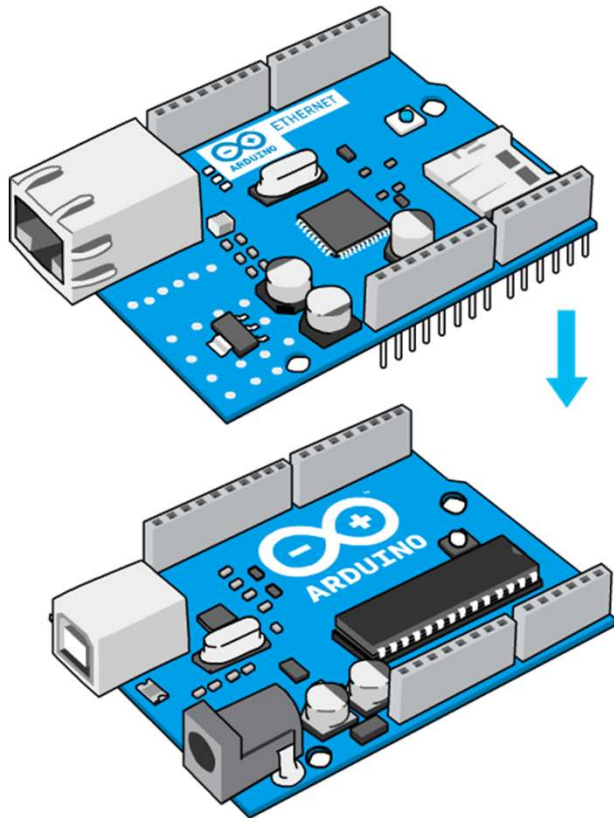
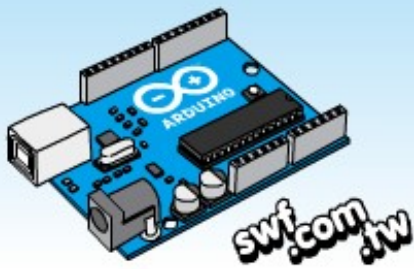


第九章 LCD顯示器 + 溫濕度感測 + 超音波感測器

- 串列連接LCD顯示器
- 數位溫濕度感測器
- 超音波感測器

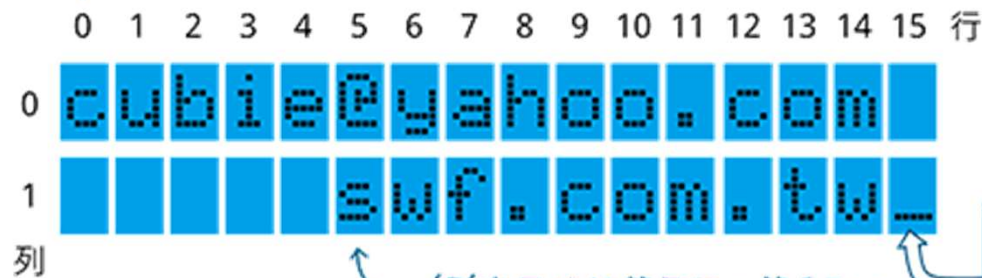




LCD顯示器程式說明

控制LCD模組的程式庫："LiquidCrystal.h"
LCD程式庫使用print()函數輸出文字。
插入文字的位置，由游標 (cursor) 決定。

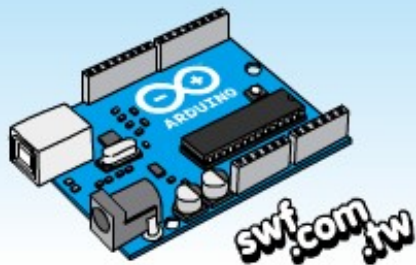
將游標設定在原點：
`lcd.home();`
或：
`lcd.setCursor(0, 0);`
或：
`lcd.clear();` ← 清除畫面並將游標重設回原點



先設定游標位置，再輸出文字：
`lcd.setCursor(5, 1);`
`lcd.print("swf.com.tw");`

游標相關函數：
`noCursor()` ● 不顯示游標
`cursor()` ● 顯示游標
`noBlink()` ● 不閃動游標
`blink()` ● 閃動游標
游標將出現在下一個可用位置

