

# 基于单片机的电容测量仪设计

## 前言

目前，随着电子工业的发展，电子元器件急剧增加，电子元器件的适用范围也逐渐广泛起来，在应用中我们常常要测定电容的大小。

在电子产品的生产和维修中，电容测量这一环节至关重要，一个好的电子产品应具备一定规格年限的使用寿命。因此在生产这一环节中，对其产品的检测至关重要，而检测电子产品是否符合出产要求的关键在于检测其内部核心的电路，电路的好坏决定了电子产品的好与坏，而电容在基本的电子产品的集成电路部分有着其不可替代的作用。同样，在维修人员在对电子产品的维修中，电路的检测是最基本的，有时需要检测电路中各个部件是否工作正常，电容器是否工作正常。因此，设计可靠，安全，便捷的电容测试仪具有极大的现实必要性。

# 1 选题背景

## 1.1 电容测试仪的发展历史及研究现状

当今电子测试领域，电容的测量已经在测量技术和产品研发中应用的十分广泛。电容通常以传感器形式出现，因此，电容测量技术的发展归根结底就是电容传感器的发展。由最初用交流不平衡电桥就能测量基本的电容传感器。最初的电容传感器有变面积型，变介质介电常数型和变极板间型。现在的电容式传感器越做越先进，现在用的比较多的有容栅式电容传感器，陶瓷电容压力传感器等。电容测量技术发展也很快现在的电容测量技术也由单一化发展为多元化。现在国内外做传感器的厂商也比较多，在世界范围内做电容传感器做的比较好的公司有：日本 figaro、德国 tectis、美国 alphasense。中国本土测量仪器设备发展的主要瓶颈。尽管本土测试测量产业得到了快速发展，但客观地说中国开发测试测量仪器还普遍比较落后。每当提起中国测试仪器落后的原因，就会有许多不同的说法，诸如精度不高，外观不好，可靠性差等。实际上，这些都还是表面现象，真正影响中国测量仪器发展的瓶颈为：

1.测试在整个产品流程中的地位偏低。由于人们的传统观念的影响，在产品的制造流程中，研发始终处于核心位置，而测试则处于从属和辅助位置。关于这一点，在几乎所有的研究机构部门配置上即可窥其一斑。这种错误观念上的原因，造成整个社会对测试的重视度不够，从而造成测试仪器方面人才的严重匮乏，造成相关的基础科学研究比较薄弱，这是中国测量仪器发展的一个主要瓶颈。实际上，即便是研发队伍本身，对测试的重视度以及对仪器本身的研究也明显不够。

2.面向应用和现代市场营销模式还没有真正建立起来。本土仪器设备厂商只是重研发，重视生产，重视狭义的市场，还没有建立起一套完整的现代营销体系和面向应用的研发模式。传统的营销模式在计划经济年代里发挥过很大作用，但无法满足目前整体解决方案流行年代的需求。所以，为了快速缩小与国外先进公司之间的差距，国内仪器研发企业应加速实现从面向仿制的研发向面向应用的研发的过渡。特别是随着国内应用需求的快速增长，为这一过渡提供了根本动力，应该利用这些动力，跟踪应用技术的快速发展。

3.缺乏标准件的材料配套体系。由于历史的原因，中国仪器配套行业的企业多为良莠不齐的小型企业，标准化的研究也没有跟上需求的快速发展，从而导致仪器的材

料配套行业的技术水平较低。虽然目前已有较大的改观，但距离整个产业的要求还有一定距离，所以，还应把标准化和模块化的研究放到重要的位置。还有，在技术水平没有达到的条件下，一味地追求精度或追求高指标，而没有处理好与稳定性之间的关系。上述这些都是制约本土仪器发展的因素。

近年来我国测量仪器的可靠性和稳定性问题得到了很多方面的重视，状况有了很大改观。测试仪器行业目前已经越过低谷阶段，重新回到了快速发展的轨道，尤其最近几年，中国本土仪器取得了长足的进步，特别是通用电子测量设备研发方面，与国外先进产品的差距正在快速缩小，对国外电子仪器巨头的垄断造成了一定的冲击。随着模块化和虚拟技术的发展，为中国的测试测量仪器行业带来了新的契机，加上各级政府日益重视，以及中国自主应用标准研究的快速进展，都在为该产业提供前所未有的动力和机遇。从中国电子信息产业统计年鉴中可以看出，中国的测试测量仪器每年都以超过 30% 以上的速度在快速增长。在此快速增长的过程中，无疑催生出了许多测试行业新创企业，也催生出了一批批可靠性和稳定性较高的产品。

## 1.2 现今的电容测量技术手段

电容器作为非常重要的一个电学元件在现代电子技术中有着非常广泛的用途，电容定义为：电容器所带的电荷量  $Q$  与电容器两极板间的电势差  $U$  的比值，即：
$$C = \frac{Q}{U}$$
。这种原始的方法必须通过测量两个物理量来计算电容的大小，而其中的  $Q$  是比较难以测量的量。目前常用的两种测量电容的实现方法：一是利用多谐振荡产生脉冲宽度与电容值成正比信号，通过低通滤波后测量输出电压实现；二是利用单稳态触发装置产生与电容值成正比门脉冲来控制通过计数器的标准计数脉冲的通断，即直接根据充放电时间判断电容值。

利用多谐振荡原理测量电容的方案硬件设计比较简单，但是软件实现相对比较复杂，而直接根据充放电时间判断电容值的方案虽然基本上没有用到软件部分，但是硬件却又十分的复杂。而且他们都无法直观的把测量的电容值大小显示出来。

根据上面两种方案的优缺点，本次设计提出了硬件设计和软件设计都相对比较简单方案：基于 AT89C51 单片机和 555 芯片的数显式电容测量。该方案主要是根据 555 芯片的应用特点，把电容的大小转变成 555 输出频率的大小，进而可以通过单片机对 555 输出的频率进行测量。本方案的硬件设计和软件设计都相对简单。

## 2 基于单片机电容测量硬件设计

### 2.1 设计方案

本次设计中考虑了三种设计方案,三种设计方案中主要区别在于硬件电路和软件设计的不同,对于本设计三种方案均能够实现,最后根据设计要求、可行性和设计成本的考虑选择了基于 AT89C51 单片机和 555 芯片构成的多谐振荡电路的测量的方案。现在一一介绍论证如下:

方案一、利用多谐振荡原理测量电容测量原理如下图所示。电容 C 电阻 R 和 555 芯片构成一个多谐振荡电路。在电源刚接通时(K 合上),电容 C 上的电压为零,多谐振荡器输出  $V_0$  为高电平  $V_0$  通过 R 对电容 C 充电。当 C 上冲得的电压  $V_C = V_{T+}$  时,施密特触发器翻转,  $V_0$  变为低电平, C 又通过 R 放电,  $V_C$  下降。当  $V_C = V_T$  时施密特触发器又翻转, 输出  $V_C$  又变为高电平, 如此往复产生震荡波形。

由理论分析可知

$$t_{PH} = RC \ln(V_{OH} - V_{T-}) / (V_{OH} - V_{T+}) \quad (1)$$

$$t_{PL} = RC \ln(V_{OL} - V_{T+}) / (V_{OL} - V_{T-}) \quad (2)$$

$$\begin{aligned} T &= t_{PH} + t_{PL} \\ &= RC (\ln(V_{OH} - V_{T-}) / (V_{OH} - V_{T+}) + \ln(V_{OL} - V_{T+}) / (V_{OL} - V_{T-})) \end{aligned} \quad (3)$$

$$\text{令 } D = \ln(V_{OH} - V_{T-}) / (V_{OH} - V_{T+}) + \ln(V_{OL} - V_{T+}) / (V_{OL} - V_{T-}) \quad (4)$$

$$\text{则 } T = RCD, T_0 = RC_0 D, T_x = RC_x D$$

$$T_x / T_0 = C_x / C_0 \quad (5)$$

$$C_x = (T_x / T_0) C_0 = (f_0 / f_x) C_0 \quad (6)$$

有式(6)和测得的校准值  $T_0$  测量值  $T_x$  及存放的软件中的标准电容值 C 可得出待测电容值  $C_x$ 。实际应用中也可以通过测量  $f_0$  和  $f_x$  来算出  $C_x$

