

第 3-8 讲：超声波测距（基于 HC-SR04）

1. 学习目的

1. 了解超声波测距原理。
2. 掌握单片机编程使用 HC-SR04 测距，并使用 LCD1602 液晶显示模块显示测试结果。

2. 超声波测距

2.1. 超声波测距原理

超声波是利用反射的原理测量距离的，被测距离一端为超声波传感器，另一端必须有能反射超声波的物体。测量距离时，将超声波传感器对准反射物发射超声波，并开始计时，超声波在空气中传播到达障碍物后被反射回来，传感器接收到反射脉冲后立即停止计时，然后根据超声波的传播速度和计时时间就能计算出两端的距离。

当介质为空气时，声速为 340m/s，根据记录的时间，即可计算出发射位置与障碍物之间的距离。

2.2. HC-SR04 超声波测距模块

1. 模块简介

艾克姆的 HC-SR04 超声波测距模块如下图所示，其核心是两个压电陶瓷超声波传感器。一个用作发射器，当接收到“Trig”引脚的触发信号后对外发射 40 KHz 超声波脉冲，另外一个用作接收器，监听到发射器发射的脉冲后，“Echo”引脚将产生一个输出脉冲，该脉冲宽度为超声波发出到接收的时间，由此可确定脉冲传播的距离。



图 1: HC-SR04 超声波测距模块

2. 模块参数

电气参数	HC-SR04超声波模块
工作电压	DC3.3V ~ DC5V
最小静态工作电流	小于5mA
工作频率	40Hz
最远射程	4.5m
最近射程	2cm
测量角度	15度
输入触发信号	10us的TTL脉冲
输出回响信号	输出TTL电平信号，与射程成比例
重量	约8.3克
规格尺寸	45*20*15mm



图 2：模块参数

3. 引脚定义

表 1：HC-SR04 超声波测距模块引脚描述

序号	引脚名称	功能描述
1	VCC	模块供电正，典型工作电压（3.3~5）VDC。
2	Trig	触发端。
3	Echo	接收端，当接收到反射信号时，引脚产生一个脉冲。脉冲的长度与检测发射信号所需的时间成正比。
4	GND	模块供电负。

4. 工作流程

HC-SR04 超声波测距模块工作时序如下图所示。

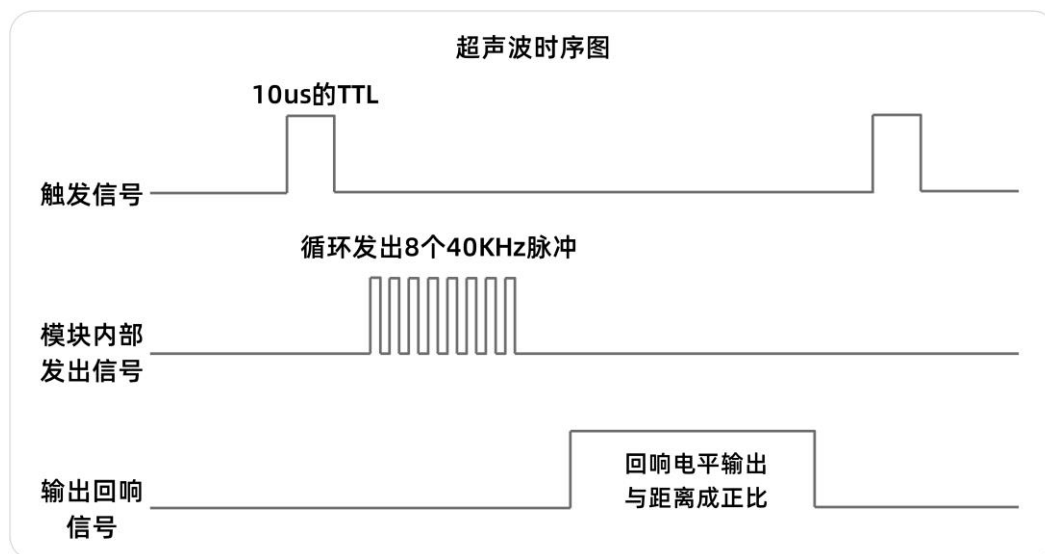


图 3：超声波时序图

流程:

