

项目名称： 低频信号发生器

项目成员： 周新凯 陈璞 李丹丹

低频信号发生器设计与总结报告

摘要: 本设计采用两片 STC89S52RC 作为波形产生和输出电压采集显示的控制核心。系统通过 D/A 转换器 (DAC0832) 将数字信号转换成模拟信号, 以产生正弦波、左锯齿波、右锯齿波、三角波、方波 5 种波形, 并可通过按键来选择 5 种波形的类型和频率, 在液晶屏 LCD1602 上显示其各自的类型和频率值。然后经过 RC 无源滤波, 最终由示波器检测。同时, 输出电压经过二极管和 RC 网络采集其峰值后, 通过 A/D 转换器 (ADC0804) 将模拟信号转换成数字信号, 并用数码管显示电压值。

程序设计采用 keil c51 在 STC89S52 系列单片机上进行编程实现, 经测试, 可选择波的类型, 可调节频率, 可调节幅值, 输出波形比较稳定, 可作为测试功放电路等性能指标的简易输入源。

关键字: 低频信号发生器 单片机 AD DA

项目组成员及工作任务分配				
序号	班 级	姓 名	工作内容	联系方式
1	光信 112	周新凯 (组长)	画原理图, 焊接电路板, 修改程序	18710629437
2	光信 111	陈璞	编写程序	15209187958
3	仪 114	李丹丹	完成实验报告	15202420206
2013 年 6 月 2 日				

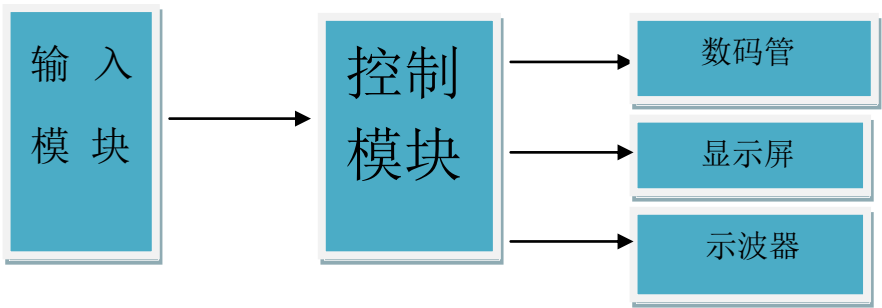
目录

1. 方案设计与论证
 - 1.1 方案设计
 - 1.2 方案论证
2. 系统原理框图
3. 单元电路设计
 - 3.1 最小系统
 - 3.2 D/A 转换模块
 - 3.3 A/D 转换模块
 - 3.4 显示模块
 - 3.5 集成运放及滤波模块
 - 3.6 按键模块
4. 软件设计框图
 - 4.1 波形产生的设计
 - 4.2 波频率的设计
 - 4.3 液晶屏 1602 显示波类型和频率的设计
5. 系统测试
 - 5.1 波形的检测
 - 5.2 波类型和频率的显示
 - 5.3 输出电压峰值的显示
6. 缺点总结
7. 附录 程序清单