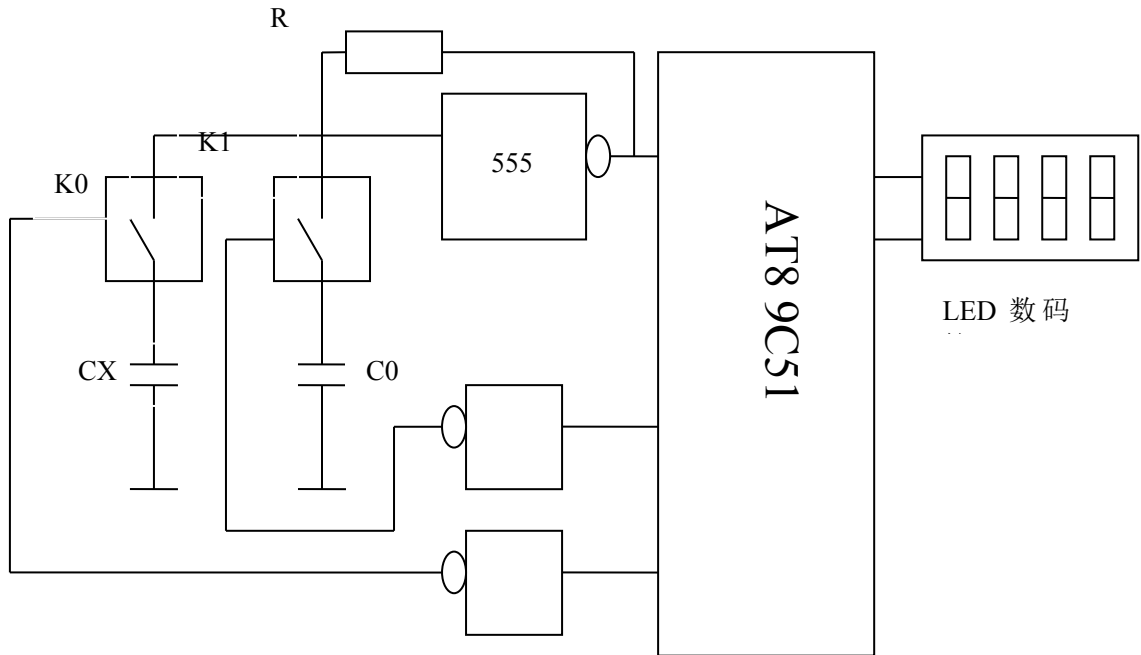


图1 电容测量原理图

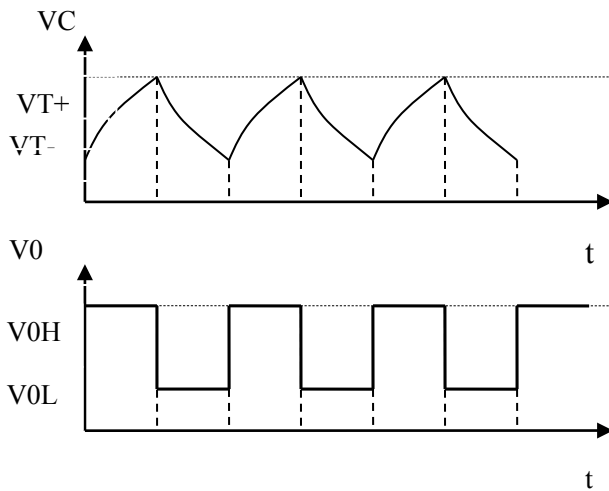


图2 震荡波形图

测量误差分析：由式(6)可以看出，经过软件校准后得出的  $C_x$  结果与  $T_x/T_0$  的值有关。这样单片机晶振频率的绝对精度，环境温度的变化和电源电压的绝对精度引起的误差被消除。测量结果主要受标准电容  $C_0$  的绝对精度影响，因此应该选择精度高、稳定性好的  $C_0$ ；其他误差来源包括周期测量的量化误差，除法运算产生的余数误差，电源电压的波动造成谐振频率偏移带来的误差，因此电路要用稳压性能好的稳压电源

这种方法的利用了一个参考的电容实现，虽然硬件结构简单，软件实现却相对比较复杂。

方案二、直接根据充放电时间判断电容值

这种电容测量方法主要利用了电容的充放电特性  $Q = UC$ ，放电常数  $\tau = RC$ ，通过测量与被测电容相关电路的充放电时间来确定电容值。一般情况下，可设计电路使  $T = ARC$  (T 为振荡周期或触发时间；A 为电路常数与电路参数有关)。这种方法中应用了 555 芯片组成的单稳态触发器，在秒脉冲的作用下产生触发脉冲，来控制门电路实现计数，从而确定脉冲时间，通过设计合理的电路参数，使计数值与被测电容相对应。其原理框图如图 3 所示。

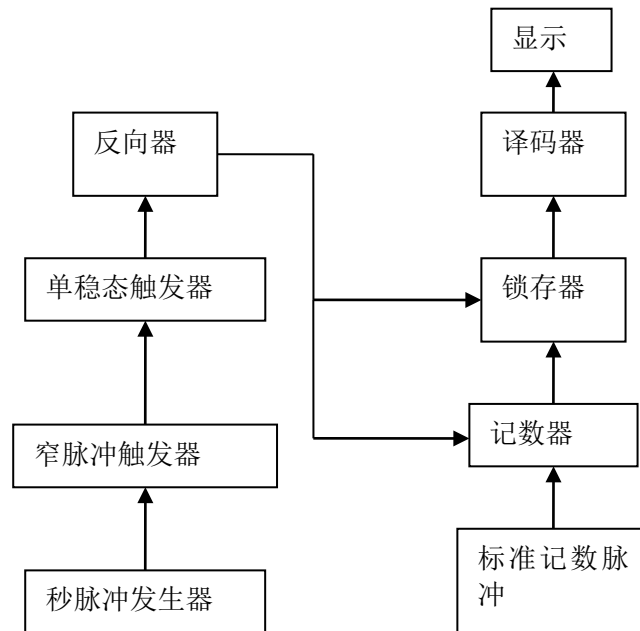


图 3 电路原理框图

误差分析：这种电容测量方法的误差主要由两部分组成：一部分是由 555 芯片构成的振荡电路和触发电路由于非线性造成的误差，其中最重要的是单稳态触发电路的非线性误差， $\Delta C_0 / C_0 = \Delta T / T$  (T 由充放电时间决定， $C_0$  是被测电容值)；另一部分是由数字电路的量化误差引起，是数字电路特有的误差该误差相对影响较小，可忽略不计。

这种方法硬件结构相对复杂，实际上是通过牺牲硬件部分来减轻软件部分的负担，但在具体设计中会碰到很大问题，而且硬件一旦设计好，可变性不大。

方案三、基于 AT89C51 单片机和 555 芯片构成的多谐振荡电路电容测量

这种电容测量方法主要是通过一块 555 芯片来测量电容，让 555 芯片工作在直接反馈无稳态的状态下，555 芯片输出一定频率的方波，其频率的大小跟被测量的电容之间的关系是： $f = 0.772 / (R * C_x)$ ，我们固定  $R$  的大小，其公式就可以写为： $f = k / C_x$ ，只要我们能够测量出 555 芯片输出的频率，就可以计算出测量的电容。计算频率的方法可以利用单片机的计数器  $T_0$  和中断  $\overline{INT0}$  配合使用来测量，这种研究方法相当的简单。系统框图见图 4。

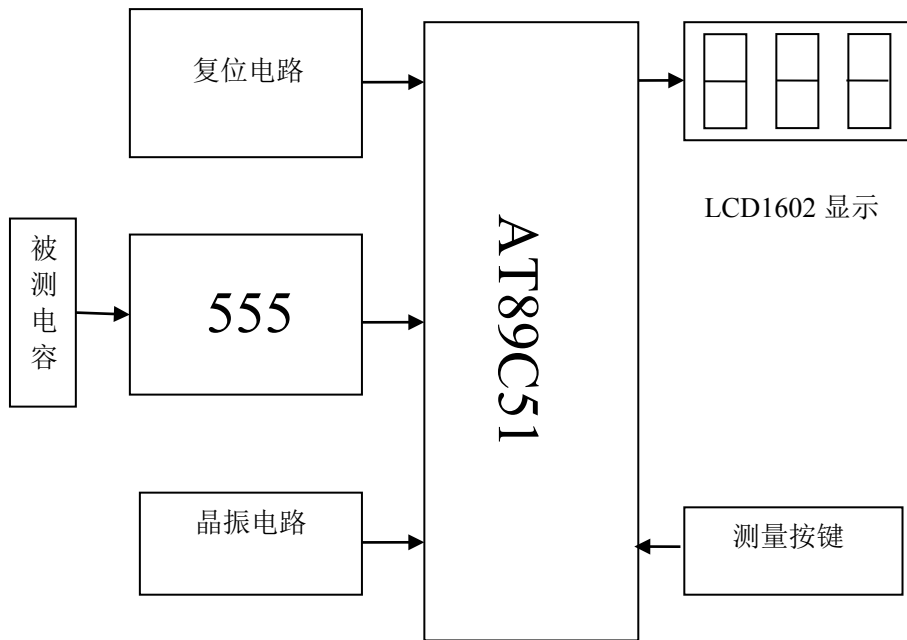


图 4 系统框图

图中给出了整个系统设计的系统框图，系统主要由四个主要部分组成，单片机和晶振电路设计，555 芯片电路设计，显示电路设计，复位电路设计。

## 2.2 硬件设计的任务

AT89C51 基本工作电路设计：使单片机正常工作；

时钟电路：为单片机提供时钟信号；

复位电路：为单片机提供高电平复位信号；

555 芯片电路：把电容的大小转变成输出频率的大小；

显示电路：显示当前测量电容的大小；

按键电路：开始测量电容；

## 2.3 基于 AT89C51 电容测量系统硬件设计详细分析

### 2.3.1 AT89C51 单片机工作电路

本设计的核心是单片机电路，考虑到需要一个中断输入，存储容量、外部接口对单片机端口的需要以及兼顾到节约成本的原则，选用了常用的 AT89C51 单片机。AT89C51 是低功耗、高性能、经济的 8 位 CMOS 微处理器，工作频率为 0—24MHz，内置 4K 字节可编程只读闪存，128x8 位的内部 RAM，16 位可编程 I/O 总线。它采用 Atmel 公司的非易存储器制造技术，与 MCS51 的指令设置和芯片引脚可兼容。AT89C51 可以按照常规方法进行编程，也可以在线编程。其将通用的微处理器和 Flash 存储器结合在一起，特别是可反复擦写的 Flash 存储器可有效降低开发成本。AT89C51 工作的最简单的电路是其外围接一个晶振和一个复位电路，给单片机接上电源和地，单片机就可以工作了。其最简单的工作原理图如下图。