- 1) 通过 Trig 发送触发信号，给宽度至少 10us 的脉冲信号；
- 2) 超声波模块自动发送 8 个 40khz 的方波，自动检测是否有信号返回；
- 3) 有信号返回后，超声波模块 Echo 引脚输出一个高电平，高电平持续的时间就是超声波从发射到返回的时间，距离计算公式如下：

$$\text{距离} = \frac{\text{高电平时间}(S) \times 340(M/S)}{2} M$$

3. 软件设计

3.1. 超声波测距实验（串口输出测试结果）

✧ 注：本节的实验是在“实验 2-6-1：串口 1 数据收发实验”的基础上修改，本节对应的实验源码是：“实验 3-8-1：超声波测距实验（串口输出测试结果）”。

3.1.1. 实验内容

- 1) STC8A8K64D4 单片机的 P1.7 连接 HC-SR04 超声波模块的 Trig 引脚，测量距离时发送触发信号；P1.7 连接 HC-SR04 超声波模块的 Echo 引脚，获取回响信号。
- 2) 使用 Timer0 测量回响信号高电平持续时间，定时器时钟配置为 12T，在 24MHz 主频下，一个计数值的时间为 0.5us。
- 3) 每 200ms 测量一次，计算出距离后通过串口输出结果。

3.1.2. 代码编写

1. 新建一个名称为“HC-SR04.c”的文件及其头文件“HC-SR04.h”保存到工程的“Source”文件夹，并将“HC-SR04.c”加入到 Keil 工程中的“SOURCE”组。该文件用于存放 HC-SR04 测距操作相关的函数。

2. 引用头文件

因为在“main.c”文件中使用了“hc_sr04.c”文件中的函数，所以需要引用下面的头文件“hc_sr04.h”。

代码清单：引用头文件

```
1. //引用 HC-SR04 的头文件
2. #include "hc_sr04.h"
```

3. 定义引脚

本例中，我们用 P1.6 和 P1.7 分别用于 HC-SR04 超声波测距模块的 ECHO 和 TRIG 信号引脚，定义如下。

代码清单：定义用于 I2C 通信的引脚

```
1. sbit ECHO = P1^6; //连接 HC-SR04 超声波模块的 Echo 引脚，获取回响信号
2. sbit TRIG = P1^7; //连接 HC-SR04 超声波模块的 Trig 引脚，发送触发信号
```

4. 初始化定时器 0

Timer0 用于测试回响信号高电平持续时间，这里我们需要计算一下，以确保 Timer0 的最大溢出时间大于回响信号高电平持续时间，这样，我们每次测量时，就可以让 Timer0 从

0 开始计数，测量完成后读出 Timer0 的计数值即可计算出时间，同时，还可以利用定时器的溢出中断判断测量是否超时。

- 1) 超声波最大测试距离为 4.5m，对应的回响高电平持续时间 = $(4.5 \times 2) / 340(S) \approx 0.0265S$ ；
- 2) Timer0 时钟配置为 12T，在 24MHz 主频下，一个计数值的时间为 0.5us，最大溢出时间 = $65536 \times 0.5us = 0.0327S$ ，满足要求。

定时器初始化代码清单如下：

代码清单：初始化 Timer0

```

1.  /*****
2.  功能描述: 初始化 Timer0 (系统时钟使用 24MHz)
3.  参 数: 无
4.  返回值: 无
5.  *****/
6.  void timer0_init(void)
7.  {
8.      AUXR &= 0x7F;      //定时器时钟 12T 模式
9.      TMOD &= 0xF0;      //配置 Timer0 为定时器
10.     TL0 = 0x00;         //设置定时初始值
11.     TH0 = 0x00;         //设置定时初始值
12.     ET0 = 1;           //使能定时器 0 中断
13. }
```

