

US-015 超声波测距模块 V2.0

1. 概述

US-015 是目前市场上分辨率最高，重复测量一致性最好的超声波测距模块；US-015 的分辨率高于 1mm，可达 0.5mm，测距精度高；重复测量一致性好，测距稳定可靠。

US-015 超声波测距模块可实现 2cm~4m 的非接触测距功能，供电电压为 5V，工作电流为 2.2mA，支持 GPIO 通信模式，工作稳定可靠。

2. 分辨率及可重复性测试截图

图 2.1 为手拿 US-015 进行测量，手有微小抖动时的测量截图，可见小于 1mm 的抖动都能测量出来；显示分辨率为 0.01mm。

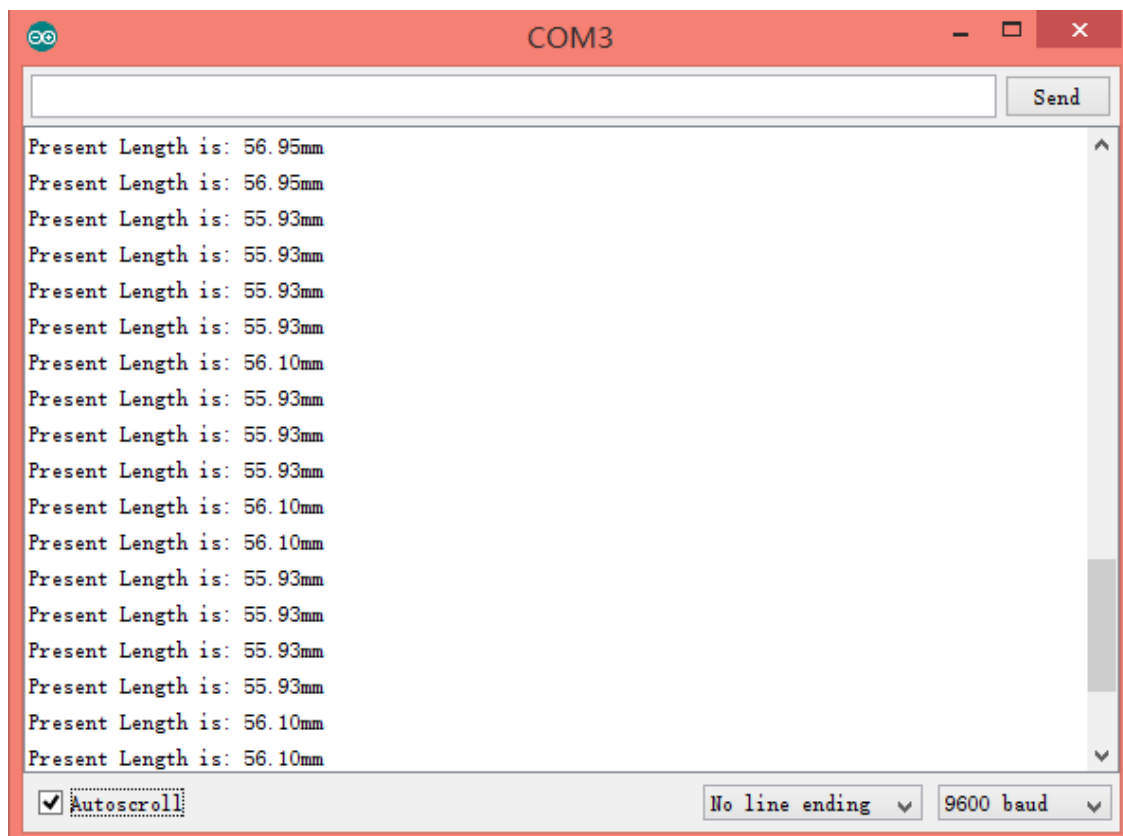


图 2.1：手持 US-015 微小抖动测量截图

图 2.2 为将 US-015 固定后，经过一段时间测量后的截图，可见重复测量一致性好。显示分辨率为 0.01mm

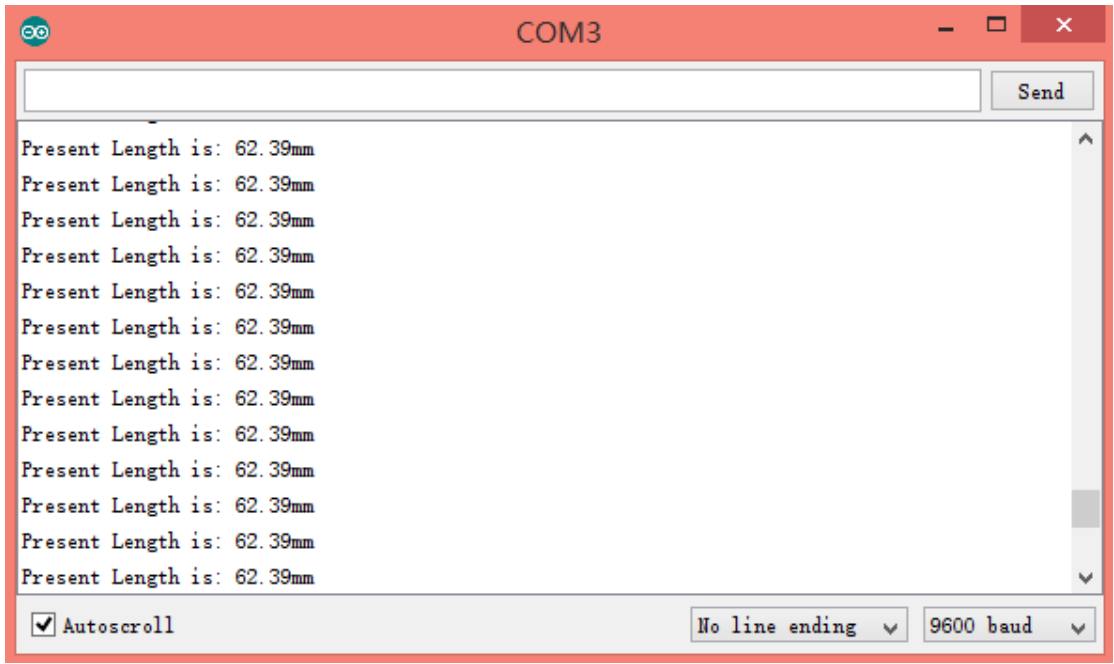


图 2.2：重复测量截图

图 2.1 及图 2.2 所用例程请参考后文附录。

3. 主要技术参数

电气参数	US-015 超声波测距模块
工作电压	DC 5V
