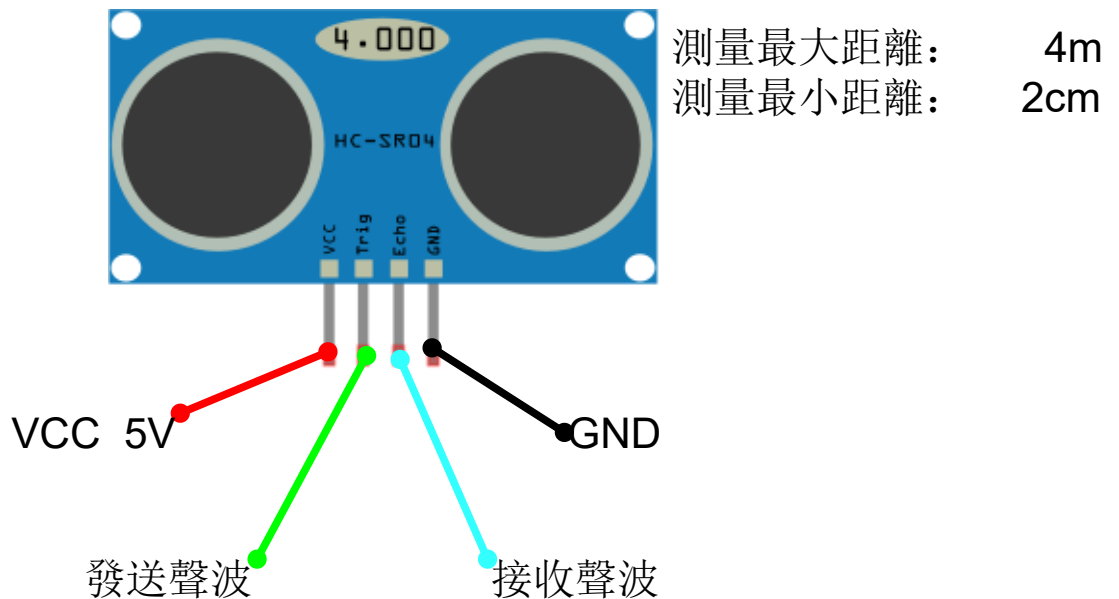


超音波感測器 HC-SR04

此章節解說如何透過 NodeMCU 擷取超音波感測器(HC-SR04)的數值，及數值的意義。如果對於建置環境不了解，先參考「NodeMCU_HelloWorld」章節

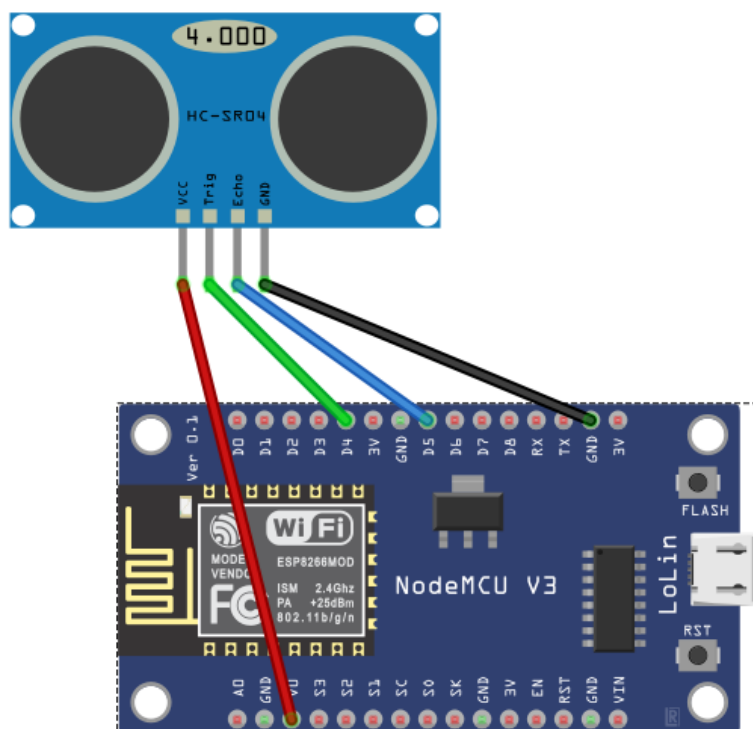
硬體介紹：



前置作業：

將 VCC 與 GND 接上，

發送聲波接在 D4 Pin 腳，接收聲波接在 D5 Pin 腳



1. 原理解說

HC-SR04 透過 40kHz的超聲波運作，此頻率人類無法聽到。當 Trigger 發送超聲波後，超聲波遇到障礙物(**Object**)而反射，此時Echo接收到反射的超聲波，得到測量時間(**Duration**)。我們知道聲音的速度約 340M/sec，因此可從公式得出聲音走的距離(**Sonar Distance**):