顯示文字暫存在控制晶片的記憶體裡，記憶體保存一列40個字。



顯示特殊符號與日文片假名

HD44780晶片的內建字體儲存在CGROM

字元'A'的編碼是0100 0001 (二進位) 或0x41 (16進位) 或65 (十進位)

Lower 4 Bits	Upper 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	(1)				0	a	P	`	P				-	9	ε	α	p
xxxx0001	(2)	!	1	A	a	9						α	7	4	ä	g	
xxxx0010	(3)	"	?	P	b	r						Γ	ι	υ	α	Α	
xxxx0011	(4)	-	=	M	I	n	/					α	z	\	∫	ε	÷
xxxx0100	(5)	.	>	N	^	n	+					α	ε	ホ	°	ñ	
xxxx0101	(6)	/	?	0	_	o	+					ウ	ソ	マ	°	ö	
xxxx0110	(7)																
xxxx0111	(8)																

這是溫度符號，1101 1111 (二進位) 或0xDF (16進位)

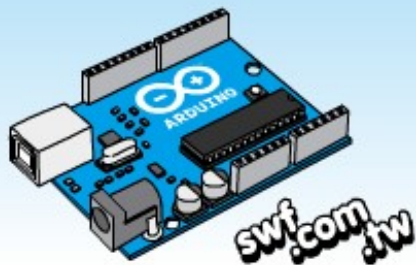
1100 1111
高位元 低位元

マ イ コ ン
0xCF 0xB2 0xBA 0xDD

```
char str[] = {'8', 'b', 'i', 't', ' ', 0xCF, 0xB2, 0xBA, 0xDD, 0};
lcd.print(str);
```

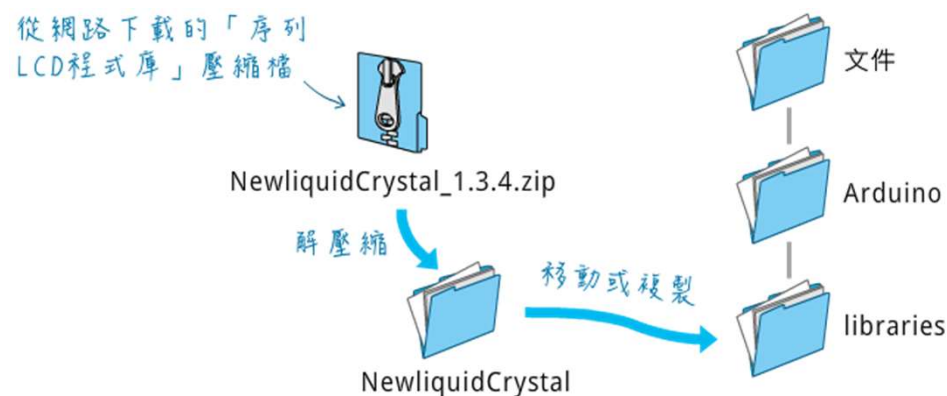
顯示
'8bit マイコン'