工作电流	2.2mA
工作温度	0~+70 度
输出方式	GPIO
感应角度	小于 15 度

探测距离	2cm-400cm
探测精度	0.1cm+1%
分辨率	高于 1mm（可达 0.5mm）

4. 本模块实物图及尺寸

本模块如图 4.1 和图 4.2 所示：



图 4.1： US-015 正面图

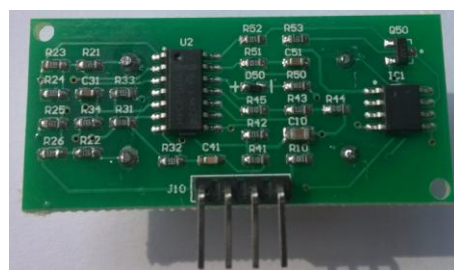


图 4.2： US-015 背面图

本模块的尺寸：45mm*20mm*1.2mm。板上有两个半径为 1mm 的机械孔，如图 4.3 所示：

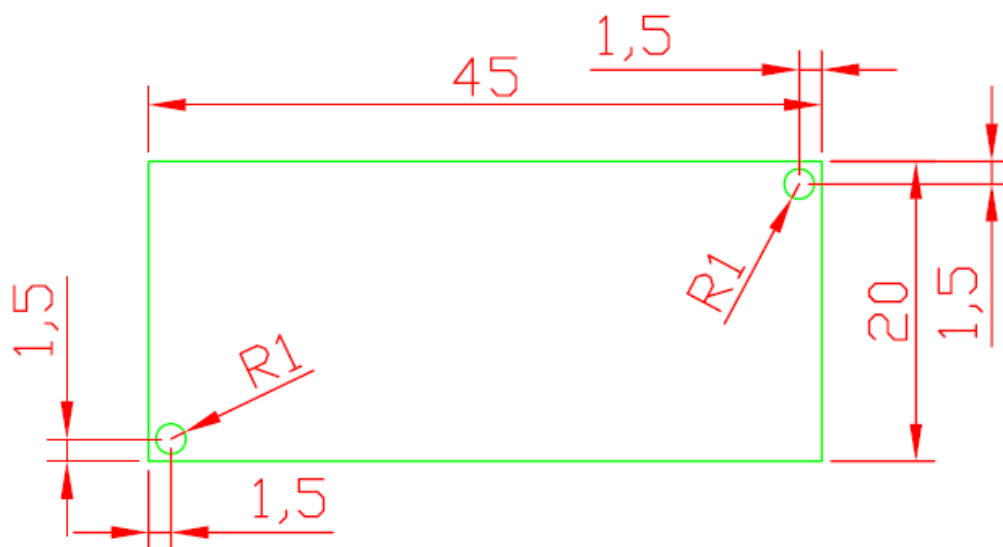


图 4.3: US-015 尺寸图

5. 接口说明

本模块有一个接口：4 Pin 供电及通信接口。

4 Pin 接口为 2.54mm 间距的弯排针，如图 5.1 所示：



图 5.1: 4 Pin 接口

从左到右依次编号 1, 2, 3, 4。它们的定义如下：

- 1 号 Pin: 接 VCC 电源（直流 5V）。
- 2 号 Pin: 接外部电路的 Trig 端，向此管脚输入一个 10uS 以上的高电平，可触发模块测距。