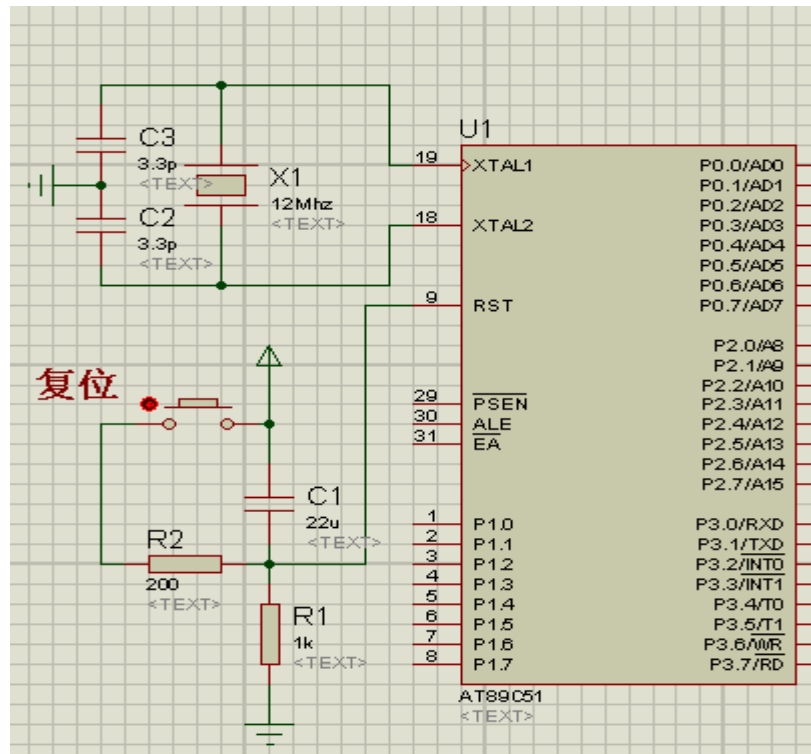


图 5 AT89C51 单片机工作电路



### 2.3.2 基于 AT89C51 电容测量系统复位电路

MCS-51 的复位是由外部的复位电路来实现的。MCS-51 单片机片内复位，复位引脚 RST 通过一个斯密特触发器用来抑制噪声，在每个机器周期的 S5P2，斯密特触发器的输出电平由复位电路采样一次，然后才能得到内部复位操作所需要的信号。

复位电路通常采用上电自动复位和按钮复位两种方式。上电自动复位是通过外部复位电路的电容充电来实现的。只要 Vcc 的上升时间不超过 1ms，就可以实现自动上电复位。除了上电复位外，有时还需要按键手动复位。按键手动复位有电平方式和脉冲方式两种。其中电平复位是通过 RST 端经电阻与电源 Vcc 接通而实现的。按键脉冲复位则是利用 RC 微分电路产生的正脉冲来实现的。

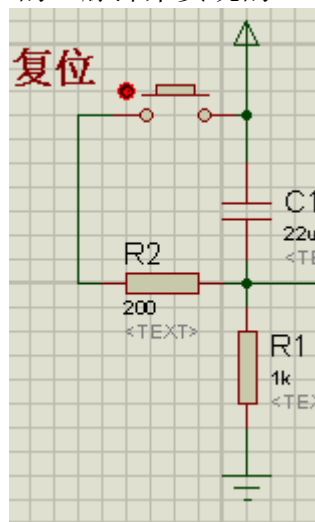


图 6 上电复位电路

上图兼有上电复位和按钮复位的电路。在单片机设计中，若有外部扩展的 I/O 接口电路需初始复位，如果它们的复位端和 MCS-51 单片机的复位端相连，复位电路中的 R、C 参数要受到影响，这时复位电路中的 R、C 参数要统一考虑以保证可靠的复位。如果单片机 MCS-51 单片机与外围 I/O 接口电路的复位电路和复位时间不完全一致，使单片机初始化程序不能正常运行，外围 I/O 接口电路的复位也可以不和 MCS-51 单片机复位端相连，仅采用独立的上电复位电路。一般来说，单片机的复位速度比外围 I/O 快些。若 RC 上电复位电路接 MCS-51 单片机和外围电路复位端，则能使系统可靠地同步复位。为保证系统可靠复位，在初始化程序中应用到一定的复位延迟时间。

复位电路软件程序跑飞或者硬件发生错误的时候产生一个复位信号，控制 MCS-51 单片机从 0000H 单元开始执行程序，重新执行软件程序。此电路的输出端 RESET 接在单片机的复位引脚。

### 2.3.3 基于 AT89C51 电容测量系统时钟电路

时钟在单片机中非常重要，单片机各功能部件的运行都是以时钟频率为基准。时钟频率直接影响单片机的速度，时钟电路的质量也直接影响单片机系统的稳定性。常用的时钟电路有两种方式，一种是内部时钟方式，另一种为外部时钟方式。

内部时钟方式：内部时钟方式电路图如下图所示。

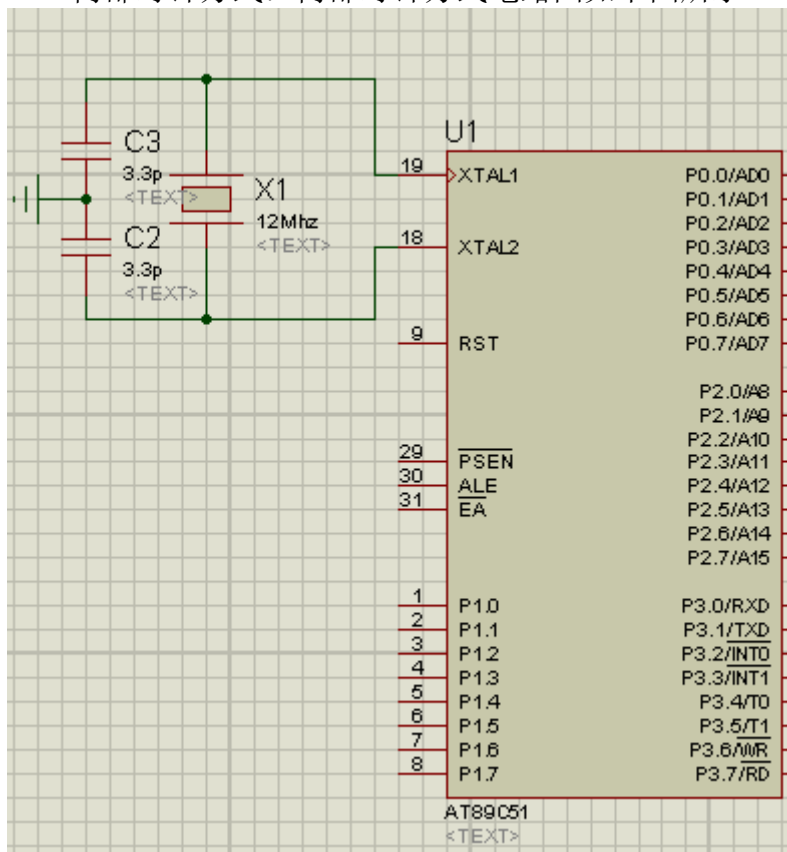


图7 内部时钟电路

MCS-51 单片机内部有一个用与构成振荡器的高增益反相放大器，该高增益反相放大器的输入端为芯片引脚 XTAL1，输出端为引脚 XTAL2。这两个引脚接石英晶体振荡器和微调电容，就构成一个稳定的自激振荡器电路。

电路中的电容 C1 和 C2 典型值通常选择为 30PF

左右。对外接电容的值虽然没有严格的要求，但是电容的大小会影响振荡器频率的高低、振荡器的稳定性和起振的快速性。晶体的振荡频率的范围通常是在 1.2MHz—12MHz 之间。晶体的频率越高，则系统的时钟频率也就越高，单片机的运行速度也就越快。为了提高温度稳定性，应采用温度稳定性能好的 NPO 高频电容。MCS-51 单片机常选择振荡频率 6MHz 或 12MHz 的石英晶体。