5. 测量距离

按照 HC-SR04 超声波模块工作流程编写测量函数如下，需要注意的是：在使用 while 循环等待回响信号时，要进行超时判断（这里也可以使用 IO 中断来判断）。

代码清单：触发超声波测量一次距离

```

1.  /*****
2.  * 描 述 : 触发超声波测量一次距离
3.  * 参 数 : 无
4.  * 返回值 : =0: 测量超时, 测量错误; >0: 测量的距离, 单位 CM
5.  *****/
6.  float measure_distance(void)
7.  {
8.      u16 t;
9.      float dist;
10.
11.     TL0 = 0x00;         //设置定时初始值
12.     TH0 = 0x00;         //设置定时初始值
13.     TRIG=1;             //发出触发信号 (Trig 输出高电平), 持续时间大于 10uS
14.     delay_ms(1);
15.     TRIG=0;
16.     while((!ECHO) && (meas_time_out == false)); //等待 ECHO 端口回响信号(ECHO 端口由低电平变为高电平)
17.     TR0=1;              //启动定时器 0 开始计数
```

```

18. while((ECHO)&&(meas_time_out==false)); //等待 ECHO 端口回响信号结束(ECHO 端口由高电平变为低电平)
19. TR0=0; //停止定时器 0 计数
20. meas_time_out = false;
21. if(meas_time_out == true)return 0; //定时器溢出, 测量出错
22. t = (u16)(((TH0<<8) + TL0)/2); //读取 Timer0 计数值,换算成 us
23. dist = (float)(1.7 *t /100); //计算出距离, 单位 CM
24.
25. return dist;
26. }

```

6. 主函数

主函数中完成相关的初始化, 之后, 在主循环中每 200ms 测量距离, 并通过串口输出测量结果。

代码清单: 主函数

```

1. /*****
2. 功能描述: 主函数
3. 入口参数: 无
4. 返回值: int 类型
5. *****/
6. int main(void)
7. {
8.     float dist_cm;
9.
10.    P2M1 &= 0xBF; P2M0 &= 0xBF; //设置 P2.6 为准双向口 (LED1)
11.    P3M1 &= 0xFE; P3M0 &= 0xFE; //设置 P3.0 为准双向口 (串口 1 的 RxD)
12.    P3M1 &= 0xFD; P3M0 |= 0x02; //设置 P3.1 为推挽输出 (串口 1 的 TxD)
13.    P1M1 &= 0x3F; P1M0 &= 0x3F; //设置 P1.6~P1.7 为准双向口
14.    TRIG=0; //触发引脚设置为低电平
15.    uart1_init(); //串口 1 初始化
16.    timer0_init(); //定时器 0 初始化
17.    EA = 1; //使能总中断
18.
19.    while(1)
20.    {
21.        dist_cm = measure_distance(); //执行一次测距
22.        if(dist_cm != 0)
23.        {
24.            printf("distance %.2f CM\r\n",dist_cm);//串口打印测距结果, 单位 CM
25.        }
26.        else
27.        {
28.            printf("err:measure timeout\r\n"); //测距超时出错
29.        }

```

```

30.     delay_ms(200);
31.     led_toggle(LED_1);           //翻转指示灯 D1 的状态，指示一次测距完成
32. }
33. }