字串要用NULL結尾

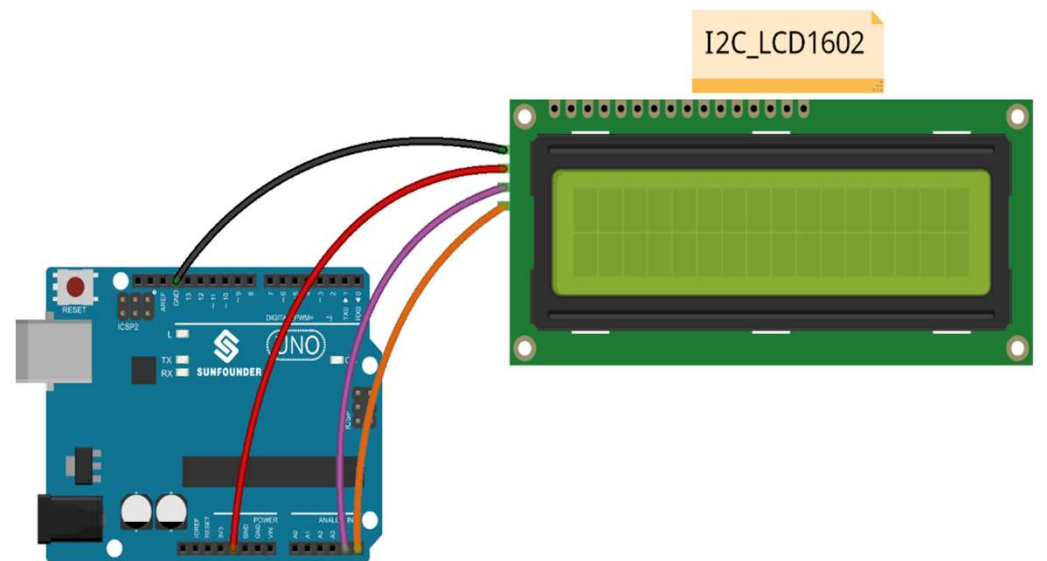
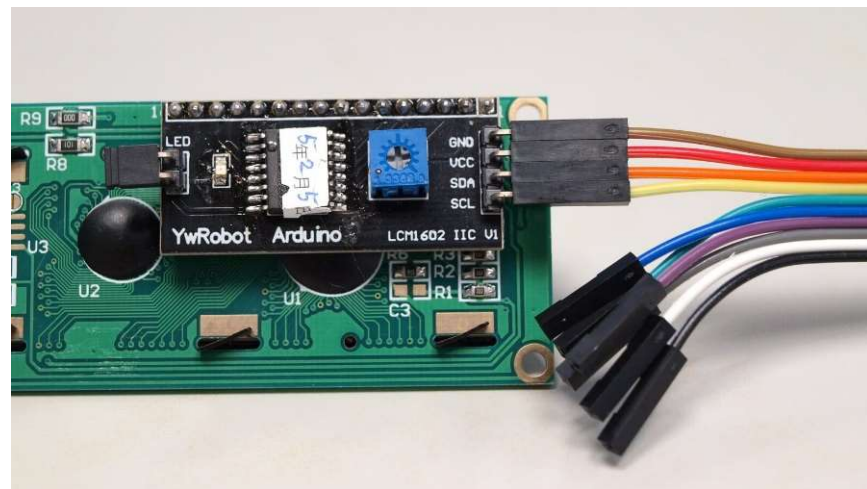
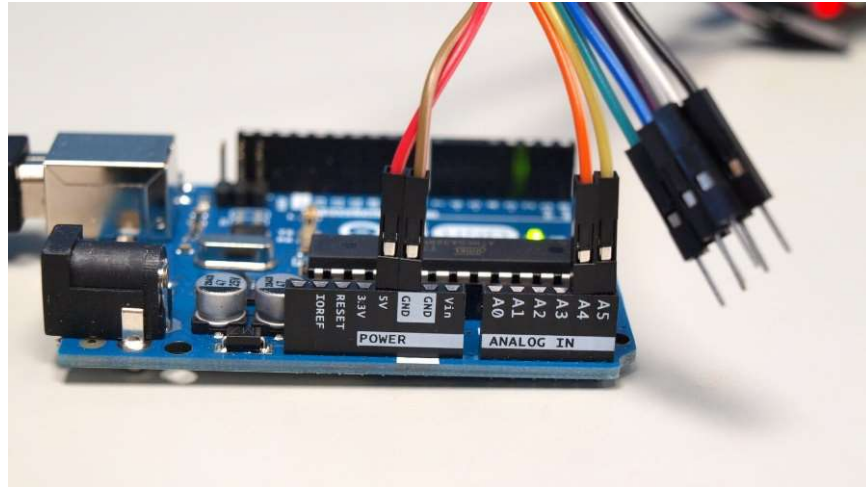
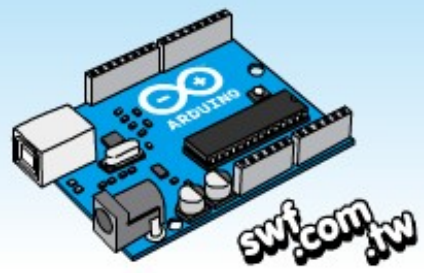


安裝與測試串連LCD模組的程式庫

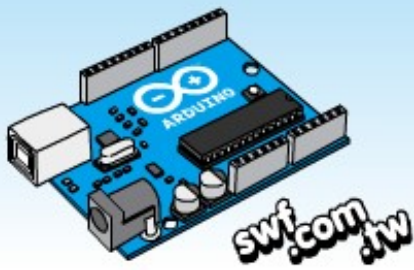
本單元採用兩線式LCD顯示模組的程式庫"[New LiquidCrystal](#)"，請先將原本的LiquidCrystal資料夾重新命名，再複製新的LiquidCrystal程式庫。