外部时钟方式：外部时钟方式电路图如下图所示。

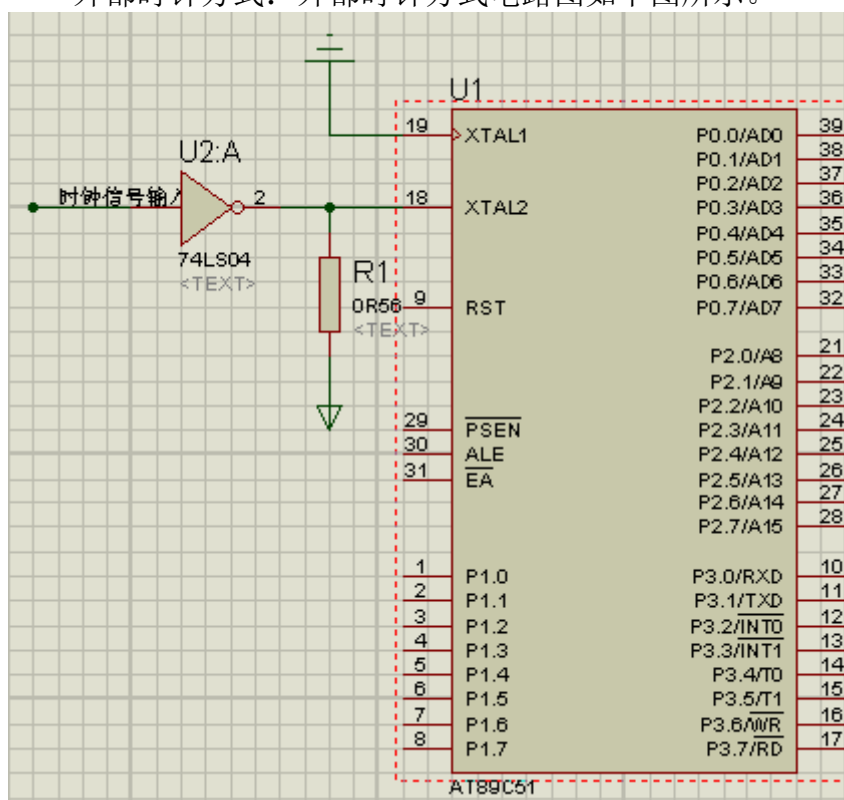


图 8 外部时钟电路

外部时钟方式是使用外部振荡脉冲信号，常用于多片 MCS-51 单片机同时工作，以便于同步。对外部脉冲信号只要求高电平的持续时间大于 20us，一般为低于 12MHz 的方波。

外部的时钟源直接接到 XTAL2 端，直接输入到片内的时钟发生器上。由于 XTAL2 的逻辑电平不是 TTL 的，因此要外接一个 4.7k~10k 的上拉电阻。

这次的设计采用 MCS-51 的内部时钟方式。因为外部时钟方式是用外部振荡脉冲信号，用于多片 MCS-51 单片机同时工作。在这次设计中只用一个 MCS-51 单片机，不需要振荡脉冲信号。

## 2.3.4 基于 AT89C51 电容测量系统按键电路



按键是实现人机对话的比较直观的接口,可以通过按键实现人们想让单片机做的不同的工作。键盘是一组按键的集合,键是一种常开型开关,平时按键的两个触点处于断开状态,按下键是它们闭合。键盘分编码键盘和非编码键盘,案件的识别由专用的硬件译码实现,并能产生键编号或键值的称为编码键盘,而缺少这种键盘编码电路要靠自编软件识别的称为非编码键盘。在单片机组成的电路系统及智能化仪器中,用的更多的是非编码键盘。图9就是一种比较典型的按键电路,在按键没有按下的时候,输出的是高电平,当按键按下去的时候,输出的低电平。



图9 按键电路

### 2.3.5 基于 AT89C51 电容测量系统 555 芯片电路

555芯片电路是一种将模拟功能与逻辑功能巧妙结合在同一硅片上的组合集成电路。它设计新颖,构思奇巧,用途广泛,备受电子专业设计人员和电子爱好者的青睐,人们将其戏称为伟大的小IC。1972年,美国西格尼蒂克斯公司(Signetix)研制出Tmer NE555双极型时基电路,设计原意是用来取代体积大,定时精度差的热延迟继电器等机械式延迟器。但该器件投放市场后,人们发现这种电路的应用远远超出原设计的使用范围,用途之广几乎遍及电子应用的各个领域,需求量极大。美国各大公司相继仿制这种电路 1974年西格尼蒂克斯公司又在同一基片上将两个双极型555单元集成在一起,取名为NF556。1978年美国英特锡尔(Intelsil)研制成功CMOS型时基电路ICM555 ICM556,后来又推出将四个时基电路集成在一个芯片上的四时基电路558 由于采用CMOS

型工艺和高度集成，使时基电路的应用从民用扩展到火箭、导弹、卫星、航天等高科技领域。在这期间，日本、西欧等各大公司和厂家也竞相仿制、生产。尽管世界各大半导体或器件公司、厂家都在生产各自型号的555 / 556时基电路，但其内部电路大同小异，且都具有相同的引出功能端。555电路，也称555芯片电路，是一种中规模集成电路。它具有功能强、使用灵活、适用范围宽的特点。通常只要外接少量几个元件，就可构成各种不同用途的脉冲电路以及许多实用电路，如多谐振荡器、单稳态电路及施密特触发器等等。从而能够实现振荡、定时、调光、调压、调速等。555集成电路有双极型和CMOS两种。CMOS型的优点是功耗低、电源电压低、输入阻抗高。对于初学者来说，可以把555电路等效看成一个带放电开关的RS触发器，这个特殊的触发器有两个输入端：阈值端TH可看成是置零端R，要求高电平，触发端R可看成置位端低电平有效。它只有一个输出端Vo，Vo可等效为触发器的Q端。放电端DIS可看成由内部放电开关控制的一个接点，放电开关由触发器的端控制：Q=1时DIS端接地；Q=0时，DIS端悬空。此外，这个触发器还有复位端MR加上低电平(<0.3V)时可使输出为低电平。该特殊的RS触发器有两个输入端，这两个输入端的触发电平要求一高一低，其中置零端R即阈值端TH要求高电平，置位端即触发端则要求低电平，也就是使它们翻转的阈值电压值不同。当VK端不接控制电压时，对TH端(即R端)来讲，大于 $2/3V_{DD}$ 是高电平1，小于 $2/3V_{DD}$ 时是低电平0；而对TR端(即端)来讲，大于 $1/3V_{DD}$ 是高电平1，小于 $1/3V_{DD}$ 是低电平0。

555芯片电路的应用电路很多，如：多个单稳、多个双稳、单稳和无稳，双稳和无稳的组合等。在实际应用中，除了单一品种的电路外，还可组合出很多不同电路。本次设计中应用的电路是直接反馈型无稳类电路。电路如图10所示。

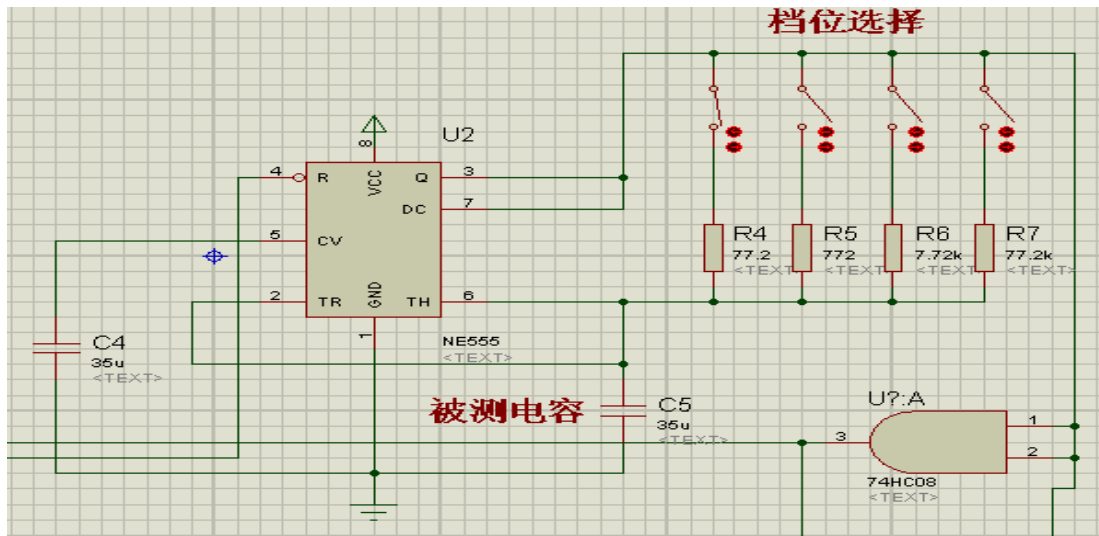


图10 555芯片电路





555 芯片输出的频率为  $f = \frac{0.772}{R \times C}$ ，只要我们改变电阻 R，就可以达到改变电阻量程的目的，图中提供了四组电阻，所以说有四组的电容测量量程，每个量程之间的跨度是 10 倍的关系。

在555芯片输出方波后，由于硬件的原因，输出的方波会有很多毛刺，为了去除这些毛刺本设计中使用了一个两输入与门（74HC08），让信号通过74HC08后会使得输出的波形毛刺减少很多，使单片机的测量结果变得精确。