```

3.1.3. 硬件连接

如下图所示，用杜邦线将 HC-SR04 超声波模块连接到开发板上。

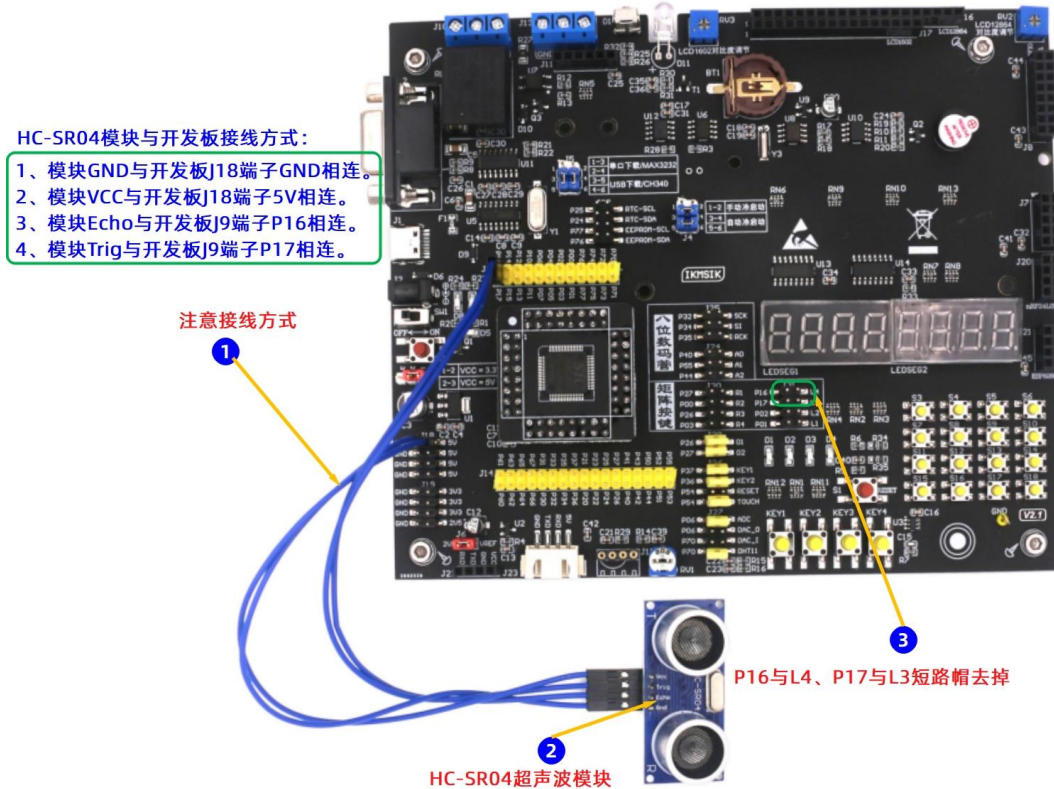


图 4：硬件连接

3.1.4. 实验步骤

- 1) 解压“…\第3部分：配套例程源码”目录下的压缩文件“实验 3-8-1：超声波测距实验（串口输出测试结果）”，将解压后得到的文件夹拷贝到合适的目录，如“D:\STC8”（这样做的目的是为了防止中文路径或者工程存放的路径过深导致打开工程出现问题）。
- 2) 双击“…\HC-SR04\project”目录下的工程文件“HC-SR04.uvproj”。
- 3) 点击编译按钮编译工程，编译成功后生成的 HEX 文件“HC-SR04.hex”位于工程的“…\HC-SR04\Project\Object”目录下。
- 4) 打开 STC-ISP 软件下载程序，下载使用内部 IRC 时钟，IRC 频率选择：24MHz。
- 5) 电脑上打开串口调试助手，选择开发板对应的串口号，将波特率设置为 9600bps。
- 6) 程序运行后，将超声波模块探头正对墙壁或其他物体测量，串口调试助手接收窗口可以看到测试结果，将超声波模块远离/靠近墙壁，可以观察到距离变大/减小，如下图所示。

- ◇ **注意事项：**测距时，被测物体的面积不少于 0.5 平方米且要尽量平整。否则会影响测试结果。



图 5：超声波测距

3.2. 光照强度测量实验（连续高精度模式）

- ◇ **注：**本节的实验是在“实验 3-8-1：超声波测距实验（串口输出测试结果）”的基础上修改，本节对应的实验源码是：“实验 3-8-2：超声波测距实验（LCD1602 显示测试结果）”。