1. 方案设计与论证

1.1 方案设计

此方案的控制部分采用的是 AT89C52 单片机，输出部分是经过 D/A 转换之后，经过集成运放放大电压，在连接上示波器，通过编程可以控制波形的输出。



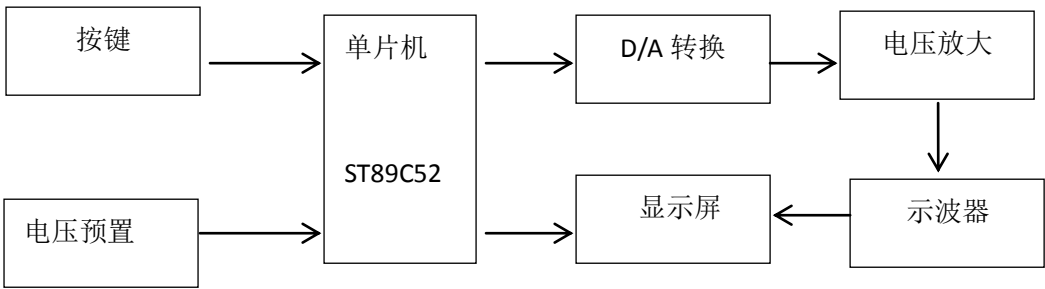
1.2 方案论证

数控部分：采用单片机 AT89C52 完成整个数控部分的功能，并且可以在单片机中编程，有利于系统功能的实现。