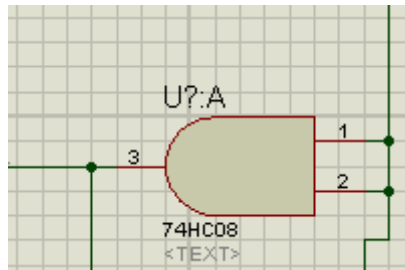


图11 74HC08电路

没有加74HC08之前输出的波形如下

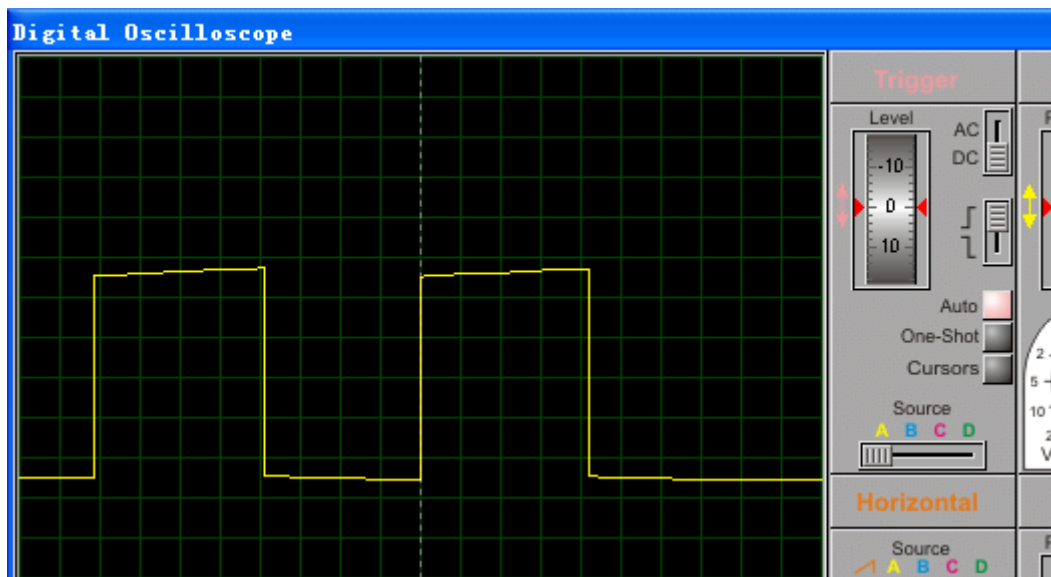


图12 没加74HC08输出波形

加了74HC08之后的输出波形如下

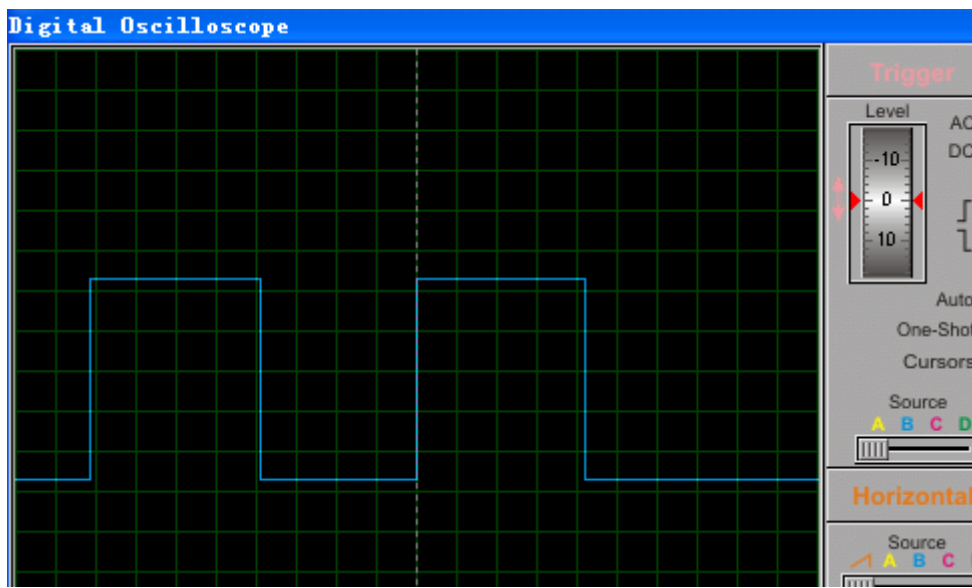


图13 加了74HC08输出波形

二者的比较如下

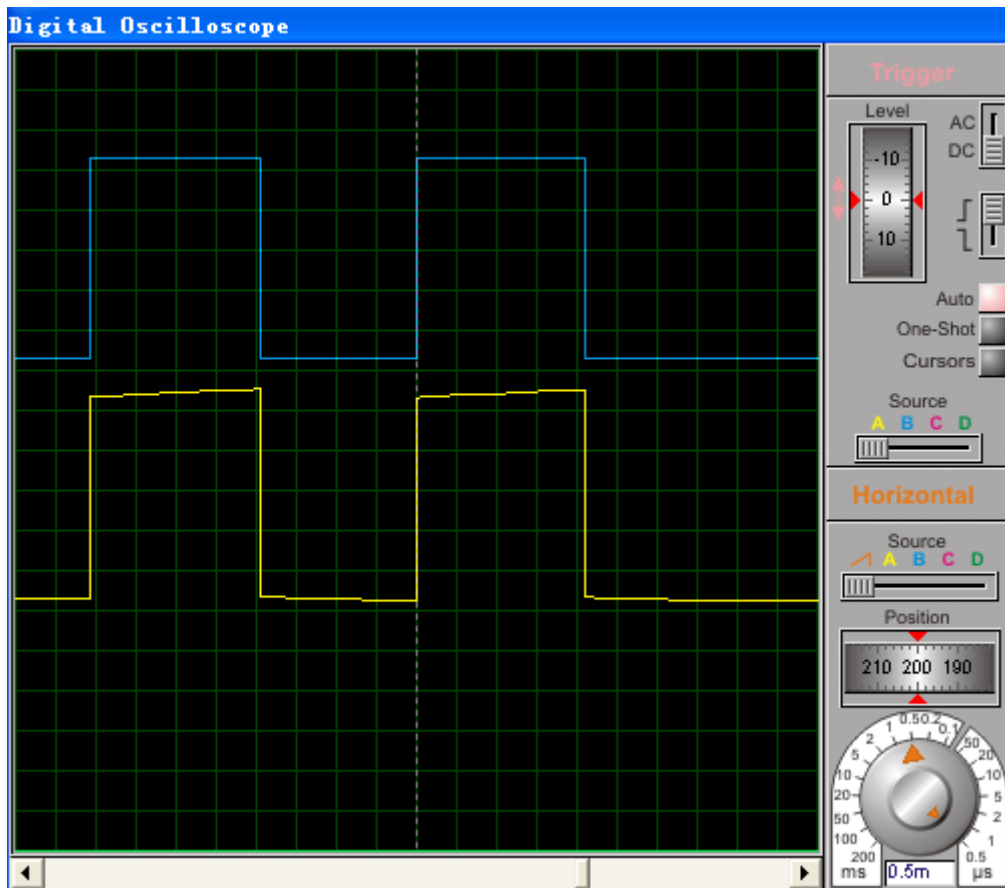


图14 二者的比较

### 2.3.6 基于AT89C51电容测量系统显示电路

LCD 以其低功耗、体积小、显示内容丰富、超薄轻巧的诸多优点，在袖珍式仪表和低功耗应用系统中得到越来越广泛的应用。这里介绍的字符型液晶模块是一种用 5x7 点阵图形来显示字符的液晶显示器，根据显示的容量可以分为 1 行 16 个字、2 行 16 个字、2 行 20 个字等等，这里我们使用的是 2 行 16 个字的 1602 液晶模块。

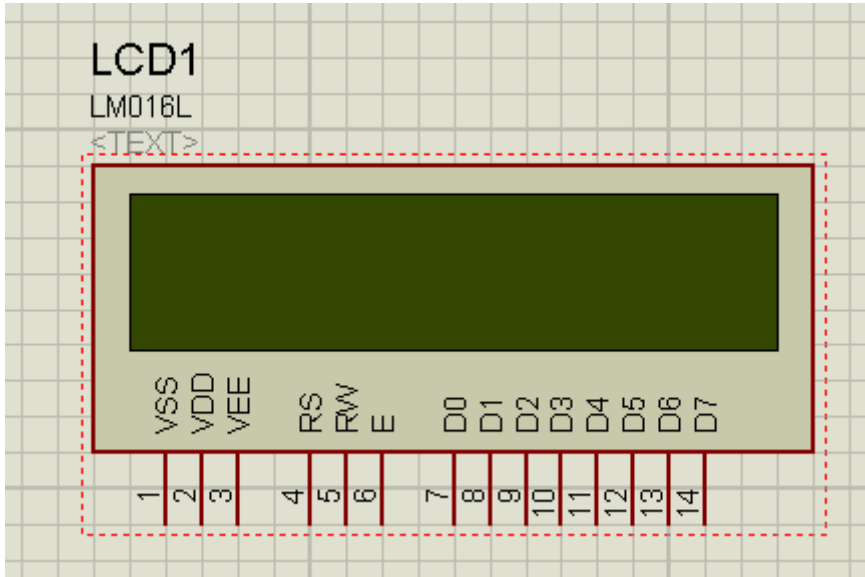


图 15 LCD1602 引脚图

1602 采用标准的 16 脚接口，其中：

第 1 脚：VSS 为地电源

第 2 脚：VDD 接 5V 正电源

第 3 脚：V0 为液晶显示器对比度调整端，接正电源时对比度最弱，接地电源时对比度最高，对比度过高时会产生“鬼影”，使用时可以通过一个 10K 的电位器调整对比度