3.2.1. 实验内容

在“实验 3-8-1：超声波测距实验（串口输出测试结果）”的基础上，将测量的距离在 LCD1602 液晶显示模块上显示出来。

3.2.2. 代码编写

1. 将“实验 3-3-1：LCD1602 显示字符”源码目录“...\lcd1602\Source”下的“lcd1602.c”的文件及其头文件“lcd1602.h”拷贝到本工程的“Source”文件夹，并将“lcd1602.c”加入到 Keil 工程中的“SOURCE”组。
2. main()函数中，调用 measure_distance()函数获取测量结果后，通过 sprintf 函数将浮点数转换为字符串存储到数组中，接着调用 LCD1602 显示函数将存储在数组中的字符显示在液晶屏上。

代码清单：主函数

```

1.  /*****
2.  功能描述：主函数
3.  参 数：无
4.  返回值：int 类型
5.  *****/
6.  int main(void)
7.  {
8.      u8 i = 0;
9.      u16 dist_int;

```

```
10. float dist_cm;
11.
12. P2M1 &= 0xBF; P2M0 &= 0xBF; //设置 P2.6 为准双向口（指示灯 D1）
13. P3M1 &= 0xFE; P3M0 &= 0xFE; //设置 P3.0 为准双向口（串口 1 的 RxD）
14. P3M1 &= 0xFD; P3M0 |= 0x02; //设置 P3.1 为推挽输出（串口 1 的 TxD）
15.
16. P3M1 &= 0xCB; P3M0 |= 0x34; //设置 P3.2,P3.4,P3.5 为强推挽输出（LCD1602 控制）
17. P6M1 &= 0x00; P6M0 &= 0x00; //设置 P6.0~P6.7 为准双向口（LCD1602 数据）
18. P1M1 &= 0x3F; P1M0 &= 0x3F; //设置 P1.6~P1.7 为准双向口（HC-SR04 的 Echo 和 Trig）
19.
20. uart1_init(); //串口 1 初始化
21. lcd1602_init(); //LCD1602 显示模块初始化
22. timer0_init(); //定时器 0 初始化
23. EA = 1; //使能总中断
24.
25. while(1)
26. {
27.     LCD1602_WriteCmd(0x01); //清屏
28.     dist_cm = measure_distance(); //执行一次测距
29.     if(dist_cm != 0)
30.     {
31.         printf("distance %.2f CM\r\n",dist_cm);//串口打印测距结果，单位 CM
32.         dist_int = (u16)dist_cm; //得到整数部分
33.         sprintf(dis_buf,"%3.02f",dist_cm); //将浮点数转换为字符串并存储到 disp_buf 数组
34.         LCD1602_DisString(0, 0, table1);
35.         if(dist_int>=100) //3 位整数
36.         {
37.             for(i=0;i<6;i++)LCD1602_DisOneChar(1, i, disp_buf[i]);
38.         }
39.         else if(dist_int>=10) //2 位整数
40.         {
41.             for(i=0;i<5;i++)LCD1602_DisOneChar(1, i, disp_buf[i]);
42.         }
43.         else if(dist_int>0) //1 位整数
44.         {
45.             for(i=0;i<4;i++)LCD1602_DisOneChar(1, i, disp_buf[i]);
46.         }
47.     }
48.     else
49.     {
50.         printf("err:measure timeout\r\n"); //测距超时出错
51.         LCD1602_DisString(1, 0, table2); //显示“err”，提示测量出错
52.     }
```

```

53.     delay_ms(1000);
54.     led_toggle(LED_1);           //翻转指示灯 D1 的状态，指示一次测距完成
55. }
56. }

