



# 海洋信息工程学院

College of Marine Information Engineering

# 电子系统设计训练

# 报 告

## 设计题目 超声波液位测距系统设计

小组成员姓名： 黄昊林， 黄惠齐， 何文涛

小组成员学号： S20244015 S20221101 S2022110

8

年 级 专 业： 20 级电子信息科学与技术

任 课 教 师：龙顺宇

完成日期： 2023 年 05 月 26 日

# 目录

|                                   |   |
|-----------------------------------|---|
| 1. 实验功能要求 .....                   | 1 |
| 2. 实验所用传感器原理 .....                | 1 |
| 2.1 工作原理 .....                    | 1 |
| 2.2 测距原理 .....                    | 1 |
| 2.3 超声波液位测距原理分析图 .....            | 2 |
| 3. 实验电路 .....                     | 3 |
| 3.1 超声波测距实验仿真图 .....              | 3 |
| 3.2 超声波传感器内部电路原理图 .....           | 3 |
| 4. 实验过程 .....                     | 4 |
| 5. 实验结果及系统流程图 .....               | 5 |
| 6. 小组所有成员的分工说明；所有成员的实验心得及得分 ..... | 7 |
| 6.1 小组成员分工 .....                  | 7 |
| 6.2 小组成员实验心得 .....                | 7 |
| 6.3 小组成员得分 .....                  | 8 |
| 7. 代码实现 .....                     | 9 |

# 超声波液位测距系统设计

## 1. 实验功能要求

使用超声波测距模块与 STC89C51 单片机进行通信，将超声波传感器模块测得的数据送到单片机进行处理。然后单片机将超声波传感器模块传来的数据进行一定运算之后，将测得的液位的距离显示到 LCD1602 上面，如果超出了设置的超声波的液位测距范围，在 LCD1602 上显示液位距离，单片机将会控制蜂鸣器进行滴滴的报警声并排出液体，否则实时显示当前与被测液位的距离。

## 2. 实验所用传感器原理

### 2.1 工作原理

超声波传感器主要由发送部分、接收部分、控制部分和电源部分构成。其中，发送部分由发送器和换能器构成，换能器可以将压电晶片受到电压激励而进行振动时产生的能量转化为超声波，发送器将产生的超声波发射出去；

接收部分由换能器和放大电路组成，换能器接收到反射回来的超声波，由于接收超声波时会产生机械振动，换能器可以将机械能转换成电能，再由放大电路对产生的电信号进行放大；

控制部分就是对整个工作系统的控制，首先控制发送器部分发射超声波，然后对接收器部分进行控制，判断接收到的是否是由自己发射出去的超声波，最后识别出接收到的超声波的大小；

电源部分就是整个系统的供电装置。这样，在电源作用下、在控制部分控制下，发送器与接收器两者协同合作，就可以完成传感器所需的功能。

### 2.2 测距原理

超声波液位测距的原理十分简单，由超声波的发射端发射一束超声波，在发射的同时，计时开始，发射出去的超声波在介质中传播，声波具有反射特性，当遇到障碍物时就会反射回来，当超声波的接收端接收到反射回来的超声波时，计时停止。介质为空气时，声速为 340m/s，根据记录的时间  $t$ ，利用公式计算出发射位置与障碍物之间的距离：

$$s = 340t / 2$$

这就是所谓的时间差测距法。

超声波测距的原理就是已知超声波在介质中的传播速度，测量出从发射到接收所需

的时间，根据测量出的时间来计算出障碍物的距离。因此，超声波测距的原理与回声定位是一样的。

测距的公式如式所示：

$$L = C \times T$$

式中  $L$  为测量的距离长度； $C$  为超声波在介质中的传播速度； $T$  为测量出传播时间的一半。由于超声波的波长相对较短，具有良好的方向性和穿透能力，在用作测量时具有很高的精度，但是仍然有一些因素可以让超声波测距产生误差。我们总结了超声波测距的原理，根据数学几何关系， $H$  为总高度，测量距离  $L$  表示为  $L = S \cos \alpha$ ，超声波探头与液面的夹角  $\alpha$  表示为  $\alpha = \arcsin(X/S)$ ，超声波传播距离  $S$  表示为  $S = 1/2vt$ ，由此推出  $L = 1/2vt \cos[\arcsin(X/S)]$ 。当测量距离  $L$  远大于  $X$  时，则可近似为  $L = 1/2vt$ 。最终，液面高度为  $h = H - L$ 。