第 4 脚：RS 为寄存器选择，高电平时选择数据寄存器、低电平时选择指令寄存器。

第 5 脚：RW 为读写信号线，高电平时进行读操作，低电平时进行写操作。当 RS 和 RW 共同为低电平时可以写入指令或者显示地址，当 RS 为低电平 RW 为高电平时可以读忙信号，当 RS 为高电平 RW 为低电平时可以写入数据。

第 6 脚：E 端为使能端，当 E 端由高电平跳变成低电平时，液晶模块执行命令。

第 7~14 脚：D0~D7 为 8 位双向数据线。

第 15~16 脚：空脚

1602 液晶模块内部的字符发生存储器（CGROM）已经存储了 160

个不同的点阵字符图形，如表 1 所示，这些字符有：阿拉伯数字、英文字母的大小写、常用的符号、和日文假名等，每一个字符都有一个固定的代码，比如大写的英文字母“A”的代码是 01000001B (41H)，显示时模块把地址 41H 中的点阵字符图形显示出来，我们就能看到字母“A”

表 1 CGROM 和 CGRAM 中字符代码与字符图形对应关系

高位 低 位	0000	0010	0011	0100	0101	0110	0111	1010	1011	1100	1101	1110	1111
XXXX0000	CGRAM (1)	0	ø	P	\	p		-	夕	三	a	P	
XXXX0001	(2)	!	1	A	Q	a	q	口	ア	チ	ム	ä	q
XXXX0010	(3)	"	2	B	R	b	r	r	イ	川	メ	β	θ
XXXX0011	(4)	#	3	C	S	c	s	」	ウ	ラ	モ	ε	∞
XXXX0100	(5)	\$	4	D	T	d	t	\	エ	ト	セ	μ	Ω
XXXX0101	(6)	%	5	E	U	e	u	ロ	オ	ナ	ユ	B	0
XXXX0110	(7)	&	6	F	V	f	v	テ	カ	ニ	ヨ	P	Σ
XXXX0111	(8)	>	7	G	W	g	w	ア	キ	ヌ	ラ	g	π
XXXX1000	(1)	(	8	H	X	h	x	イ	ク	ネ	リ	f	X
XXXX1001	(2)	)	9	I	Y	i	y	ウ	ケ	」	ル	-1	y
XXXX1010	(3)	*	:	J	Z	j	z	エ	コ	リ	レ	j	千
XXXX1011	(4)	+	:	K	[	k	{	オ	サ	ヒ	ロ	x	万
XXXX1100	(5)	フ	<	L	¥	l		セ	シ	フ	ワ	ℓ	⊂
XXXX1101	(6)	-	=	M	]	m	}	ユ	ス	へ	ソ	セ	+
XXXX1110	(7)	.	>	N	^	n	-	ヨ	セ	ホ	ハ	n̄	
XXXX1111	(8)	/	?	O	-	o	←	ツ	ソ	マ	ロ	Ö	

1602 液晶模块内部的控制器共有 11 条控制指令，其控制命令如下表所示。

表 2 LCD1602 内部 11 条控制指令

指 令		RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0
1	清显示	0	0	0	0	0	0	0	0	0	↓
2	光标返回	0	0	0	0	0	0	0	0	1	*
3	置输入模式	0	0	0	0	0	0	0	1	I/D	S
4	显示开/关控制	0	0	0	0	0	0	1	D	C	B
5	光标或字符移位	0	0	0	0	0	1	S/C	R/L	*	*
6	置功能	0	0	0	0	1	DL	N	F	*	*
7	置字符发生存储器地址	0	0	0	1	字符发生存储器地址(AGG)					
8	置数据存储器地址	0	0	1	显示数据存储器地址(ADD)						
9	读忙标志或地址	0	1	BF	计数器地址(AC)						
10	写数到 CGRAM 或 DDRAM	1	0	要写的数							
11	从 CGRAM 或 DDRAM 读数	1	1	读出的数据							

它的读写操作、屏幕和光标的操作都是通过指令编程来实现的。(说明：1为高电平、0 为低电平)

指令 1: 清显示，指令码 01H, 光标复位到地址 00H 位置

指令 2 : 光标复位，光标返回到地址 00H

- 指令 3: 光标和显示模式设置 I/D: 光标移动方向, 高电平右移, 低电平左移 S: 屏幕上所有文字是否左移或者右移。高电平表示有效, 低电平则无效
- 指令 4: 显示开关控制。D: 控制整体显示的开与关, 高电平表示开显示, 低电平表示关显示 C: 控制光标的开与关, 高电平表示有光标, 低电平表示无光标 B: 控制光标是否闪烁, 高电平闪烁, 低电平不闪烁
- 指令 5: 光标或显示移位 S/C: 高电平时移动显示的文字, 低电平时移动光标
- 指令 6: 功能设置命令 DL: 高电平时为 4 位总线, 低电平时为 8 位总线 N: 低电平时为单行显示, 高电平时双行显示 F 低电平时显示 5x7 的点阵字符, 高电平时显示 5x10 的点阵字符 (有些模块是 DL: 高电平时为 8 位总线, 低电平时为 4 位总线)
- 指令 7: 字符发生器 RAM 地址设置
- 指令 8: DDRAM 地址设置
- 指令 9: 读忙信号和光标地址 BF: 为忙标志位, 高电平表示忙, 此时模块不能接收命令或者数据, 如果为低电平表示不忙。
- 指令 10: 写数据指令 11: 读数据

LCD1602 液晶显示模块可以和单片机 AT89C51 直接接口, 电路如下图所示。

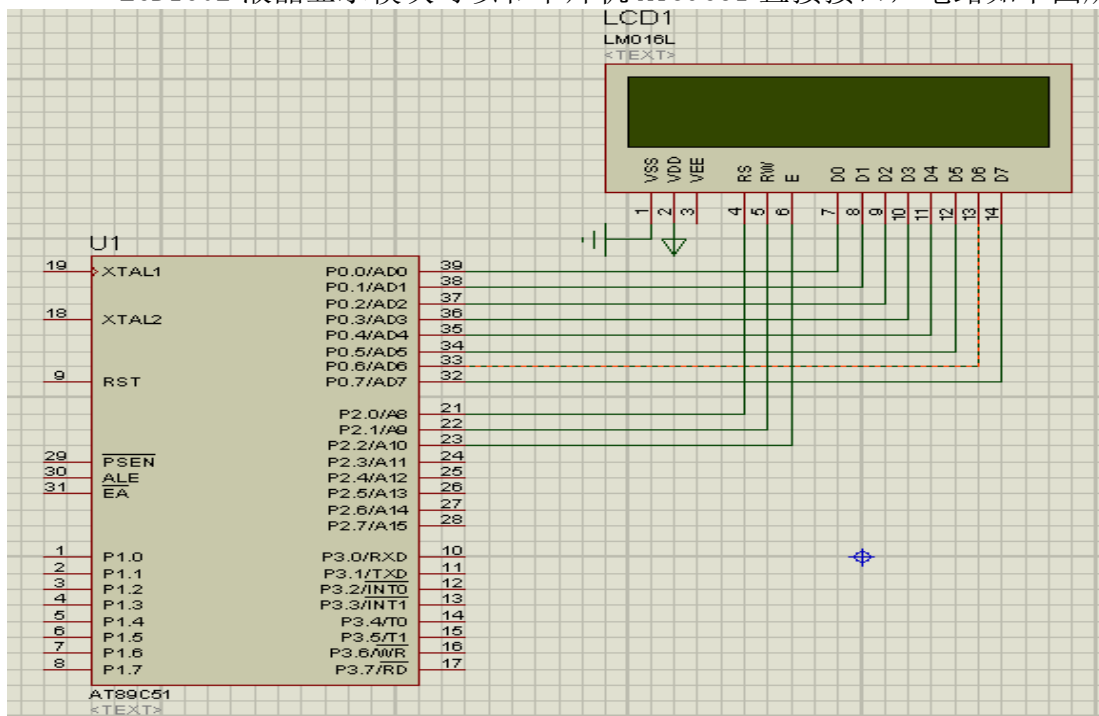


图 16 LCD1602 与 AT89C51 连接图

液晶显示模块是一个慢显示器件，所以在执行每条指令之前一定要确认模块的忙标志为低电平，表示不忙，否则此指令失效。要显示字符时要先输入显示字符地址，也就是告诉模块在哪里显示字符，下表是 LCD1602 的内部显示地址

表 3 LCD1602 内部显示地址

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	第一行
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	第二行