- 3 号 Pin: 接外部电路的 Echo 端，当测距结束时，此管脚会输出一个高电平，电平宽度为超声波往返时间之和。
- 4 号 Pin: 接外部电路的地。

6. 测距工作原理

模块测距的时序如图 6.1 所示：

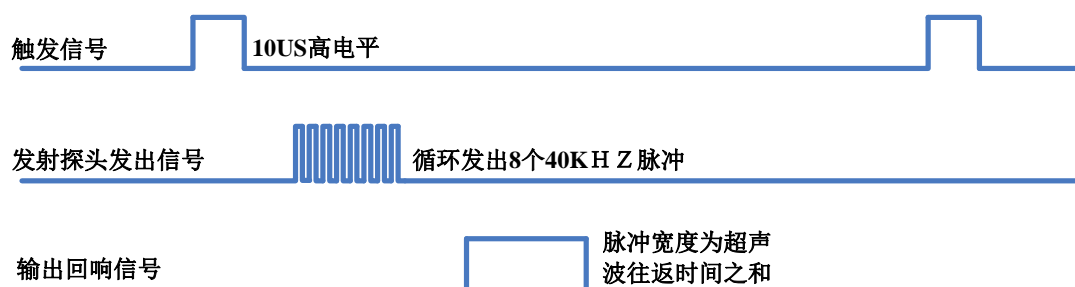


图 6.1：US-015 测距时序图

图 6.1 表明：只需要在 Trig 管脚输入一个 10US 以上的高电平，系统便可发出 8 个 40KHZ 的超声波脉冲，然后检测回波信号。当检测到回波信号后，通过 Echo 管脚输出。

根据 Echo 管脚输出高电平的持续时间可以计算距离值。即距离值为： $(\text{高电平时间} \times 340\text{m/s}) / 2$ 。

7. 超过测量范围时返回值及测量周期

当测量距离超过 US-015 的测量范围时，US-015 仍会通过 Echo 管脚输出高电平的信号，高电平的宽度约为 80ms。如图 7.1 所示：

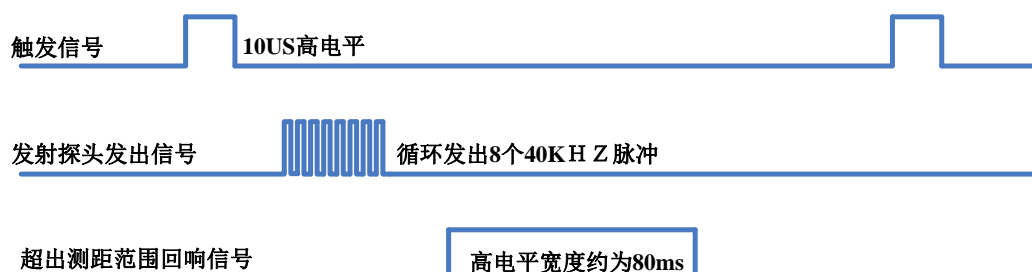


图 7.1：超出测量范围返回值

测量周期：当接收到 US-015 通过 Echo 管脚输出的高电平脉冲后，便可进行下一次测量，所以测量周期取决于测量距离，当 US-015 距离被测物体很近时，Echo 返回的脉冲宽度较窄，测量周期就很短；当 US-015 距离被测物体比较远时，Echo 返回的脉冲宽度较宽，测量周期也就相应的变长。

最坏情况下，被测物体超出 US-015 的测量范围，此时返回的脉冲宽度最长，约为 80ms，所以最坏情况下的测量周期稍大于 80ms 即可（取 85ms 足够）。

附录：US-015 高精度测距例程，（Arduino 例程）

```
unsigned int EchoPin = 2;

unsigned int TrigPin = 3;

unsigned long Time_Echo_us = 0;
```

```

//Len_mm_X100 = length*100

unsigned long Len_mm_X100  = 0;

unsigned long Len_Integer = 0; //

unsigned int Len_Fraction = 0;

void setup()

{

    Serial.begin(9600);

    pinMode(EchoPin, INPUT);

    pinMode(TrigPin, OUTPUT);

}


void loop()

{

    digitalWrite(TrigPin, HIGH);

    delayMicroseconds(50);

    digitalWrite(TrigPin, LOW);


    Time_Echo_us = pulseIn(EchoPin, HIGH);

    if((Time_Echo_us < 60000) && (Time_Echo_us > 1))

    {

        Len_mm_X100 = (Time_Echo_us*34)/2;

        Len_Integer = Len_mm_X100/100;

```

```
    Len_Fraction = Len_mm_X100%100;

    Serial.print("Present Length is: ");

    Serial.print(Len_Integer, DEC);

    Serial.print(".");

    if(Len_Fraction < 10)

        Serial.print("0");

    Serial.print(Len_Fraction, DEC);

    Serial.println("mm");

}

delay(1000);

}
```