### 2.3 超声波液位测距原理分析图

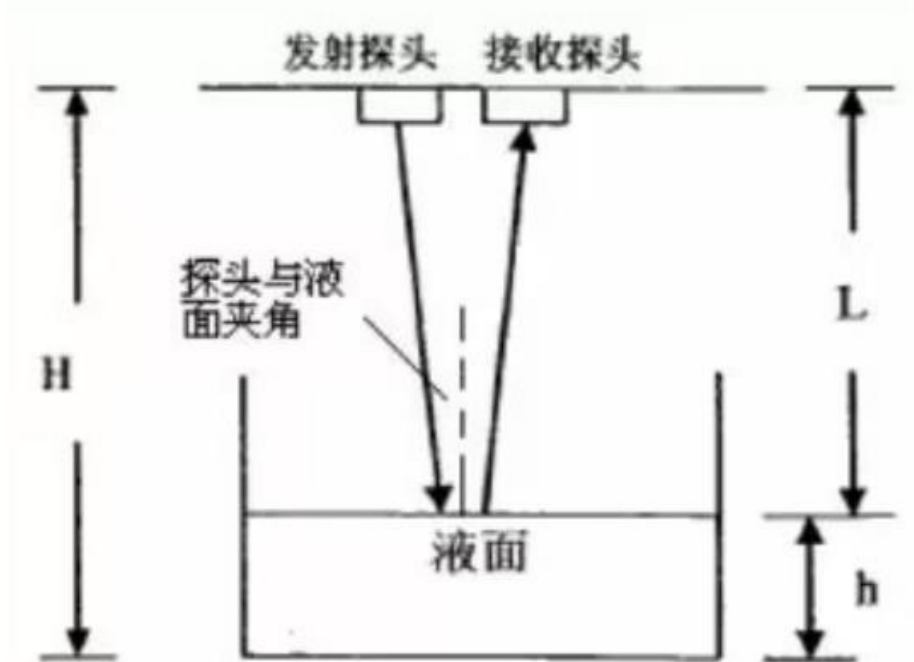


图1 原理分析图



## 4. 实验过程

①我们经过讨论后考虑到材料和技术等因素，我们将单片机由 c52 改为 c51 其中可编程控制芯片 STC89C51，由中央处理器(CPU)、存储器和/I/O 连接器组成。单片机作为驱动实现系统在此设计中占据主导地位。STC89C51 单片机配置 MCS-51 内核，具备损耗小、性能完善等显著特点，在传统单片机基础之上，功能更加强大且智能化，广泛用于多种工业设计。将编写好的 C 语言程序下载到单片机，由电路模块实现系统功能。在工作可靠性的要求下能做到降低成本，此外，STC89C51 分析处理能力强且可实现在线编程，能提高我们的工作效率 Error! Reference source not found.，在进行实验当中，我们首先学习如何创建 STC89C51 单片机的工程。在经过不断的学习之后我们小组终于学会了创建单片机的工程，并且在工程当中添加自己的文件，如何就在文件上面编写的程序。我们本次首先是控制蜂鸣器按照不同的频率发出滴答滴答的声音，这个相对来说还是比较简单的。在完成了这一步之后，我们小组下一步的学习就是将 LCD1602 液晶使用起来。因为我们到时候将超声波测回来的数据将其显示在显示屏上面，LCD1602 对于我们来说还是比较困难的。