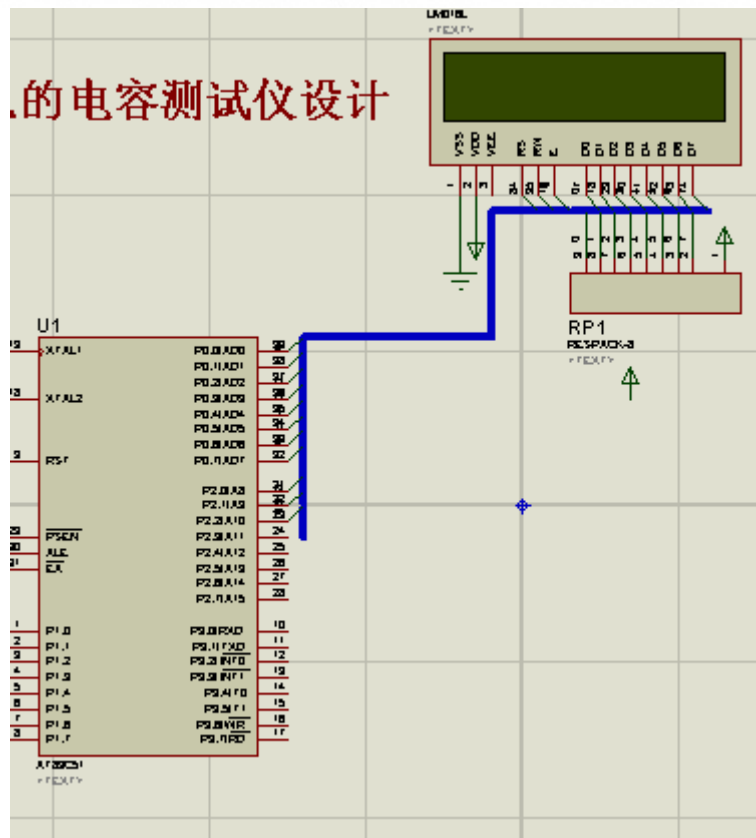


图17 LCD1602显示电路

## 2.4 各部分电路连接成整个电路图

当各个部分的电路设计完成后，下面的工作就是组合成一个总的电路图。现今用的最多的电路设计软件就是 Proteus，Proteus 软件是英国 Labcenter electronics 公司出版的 EDA 工具软件。它不仅具有其它 EDA 工具软件的仿真功能，还能仿真单片机及外围器件。它是目前最好的仿真单片机及外围器件的工具。虽然目前国内推广刚起步，但已受到单片机爱好者、从事单片机教学的教

师、致力于单片机开发应用的科技工作者的青睐。Proteus 是世界上著名的 EDA





工具(仿真软件),从原理图布图、代码调试到单片机与外围电路协同仿真,一键切换到 PCB 设计,真正实现了从概念到产品的完整设计。是目前世界上唯一将电路仿真软件、PCB 设计软件和虚拟模型仿真软件三合一的设计平台,其处理器模型支持 8051、HC11、PIC10/12/16/18/24/30/DsPIC33、AVR、ARM、8086 和 MSP430 等,目前已增加 Cortex 和 DSP 系列处理器,并持续增加其他系列处理器模型。在编译方面,它也支持 IAR、Keil 和 MPLAB 等多种编译器。

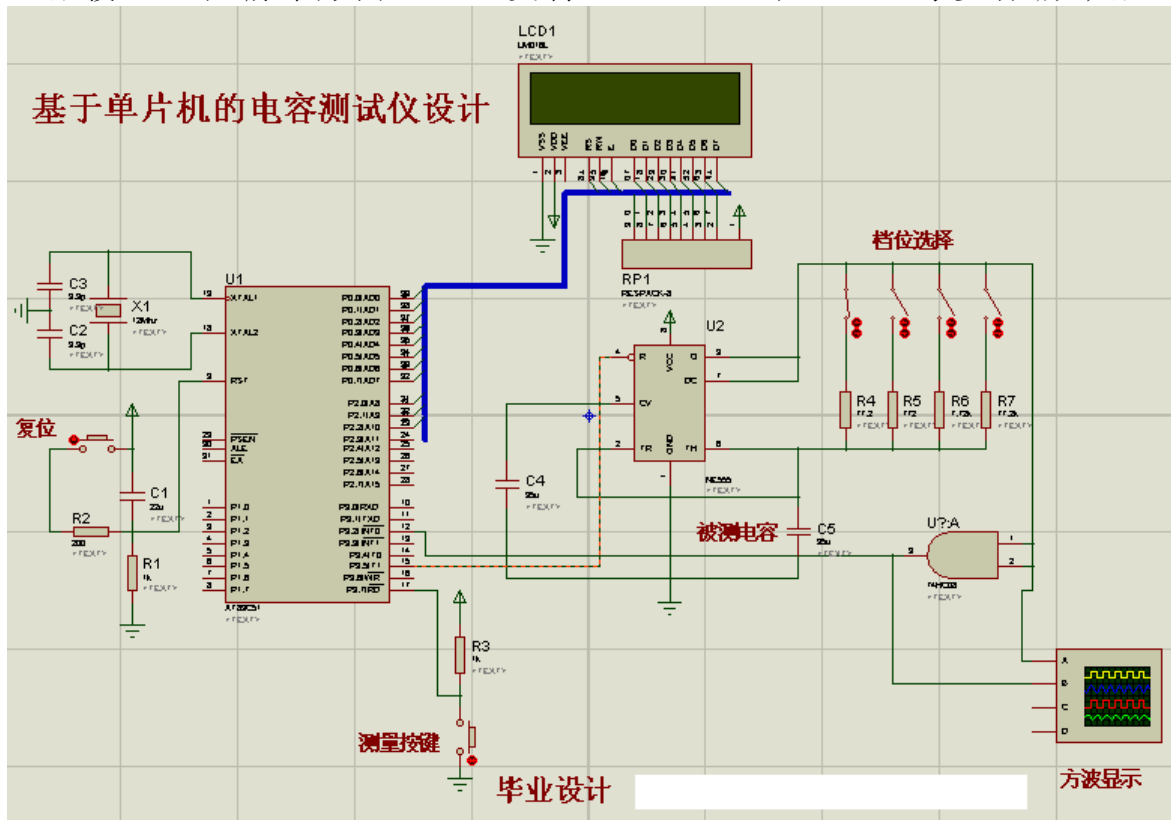


图 18 总电路图

## 3 基于单片机电容测量软件设计

### 3.1 软件设计

软件设计是一个创造性的过程，对一些设计者来说需要一定的资质，而最后设计通常都是由一些初步设计演变而来的。从书本上学不会设计，只能经过实践，通过对实际系统的研究和实践才能学会。对于高效的软件工程，良好的设计是关键，一个设计得好的软件系统应该是可直接实现和易于维护、易懂和可靠的。设计得不好的系统，尽管可以工作，但很可能维护起来费用昂贵、测试困难和不可靠，因此，设计阶段是软件开发过程中最重要的阶段。

直到最近，软件设计在很大程度上仍是一个特定过程。一般用自然语言给定一个需求集，预先作非正式设计，常常用流程图的形式说明，接着开始编码，当系统实现时设计还需修改。当实现阶段完成后，设计往往已与起初形式相去甚远以至于设计的原始文档完全不适合对系统的描述。

### 3.2 软件设计任务

软件设计主要是针对硬件设计里面的控制部分的，这里指AT89C51单片机，一般的单片机均可用汇编语言和C语言进行编程。C语言直观，相对比较的简单，但占用的程序存储器的内存比较大，汇编语言是针对硬件设计的语言，如果想用汇编语言设计的话必须要对硬件有很大的了解，相对C语言就比较的复杂，但是比较的精简，占用的程序存储器的空间比较小。作为还在学生阶段的我们，用汇编语言进行编程对我们理解单片机的内部结构、资源都有很好的帮助，但是C语言编程在以后的实际工作中将会大大简化自己的工作。因此本论文的程序都是基于C语言的。要完成的任务是初始化程序设计、按键程序设计、中断处理程序，计数器计数程序，显示程序设计等。

### 3.3 软件设计的工具

本次毕业设计所选用 Keil C51 中的编译/连接器软件 Keil uVision2 作为编译器/连接工具。

### 3.4 程序设计算法设计

整个程序设计过程中遇到的最大的问题的如何根据测量到的方波的频率来计算所测量的电容的大小。在前面的介绍中我们知道：555

时基芯片的输出频率跟所使用的电阻  $R$  和电容  $C$  的关系是：
$$f = \frac{0.772}{R \times C}$$

又因为  $T = \frac{1}{f}$ ，所以

$$T = \frac{R}{0.772} * C \tag{7}$$

即：

$$C = T * \frac{0.772}{R} \tag{8}$$

如果单片机采用 12M 的晶振，计数器 T0 的值增加 1，时间就增加  $1 \mu S$ ，我们采用中断的方式来启动和停止计数器 T0，中断的触发方式为脉冲下降沿触发，第一次中断到来启动 T0，计数器的值为  $N_1$ ，第二次中断到来停止 T0，计数器的器的值为  $N_2$ ，则测量方波的周期为  $T = (N_2 - N_1) * 1\mu s$ ，如何开始时刻计数器的值  $N_1 = 0$ ，则  $T = N * 10^{-6}$ 。