選擇主功能表的**檔案**→**範例**→**LiquidCrystal**→**HelloWorld_SR**，編譯並上傳測試。



fritzing



最前方定義完成LCD設定

```
#include <Wire.h>
```

```
#include <Wire.h> // Arduino IDE 內建
```

```
// LCD I2C Library，從這裡可以下載：
```

```
// https://bitbucket.org/fmalpartida/new-liquidcrystal/downloads---  
---只要下載NewliquidCrystal\_1.3.4.zip 即可
```

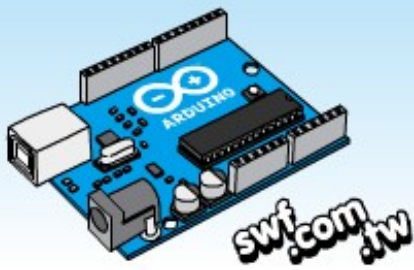
```
#include <LiquidCrystal_I2C.h> //網路下載
```

```
// Set the pins on the I2C chip used for LCD connections:
```

```
//設定 LCD I2C 位址 addr, en,rw,rs,d4,d5,d6,d7,bl,blpol
```

```
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
```

```
###若為16 pin LCD 則第一行改為 0x3F
```



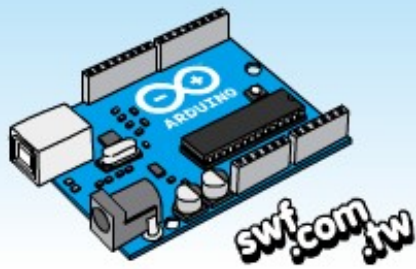
```
void setup() {  
  Serial.begin(9600); // 用於手動輸入文字  
  lcd.begin(16, 2);   // 初始化 LCD，一行 16 的字元，共  
  2 行，預設開啟背光
```

```
  // 閃爍三次  
  for(int i = 0; i < 3; i++) {  
    lcd.backlight(); // 開啟背光  
    delay(250);  
    lcd.noBacklight(); // 關閉背光  
    delay(250);  
  }
```

```
  lcd.backlight();
```

```
  // 輸出初始化文字直接顯示  
  lcd.setCursor(0, 0); // 設定游標位置在第一行行首  
  lcd.print("Good day!");  
  delay(1000);
```

```
}
```