输出部分： 经过 D/A 转换，利用集成运放电压放大器放大信号。

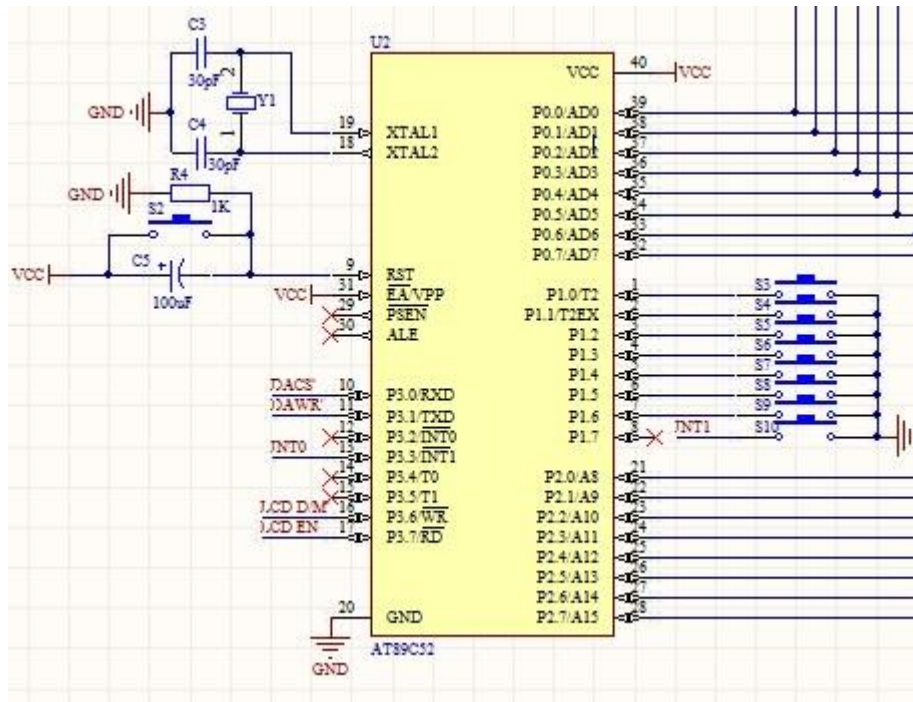
显示部分：利用示波器显示波形，波形的名称通过 LCD1602 显示屏显示。

2. 系统原理框图

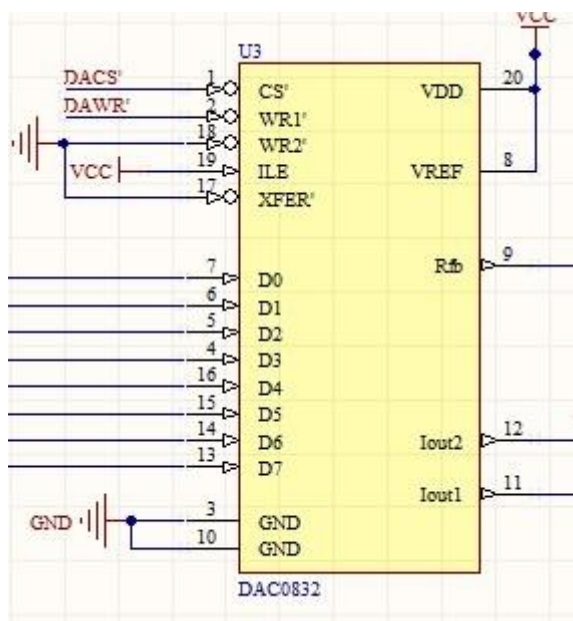


3. 单元电路设计

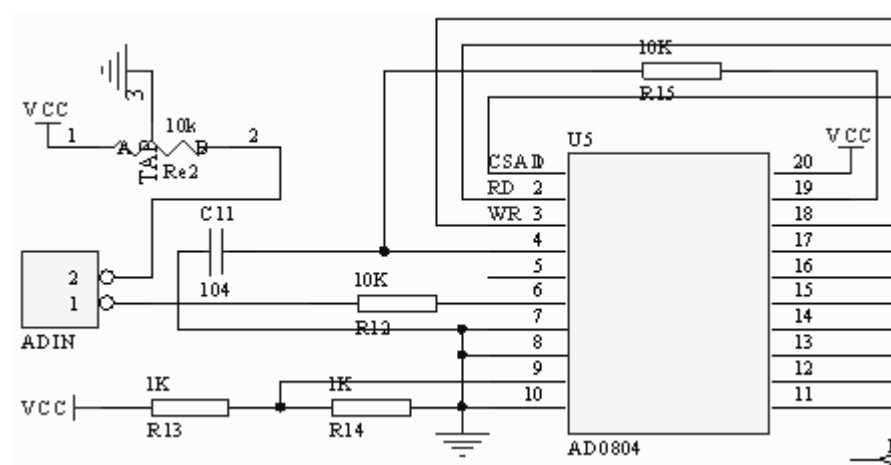