```

3.2.3. 硬件连接

如下图所示，连接 HC-SR04 超声波模块和 LCD1602 液晶显示模块。

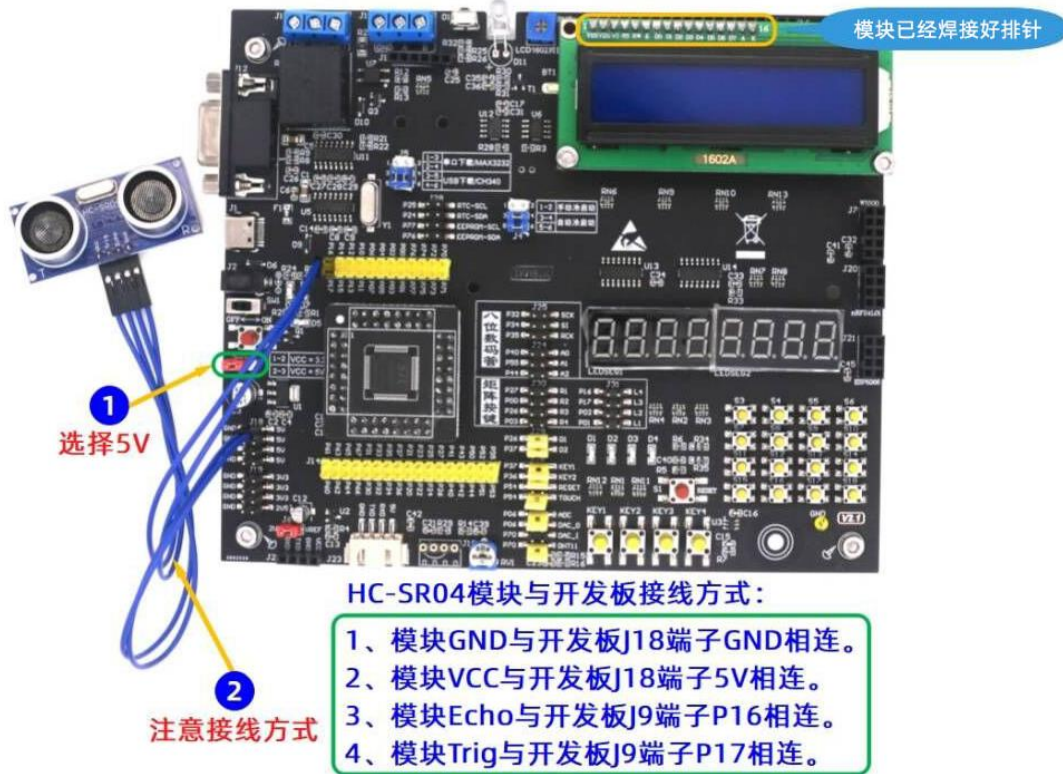


图 5：硬件连接

3.2.4. 实验步骤

- 1) 解压“…\第3部分：配套例程源码”目录下的压缩文件“实验 3-8-2：超声波测距实验（LCD1602 显示测试结果）”，将解压后得到的文件夹拷贝到合适的目录，如“D:\STC8”（这样做的目的是为了防止中文路径或者工程存放的路径过深导致打开工程出现问题）。
 - 2) 双击“…\HC-SR04-DISP\project”目录下的工程文件“HC-SR04-DISP.uvproj”。
 - 3) 点击编译按钮编译工程，编译成功后生成的 HEX 文件“HC-SR04-DISP.hex”位于工程的“…\HC-SR04-DISP\Project\Object”目录下。
 - 4) 打开 STC-ISP 软件下载程序，下载使用内部 IRC 时钟，IRC 频率选择：24MHz。
 - 5) 电脑上打开串口调试助手，选择开发板对应的串口号，将波特率设置为 9600bps。
 - 6) 程序运行后，将超声波模块探头正对墙壁或其他物体测量，LCD1602 液晶显示模块上可以看到测试结果，如下图所示。
- ❖ **注意事项：**测距时，被测物体的面积不少于 0.5 平方米且要尽量平整。否则会影响测试结果。



图 6: LCD1602 显示测量的距离