②首先它的工作电压要求在 4.5-5V 区间，开发板上面的供电一旦低于 4.5V 那显示屏将不能正常工作，所以最好将它的工作电压稳定在 5V。除此之外，它有一个 EN 使能引脚，对它写 1 则液晶进行工作，否则液晶不工作。R/W 为读/写命令选择端，高电平的时候进行读操作，低电平的时候进行写操作。R/S 为数据命令选择端，高电平的时候为数据选择端，低电平的时候为命令选择端。LCD1602 的传输数据引脚为 (D0-D7)，这八个引脚接 STC89C51 的 P0 口，并且 P0 口有外接的上拉电阻，能够驱动液晶传输数据。一开始我们先让液晶显示一个字符，后来再让液晶显示固定的字符串，在完成了这两步之后，就开始使用超声波测距了。

③超声波模块只有四个引脚。一个是 VCC（直流 5V），一个是 GND，还有另外两个分别是 Trig（控制端）接单片机的 P2<sup>1</sup> 引脚，Echo（接收端）接单片机的 P2<sup>0</sup> 引脚。了解了这些工作原理之后，我们在程序上使用单片机的定时器 0 来进行计数，一旦触发中断就计算被测物体的距离，并将计算得到的距离赋给一个变量，然后将这个变量的值显示到液晶，这就是本次实验的过程。

## 5. 实验结果及系统流程图

在本次实验当中，我们实现了将超声波传感器所测得的数据送到单片机进行处理，经过一定的换算之后将距离实时显示到了 LCD1602 上面，并且设置了测距的阈值。在本次设置当中我们的测量范围是 0-7 米，低于 0.1 米或大于 0.2 米蜂鸣器将会一直发出滴答滴答的响声，测距的距离在大于 7 米的时候超出测量范围，屏幕显示“-.-”，而蜂鸣器不会停止发声；理想的液位高度为 0.1-0.2 米，测量值在此区间内蜂鸣器停止发声。但是这个超声波测距模块有一个缺点，那就是当被测距离非常近的时候，超声波传感器测不出来，它会显示超出了测量范围。