3. 最小系统



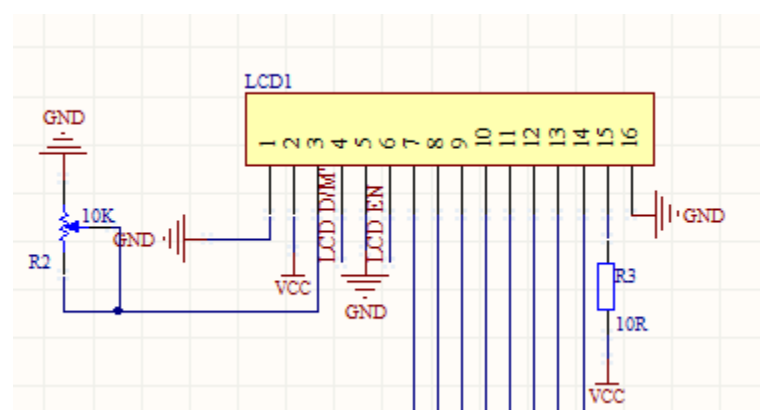
3.2 D/A 转换



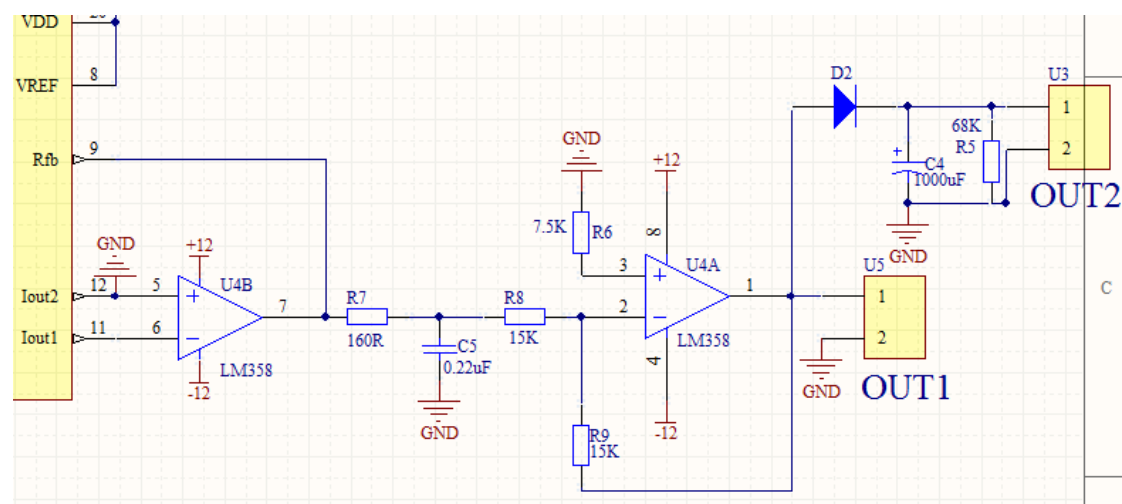
3.3 A/D 转换模块



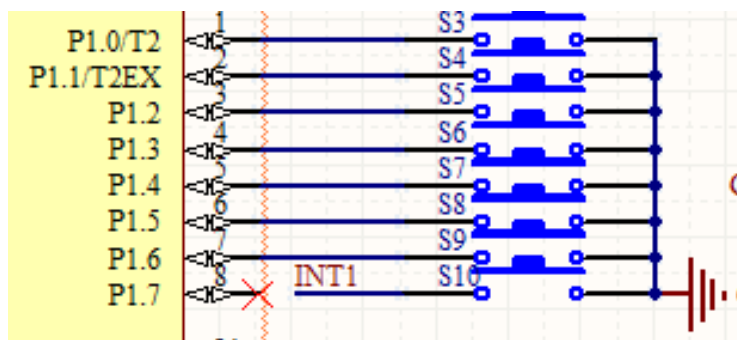
3.4 显示模块



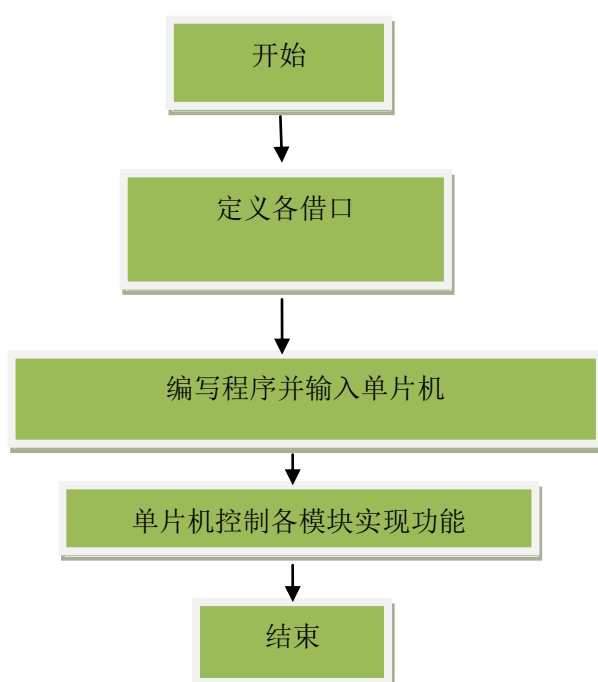
