_DOT3);
}

/*****
* 名      称: LCD_Displaly()
* 功      能: 将电阻值显示出来
* 入口参数: 无
* 出口参数: 无
* 说      明: 拆分数字等几部分
* 范      例: 无
*****/

void LCD_Display()
{
    //-----清除6位显示数字-----
    LCD_DisplayDigit(LCD_DIGIT_CLEAR,1);
    LCD_DisplayDigit(LCD_DIGIT_CLEAR,2);
    LCD_DisplayDigit(LCD_DIGIT_CLEAR,3);
    LCD_DisplayDigit(LCD_DIGIT_CLEAR,4);
    LCD_DisplayDigit(LCD_DIGIT_CLEAR,5);
    LCD_DisplayDigit(LCD_DIGIT_CLEAR,6);
    //-----根据RES的大小拆分并显示数字-----
    if(RES>=100000)//100000 (6位)
    {
        LCD_DisplayDigit(RES/100000,1);
        LCD_DisplayDigit((RES%100000)/10000,2);
        LCD_DisplayDigit((RES%10000)/1000,3);
    }
}

```

```

        LCD_DisplayDigit((RES%1000)/100,4);
        LCD_DisplayDigit((RES%100)/10,5);
        LCD_DisplayDigit(RES%10,6);
    }
    else if(RES>9999)//10000~99999 (5位)
    {
        LCD_DisplayDigit(RES/10000,2);
        LCD_DisplayDigit((RES%10000)/1000,3);
        LCD_DisplayDigit((RES%1000)/100,4);
        LCD_DisplayDigit((RES%100)/10,5);
        LCD_DisplayDigit(RES%10,6);
    }
    else if(RES>999)//1000~9999 (4位)
    {
        LCD_DisplayDigit(RES/1000,3);
        LCD_DisplayDigit((RES%1000)/100,4);
        LCD_DisplayDigit((RES%100)/10,5);
        LCD_DisplayDigit(RES%10,6);
    }
    else if(RES>99)//100~999 (3位)
    {
        LCD_DisplayDigit(RES/100,4);
        LCD_DisplayDigit((RES%100)/10,5);
        LCD_DisplayDigit(RES%10,6);
    }
    else
    {
        LCD_DisplayDigit(0,4);
        LCD_DisplayDigit(RES/10,5);
        LCD_DisplayDigit(RES%10,6);
    }
    HT1621_Reflash(LCD_Buffer);//-----更新缓存, 真正显示-----
}

/*****
* 名      称: ADC10_init(void)
* 功      能: 初始化ADC10, 单次手动采样
* 入口参数: 无
* 出口参数: 无
* 说      明: 无
* 范      例: 无
*****/

```

```

void ADC10_init(void)
{
    ADC10CTL0 &= ~ENC;
    ADC10CTL0 = ADC10IE|ADC10ON | REFON| REF2_5V | ADC10SHT_3 | SREF_1;
    //启用内部2.5V参考电压，采样保持时间为64 x ADC10CLKs
    ADC10CTL1 = CONSEQ_0 | ADC10SSEL_0 | ADC10DIV_7 | SHS_0 | INCH_0;
    //ADC时钟分频为7分频
    ADC10AE0 = 0x1; //P1.0为ADC输入端
    __delay_cycles(30000);
    ADC10CTL0 |= ENC;
}

```

```

/*****
* 名      称: ADC10_ISR_HOOK
* 功      能: ADC转换完成后唤醒CPU
* 入口参数: 无
* 出口参数: 无
* 说      明: 直接从CCS的Example中移植过来
* 范      例: 无
*****/

#pragma vector=ADC10_VECTOR
__interrupt void ADC10_ISR(void)
{
    _bic_SR_register_on_exit(LPM4_bits);
}

```



元器件名称	数量
LDE-Yellow	1
4 路拨码开关	1
2 路拨码开关	1
DIP-8 插座	3
104pF 电容	4
1K $\Omega$ 电阻	2
200 $\Omega$ 电阻	1
390 $\Omega$ 电阻	2
620 $\Omega$ 电阻	1
50k $\Omega$ 电阻	4
OP07 芯片	2
TL431	1
蓝白自锁开关	1
排针	1

表 3 焊接元件清单