实验结果图如下所示：

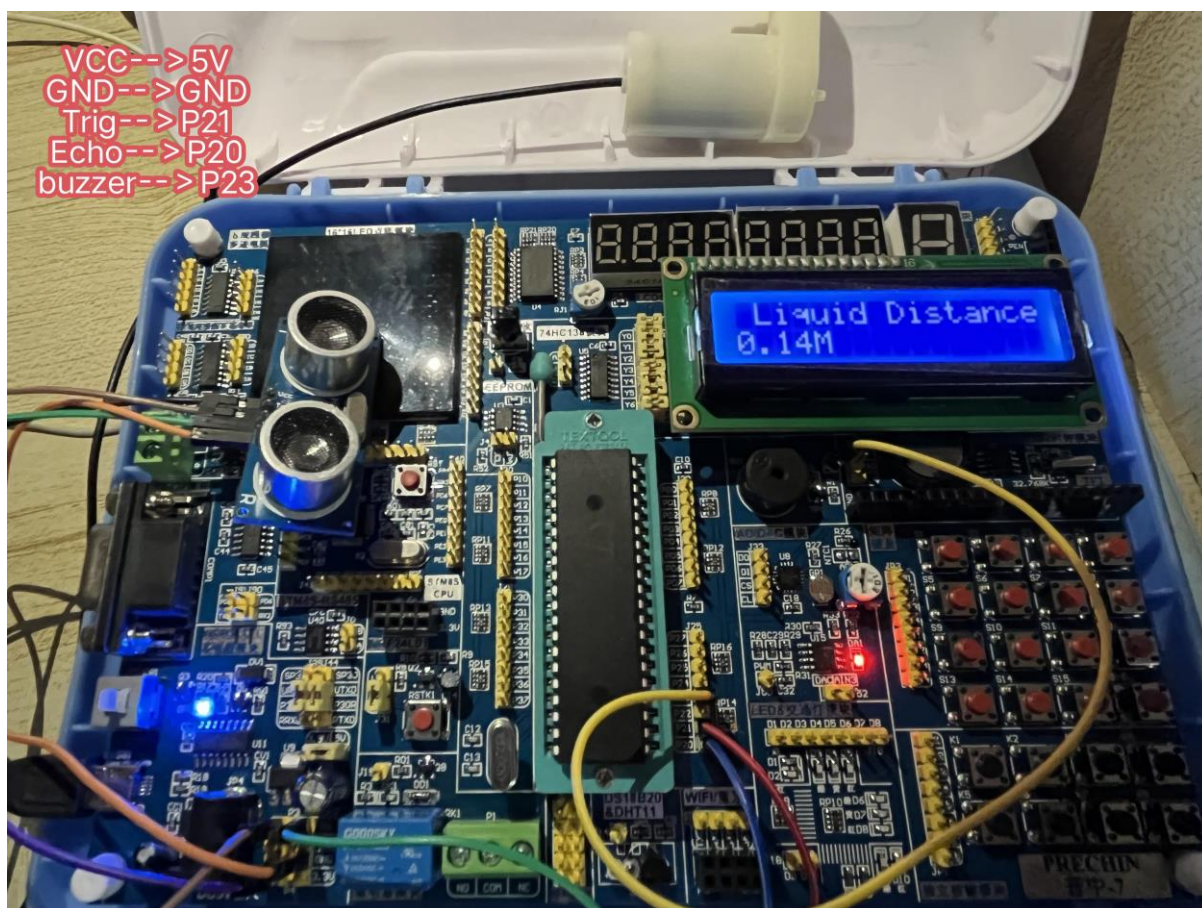


图 4 连接及显示图



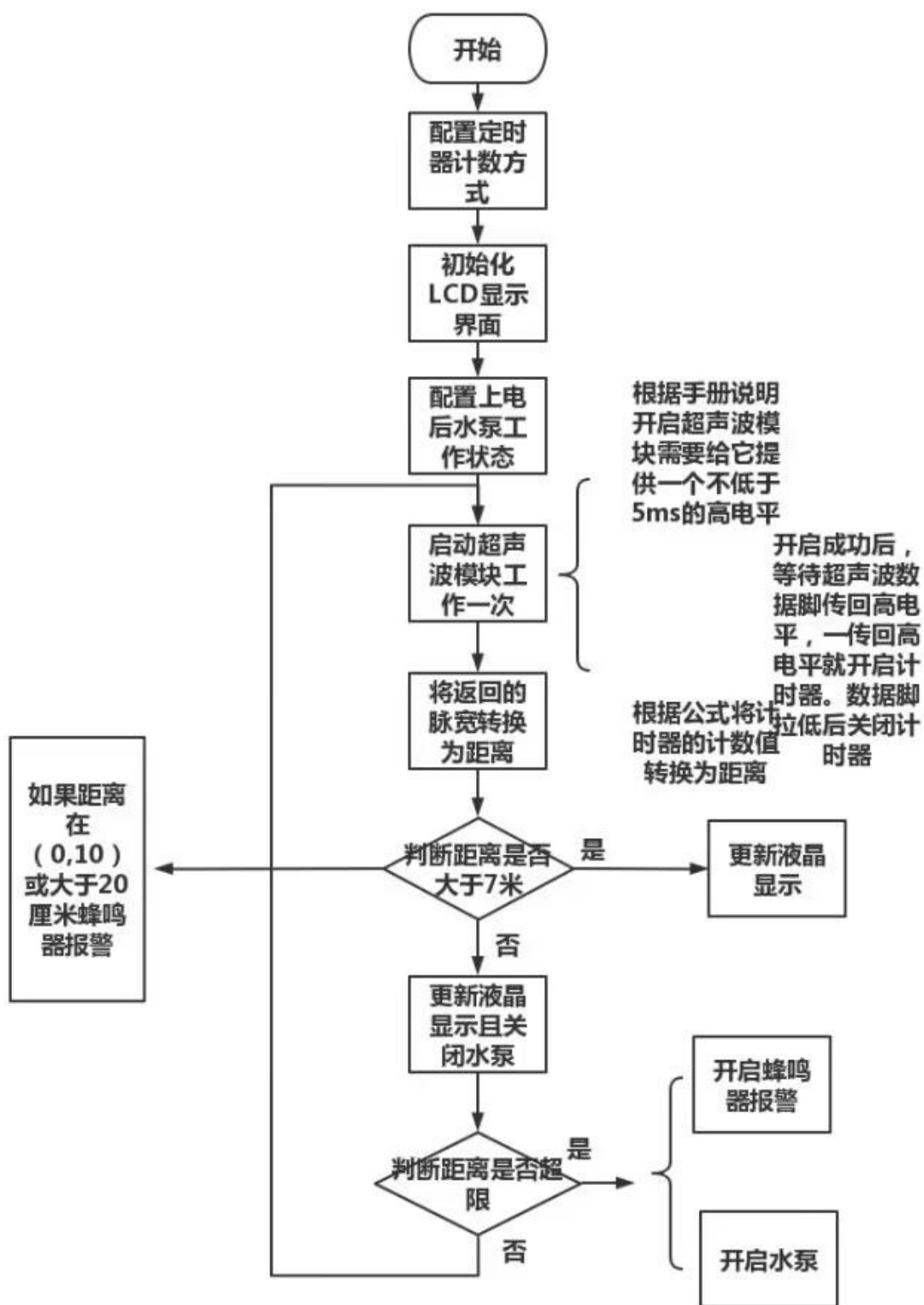


图 5 系统流程图

## 6. 小组所有成员的分工说明；所有成员的实验心得及得分

### 6.1 小组成员分工

黄惠齐：负责编写超声波模块与单片机进行数据传输的程序，LCD1602 与单片机的通信，完成在 1602 液晶上面显示超声波模块测得的数据并负责后期的测试。同时组织本小组完成本次的项目，最后对整个程序进行测试与讲解。

黄昊林：负责编写超声波模块、水泵、蜂鸣器与单片机的通信代码程序，研究本次实验超声波的硬件接线，并在网上查找单片机与超声波模块的通信时序图，完成在 1602 液晶上面显示超声波模块测得的数据并负责后期的测试。完成最终周志及报告论文的编写修改。

何文涛：负责研究本次实验超声波的硬件接线，蜂鸣器应用，控制蜂鸣器的声响。并在网上查找单片机与超声波传感器模块的通信时序，并负责在这个程序当中排除 bug，完善实验的程序，与小组成员予以讲解。

### 6.2 小组成员实验心得

何文涛：在本次实验当中，我认识到单片机在电子领域当中的作用。这种计算机的最小系统只用了一片集成电路，即可进行简单运算和控制。为学习、应用和开发提供了便利条件。

黄昊林：在本次实验当中，我们使用的是超声波传感器模块，然而我们在应用当中发现，当我们将超声波模块靠得很近的时候，超声波模块显示出来的数据已经超出了传感器的测量范围了，这让我不得不思考，在设计电路的时候，应该尽可能的提高测量精度，从而减少误差的出现。

黄惠齐：在本次实验当中，我们只是简单地应用单片机，只是按照教程进行硬件接线才能完成我们的实验。后来我了解到做这个应用实验不单单只是完成这个实验现象而已，而更应该去理解这个实验的每一个细节。