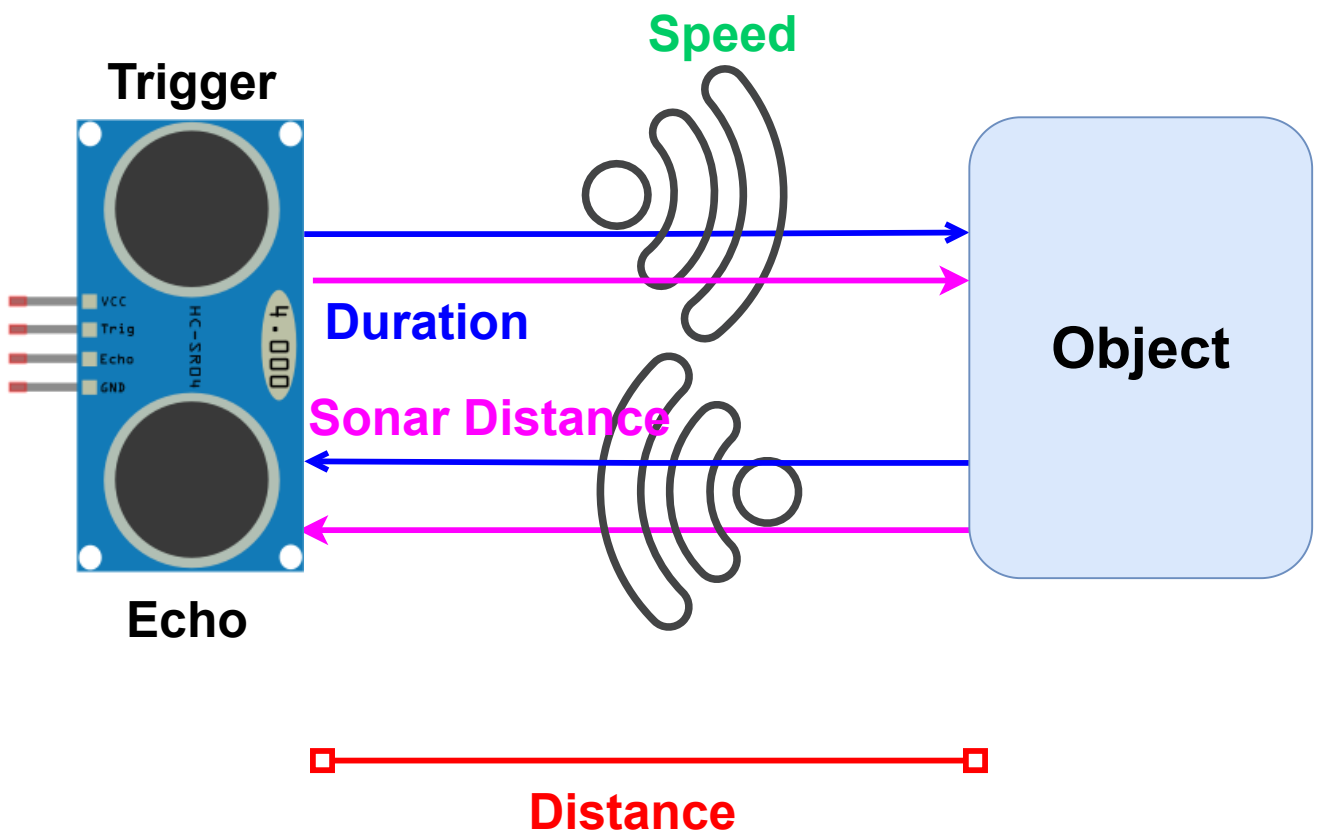
$$\text{Sonar Distance} = \text{Speed} * \text{Duration}$$