3.5 集成运放及滤波模块



3.6 按键模块



4. 软件设计框图



4.1 波形产生的设计

定义各个单片机的接口，在程序中自行定义一个 rom 表，中断每发生一次，程序查 rom 表，给出一个电压值，多次查表后，形成波形，通过电压调节波形的幅值。

4.2 波形频率的设计

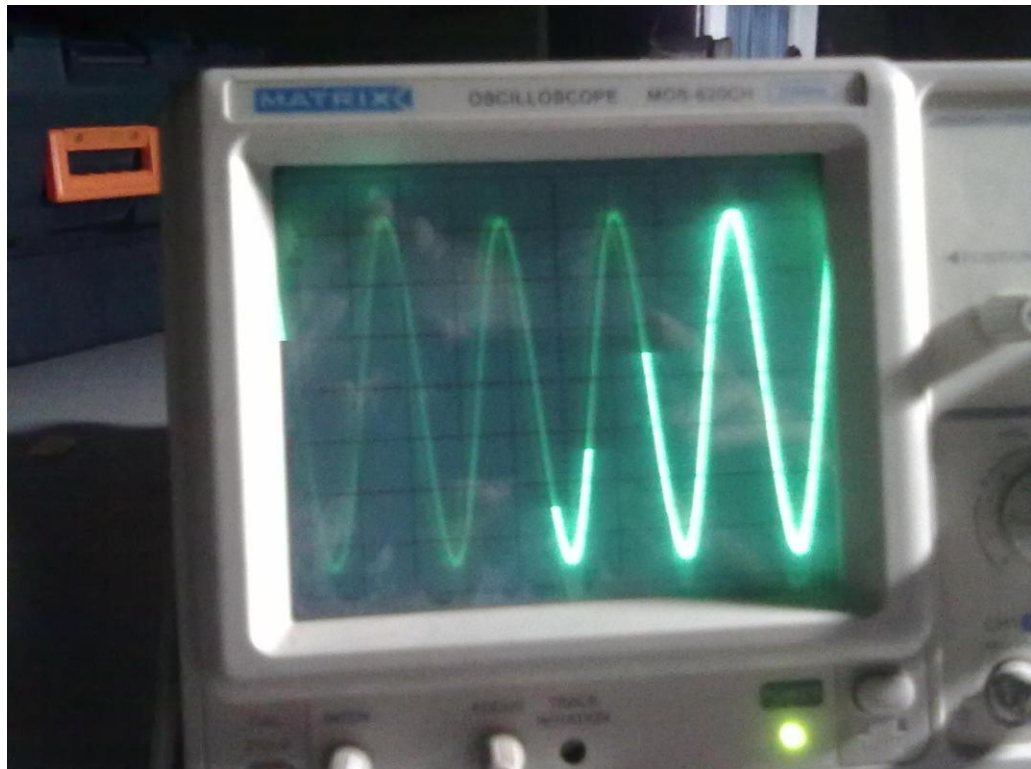