简单时序图如下。

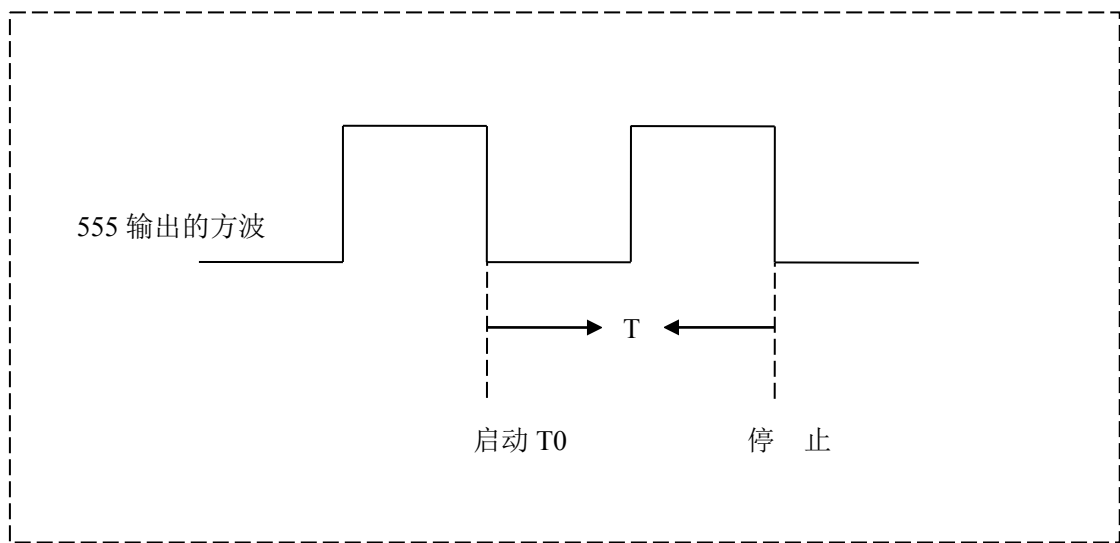


图 19 时序图

则：
$$C = N * \frac{0.772}{R} * 10^{-6} \tag{9}$$

单片机的计数器的值  $N=0-65535$ ，为了测量的精度， $N$  的取值一般在 100~5000，当电阻  $R$  越大，电容  $C$  的值就越小。我们取不同的电阻值，就得到不同的电容测量的量程。

第一档:	$R_1 = 77.2\Omega$	$C = \frac{N}{100} * 10^{-6}$	$C = \frac{N}{100} * 1\mu F$	1~50uF
第二档:	$R_2 = 772\Omega$	$C = \frac{N}{100} * 10^{-1} * 10^{-6}$	$C = \frac{N}{100} * 10^{-1} * 1\mu F$	0.1~5 uF
第三档:	$R_3 = 7.72K\Omega$	$C = \frac{N}{100} * 10^{-2} * 10^{-6}$	$C = \frac{N}{100} * 10^{-2} * 1\mu F$	0.01~0.5 uF
第四档:	$R_4 = 77.2K\Omega$	$C = \frac{N}{100} * 10^{-3} * 10^{-6}$	$C = \frac{N}{100} * 10^{-3} * 1\mu F$	0.001~0.05 uF

为了编写程序的方便，我们只计算  $\frac{N}{100}$ ，后面的单位可以根据使用的量程自行添加。测量范围的大小 0.001uF~655.35uF。

### 3.5 软件设计流程

流程图是一种传统的算法表示法，它利用几何图形的框来代表各种不同性质的操作，用流程线来指示算法的执行方向。由于它简单直观，所以应用广泛，特别是在早期语言阶段，只有通过流程图才能简明地表述算法，流程图成为程序员们交流的重要手段。

本次毕业设计在软件设计方面的难度是有点大。图 20 为整个程序设计的流程。

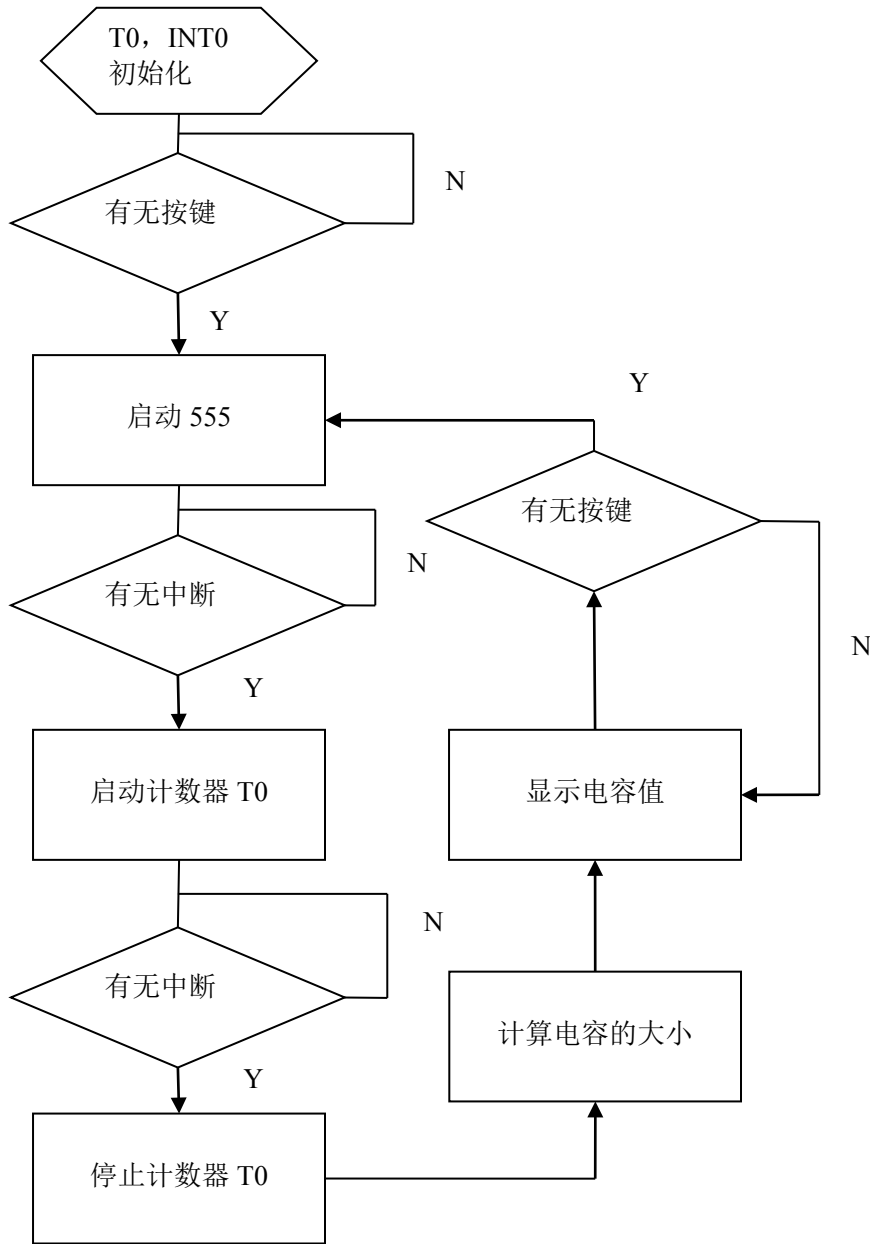


图 20 程序设计流程图

### 3.6 编写程序

有了前面的对整个系统的分析后画出的流程图，下面的工作就是根据流程图编写程序。编写程序是一个相当复杂的过程，要求编程人员具有很强逻辑思维，而且要在对整个系统工作原理相当熟悉的基础上才能完成任务。在编写程序的时候要养成作注释的习惯，这样既利于自己以后的修改，又利于以后程序的维护。尤其是使用C语言编写程序的时候更是要注释，因为C语言是一门比较高级的语言，C语言比较的烦琐。当然编写程序遇到困难的时候还需要很大技巧。

根据上面的流程图，编写程序：

```
#include "reg51.h"

#include"intrins.h"//库函数

#define DATA P0

sbit RW=P2^1;//1602 写数据

sbit RS=P2^0;//1602 写地址

sbit EN=P2^2; //1602 工作使能

sbit b_test=P3^7; //开始测量电容的按键输入

sbit _reset=P3^5; //555 时基芯片工作控制信号

unsigned int T_flag,N,C,i,Dis1,Dis0;

unsigned int b[6]={0X13,0X0D,0X00,0X00,0X25,0X16}; //显示 C=00UF

/*****延时 1MS*****/

void Delay1ms(unsigned int mm)
{unsigned int i;
  for(mm;mm>0;mm--)
    for(i=100;i>0;i--);
}

/*****检查忙否*****/

void Checkstates()
{
  unsigned char dat;

  RS=0;

  RW=1;

  do{EN=1;//          下          降          沿
```

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/468036070042006057>