進一步得到測量距離(**Distance**):

$$\text{Distance} = \text{Sonar Distance} / 2$$

公式總結為:

$$\text{Distance} = \text{Speed} * \text{Duration} / 2$$



2. 編寫草稿碼 -> 上傳至 NodeMCU 開發板



```
sketch_may04a | Arduino 1.8.12
檔案 編輯 草稿碼 工具 說明

sketch_may04a

//Sensor-VCC -> DUT-Vin pin

const int TRIGGER = D4; //D4
const int ECHO = D5;    //D5

// defines variables
long duration;
float distance;

void setup() {
  //console baud rate
  Serial.begin(115200);

  pinMode(TRIGGER, OUTPUT); // 定義 trigger 為送出模式
  pinMode(ECHO, INPUT);    // 定義 echo 為接受模式
}

void loop() {
  digitalWrite(TRIGGER, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIGGER, HIGH);
  delayMicroseconds(10);           //Trigger 需要維持10us的高電位才會發送聲波
  duration = pulseIn(ECHO, HIGH);  //等待收到Trigger的訊號。
  printf("=====\n");

  //正確的公式包含溫度: Speed of sound m/s = 331.4 + (0.606 * Temp) + (0.0124 * Humidity)
  distance = (float)34000/1000000 * duration / 2; //計算CM的距離: 聲速(34000CM/1000000us) * 測量時間 = 聲音走的距離, 聲音走的距離/2 = 測量距離
  printf("Duration:%ld us, Distance:%.1f CM\n", duration, distance);

  delay(1000);                    //每秒偵測物體距離
}

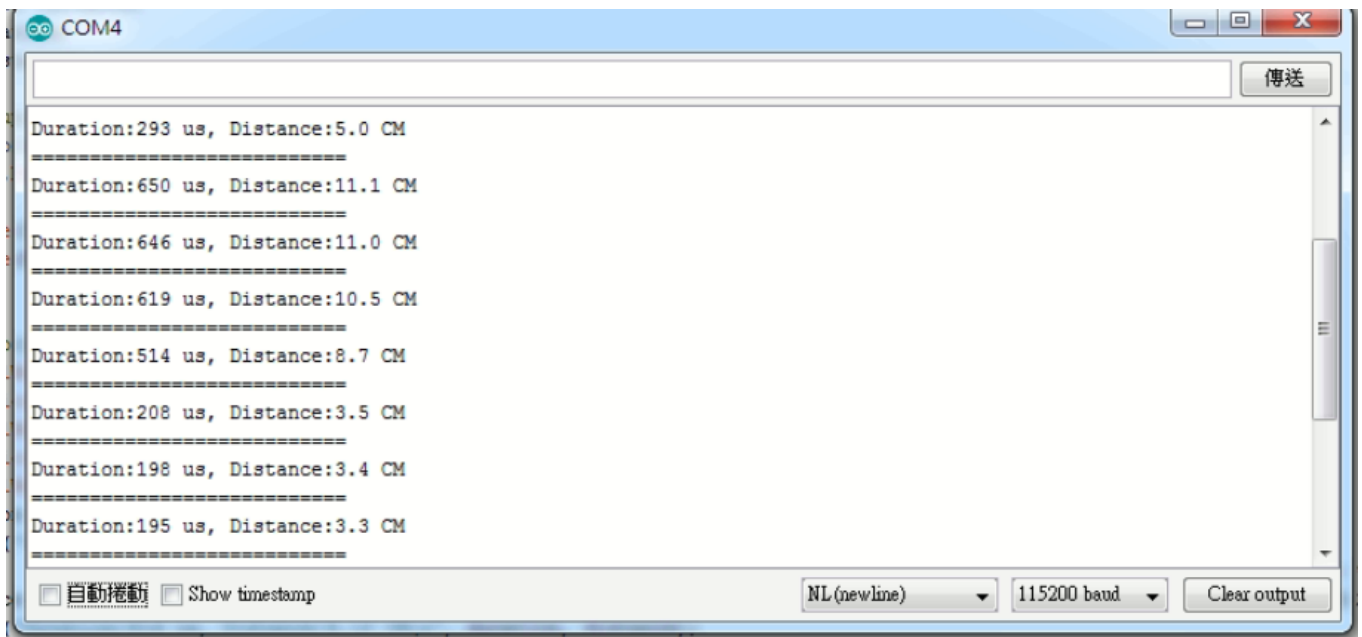
上傳完畢。
Writing at 0x00028000... (84 %)
Writing at 0x0002c000... (92 %)
Writing at 0x00030000... (100 %)
Wrote 272464 bytes (200292 compressed) at 0x00000000 in 19.2 seconds (effective 113.6 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

29 NodeMCU 1.0 (ESP-12E Module) 於 COM4
```

由於接受聲波的函式pulseIn，回傳值的單位為 microsecond，將340m/sec 轉換為 34000(cm)/1000000(us)

3. 觀看結果



測量距離會隨著物件的移動而改變