在实现输出波形之后，利用按键模块改变定时器初值可以改变波形的频率。

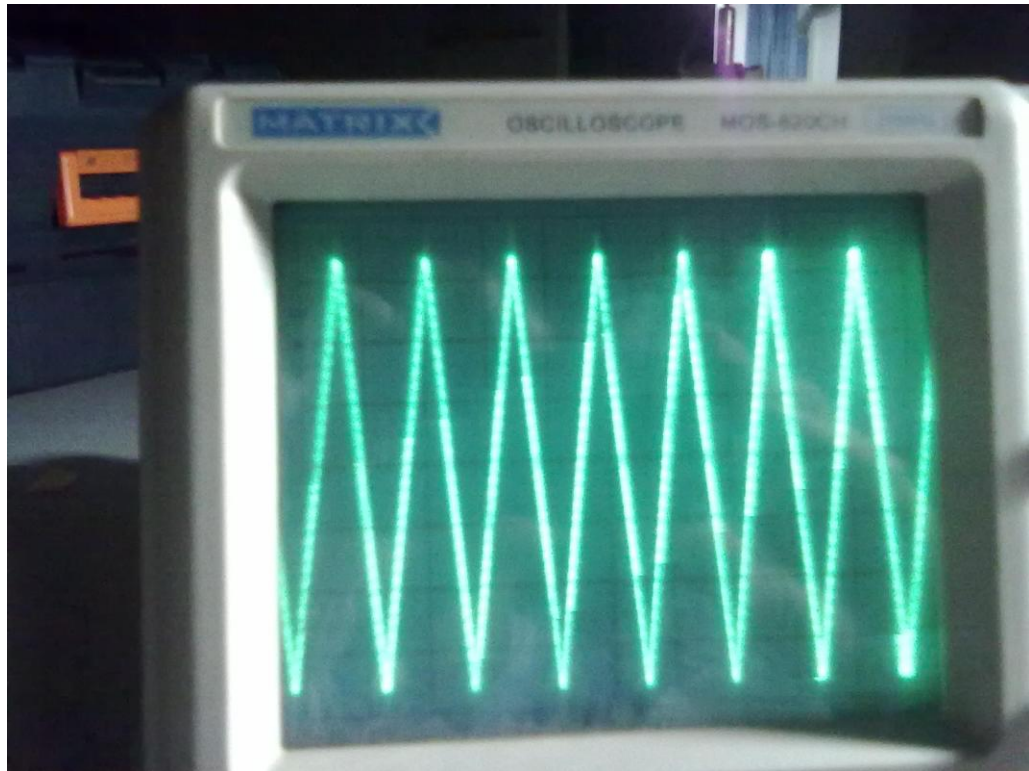
4.3 液晶屏 1602 显示波类型和频率的设计

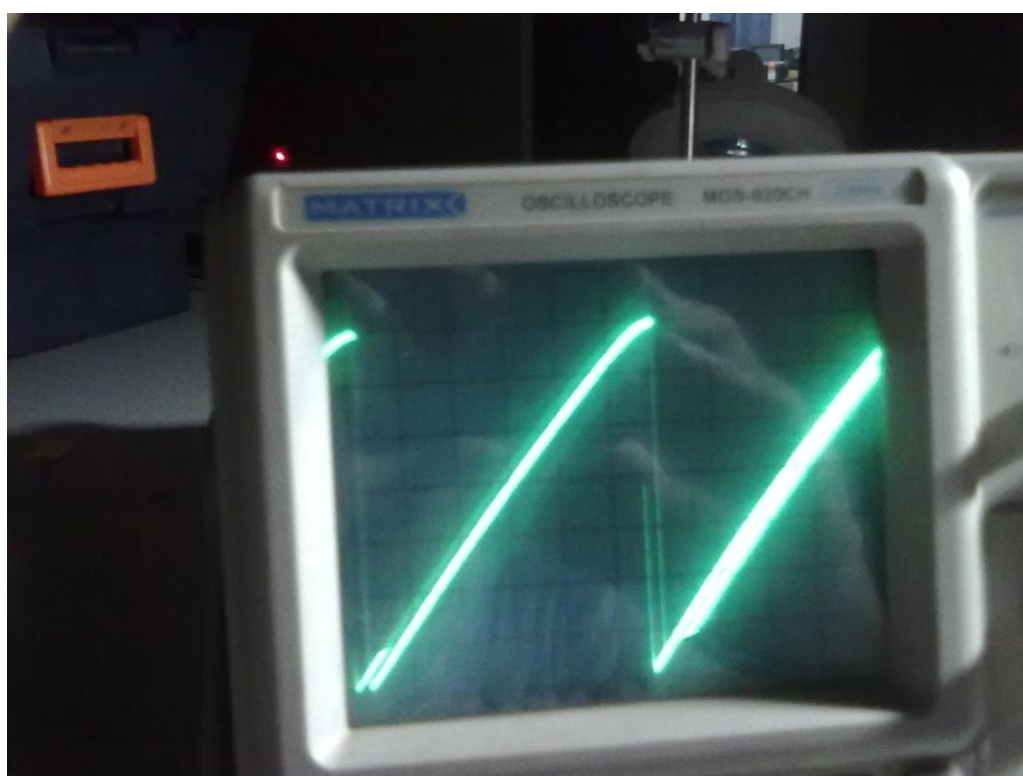
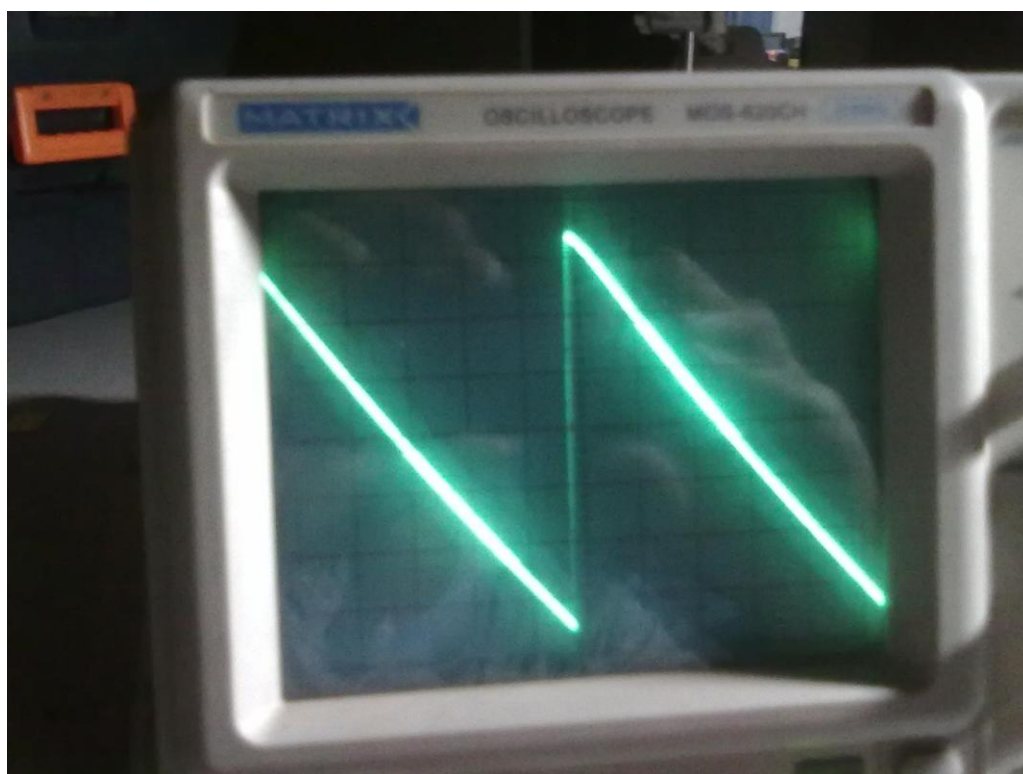
完成波形输出之后，给定频率，液晶屏上可以显示出波形的类型以及频率。