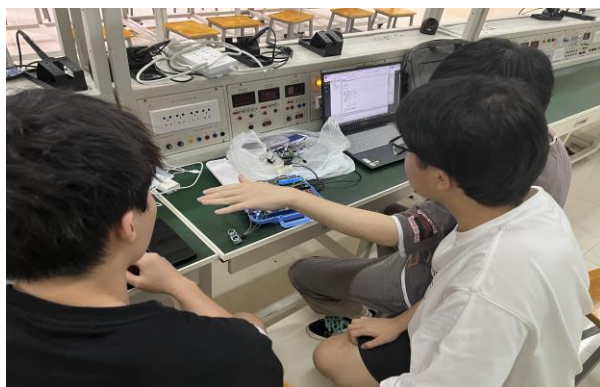


图 6 实验过程照片图①



图 7 实验过程照片图②



图 8 实验过程照片图③

### 6.3 小组成员得分

黄昊林：100 分

黄惠齐：98 分

何文涛：98 分

## 7. 代码实现

```
#include<reg51.h>
#include<intrins.h>
#include"lcd.h"
```

```
sbit Trig = P2^1;
sbit Echo = P2^0;
sbit buzzer = P2^3;
sbit motor = P1^0;
```

```
unsigned char PuZh[]=" Liquid Distance ";
unsigned char code ASCII[15] =
{'0','1','2','3','4','5','6','7','8','9','.',',','-','M'};
```

```
void delay(unsigned int ms);
```

```
static unsigned char DisNum = 0; //显示用指针
```

```
unsigned int time=0;
unsigned long S=0, H=25;
bit flag=0;
```

```
unsigned char disbuff[4] = { 0,0,0,0};
```

```
void Conut(void) {
```

```
unsigned int i = 100;
time=TH0*256+TL0;
TH0=0;
TL0=0;
```

```
if((S>0)&&(S<10)){
```

```
while(i--)
```

```
{
buzzer=!buzzer;
```

```
//蜂鸣器报警
```

```
delay(2);
```

```
}
```

```
}
```

```
if(S > 15){
```

```
while(i--)
```

```
{
```

```
buzzer=!buzzer;
```

```
delay(2);
```

```
}
```

```
}
```

```
S=H-(time*1.7)/100; //算出来是 CM
```

```
if((S>=700)||flag==1) //超出测量范围显示 “-”
{
flag=0;
```

```
DisplayOneChar(0, 1, ASCII[11]);
```

```
DisplayOneChar(1, 1, ASCII[10]); //显示点
```

```
DisplayOneChar(2, 1, ASCII[11]);
```

```
DisplayOneChar(3, 1, ASCII[11]);
```

```
DisplayOneChar(4, 1, ASCII[12]); //显示 M
```

```
}
```

```
else
```

```
{
```

```
disbuff[0]=S%1000/100;
```

```
disbuff[1]=S%1000%100/10;
```

```
disbuff[2]=S%1000%10 %10;
```

```
DisplayOneChar(0, 1, ASCII[disbuff[0]]);
```

```
DisplayOneChar(1, 1, ASCII[10]); //显示
```

点

```
DisplayOneChar(2, 1, ASCII[disbuff[1]]);
```

```
DisplayOneChar(3, 1, ASCII[disbuff[2]]);
```

```
DisplayOneChar(4, 1, ASCII[12]); //显示
```

M

```
motor=0;
```

```
}
```

```
}
```

```
void zd0() interrupt 1 { //T0 中断用来计数器溢出,
超过测距范围
```

```
flag=1;
```

```
//中断溢出标志
```

```
}
```

```
void StartModule() { //启动模块
```

```
Trig=1;
```

```
//启动一次模块
```

```
_nop_();
```

```
_nop_();
```

```
_nop_();
```

```
_nop_();
```

```
_nop_();
```

```
_nop_();
```

```
_nop_();
```

```
_nop_();
```

```

        _nop_();
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        _nop_();
        Trig=0;
    }

void delayms(unsigned int ms){
    unsigned char i=100,j;
    for(;ms;ms--)
    {
        while(--i)
        {
            j=10;
            while(--j);
        }
    }
}

void delay(unsigned int ms){
    unsigned char i=1,j;
    for(;ms;ms--)
    {
        while(--i)
        {
            j=1;
            while(--j);
        }
    }
}

    }
}

unsigned char buzzerflag = 0;
void buzzer1() interrupt 2{
}

/**主函数**/
void main(void){
    unsigned int i = 50;
    TMOD=0x01;           //设 T0 为方式 1,
    GATE=1;
    TH0=0;
    TL0=0;
    ET0=1;               //允许 T0 中断
    IT1 = 0;
    EX1 = 1;             //打开 T1 中断
    EA=1;                //开启总中断
    LcdInit();
    LcdShowStr(0,0,PuZh);
    motor=0;
    while(1)
    {
        StartModule();
        while(!Echo);    //当 RX 为零时等待
        TR0=1;           //开启计数
        while(Echo);     //当 RX 为 1 计数并等待
        TR0=0;           //关闭计数
        Conut();         //计算

        if(S > 15){
            motor=1;
        }

        delayms(20);
    }
}

```