5. 系统测试

5.1 波形的检测



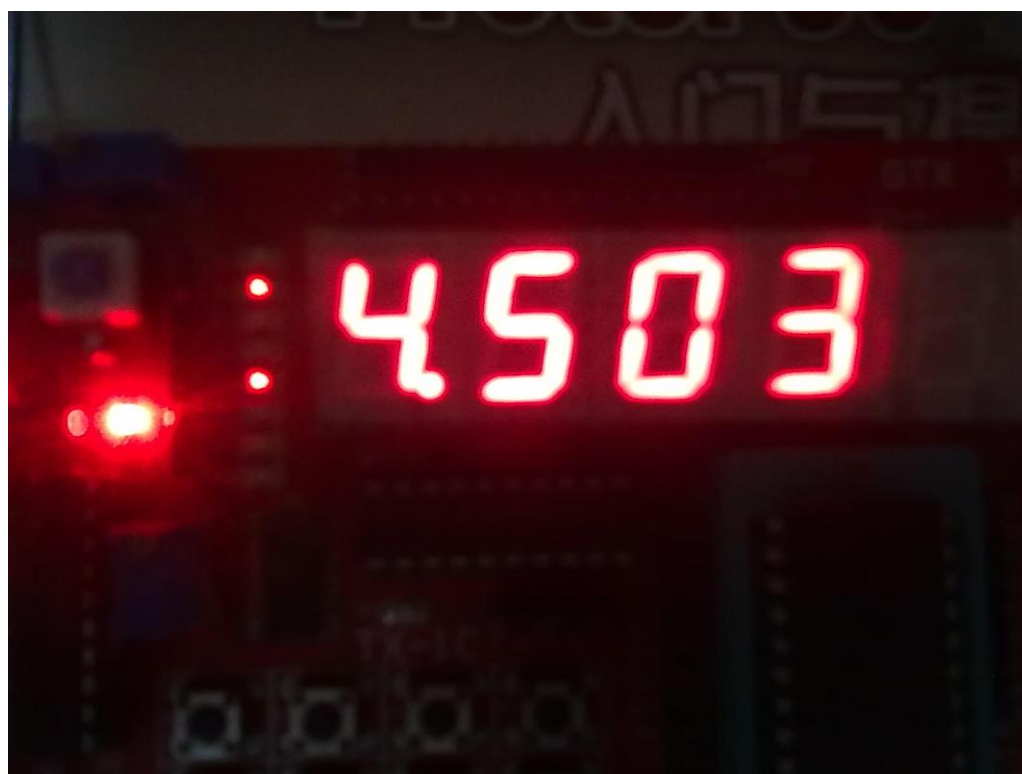




5.2 波类型和频率的显示



5.3 输出电压峰值的显示



6. 缺点总结

这次做的这个信号发生器，基本功能已实现，缺点主要有以下 4 个。

6.1 虽然频率可调，但是由于能力的问题，我们做出的只是离散的调频率，且由于电路本身的原因频率范围比较低，是低频信号发生器。

6.2 虽然幅值可调，但范围只在 0 到 5v 之间，且幅值太大太小都会出现失真。

6.3 滤波采用的是无源滤波，当带上负载后会对波形产生影响，且采集回来的电压由于频率变化的原因不一定是准确的峰值。

6.4 对于方波不能调节其占空比，对于三角波和锯齿波不能调节其斜率。

7. 附录 程序清单

程序也没啥亮点，中断用的是定时器的方式二：8 位初值自动重装，按键检测没用传统的延时去抖的方法，而是用的以下标志位的方法：

```
void keyscan ()
{
    if(s1&& s2&& s3&& s4&& s5&& s6&& s7)
    {
        key=0;
        key_push=0;
    }
    if (!s1&&!key_push)
    {
        key_push=1;
        s1_push=1;
        s2_push=0;
        s3_push=0;
        s4_push=0;
        s5_push=0;
    }
}
```

```

if (!s2&&!key_push)
{
    key_push=1;
    s2_push=1;
    s1_push=0;
    s3_push=0;
    s4_push=0;
    s5_push=0;
}
if (!s3&&!key_push)
{
    key_push=1;
    s3_push=1;
    s2_push=0;
    s1_push=0;
    s4_push=0;
    s5_push=0;
}
if (!s4&&!key_push)
{
    key_push=1;
    s4_push=1;
    s2_push=0;
    s3_push=0;
    s1_push=0;
    s5_push=0;
}

if (!s5&&!key_push)
{
    key_push=1;
    s4_push=0;
    s2_push=0;
    s3_push=0;
    s1_push=0;
    s5_push=1;
}

```

```
    if(!s6&&!key)
        {
            key=1;
            s6_push=1;
        }
    else s6_push=0;
    if(!s7&&!key)
        {
            key=1;
            s7_push=1;
        }
    else s7_push=0;
}
```