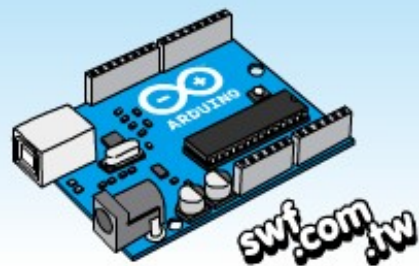


透過視窗輸入在LCD上顯示

```
#include <Wire.h> // Arduino IDE 內建
// LCD I2C Library · 從這裡可以下載：
// https://bitbucket.org/fmalpartida/new-liquidcrystal/downloads
#include <LiquidCrystal_I2C.h>

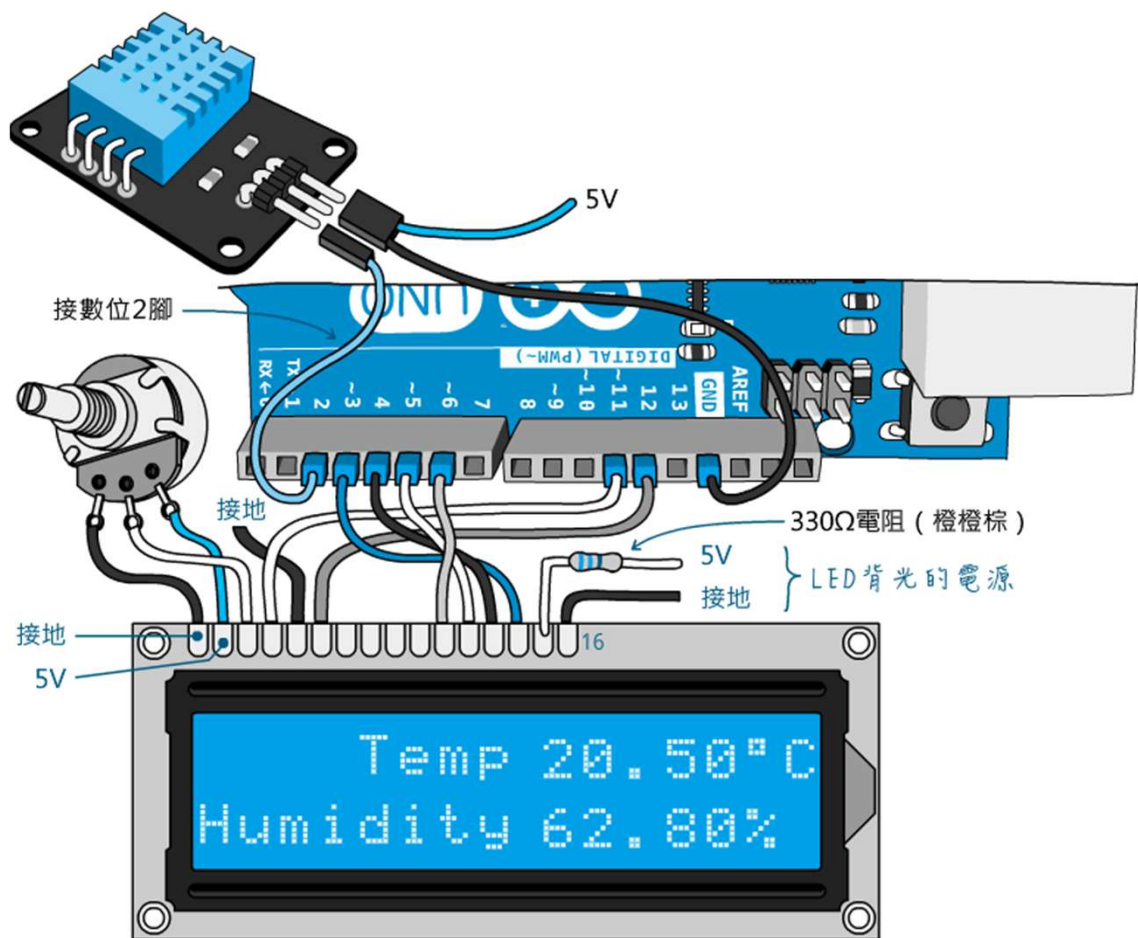
// Set the pins on the I2C chip used for LCD connections:
//          addr, en,rw,rs,d4,d5,d6,d7,bl,blpol
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); //
設定 LCD I2C 位址
char x = 0;
void setup() {
  Serial.begin(9600); // 用於手動輸入文字
  lcd.begin(16, 2); // 初始化 LCD · 一行 16 的字元 · 共 2 行 ·
  預設開啟背光

  // 閃爍三次
  for(int i = 0; i < 3; i++) {
    lcd.backlight(); // 開啟背光
    delay(250);
    lcd.noBacklight(); // 關閉背光
    delay(250); }
  lcd.backlight(); // 輸出初始化文字
  lcd.setCursor(0, 0); // 設定游標位置在第一行行首
}
void loop() {
  if(Serial.available()){
    char x = Serial.read();
    lcd.println(x);
    delay(1000); }
  lcd.clear();
}
```

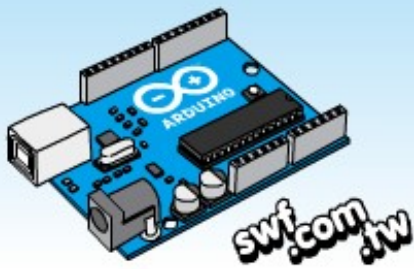



數位溫濕度感測器

DHT11是結合溫濕度感測器及訊號處理IC的感測模組

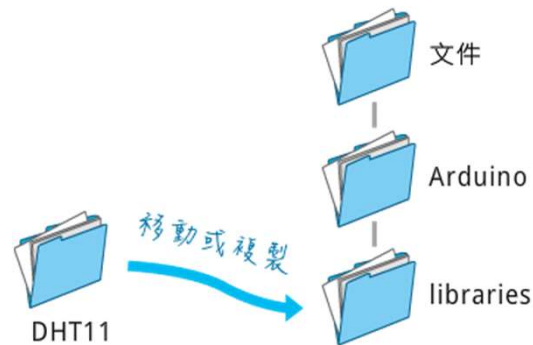


製作數位溫濕度顯示器



接收並顯示DHT11感測器 溫濕度值

請將DHT11程式庫資料夾複製到libraries裡面。

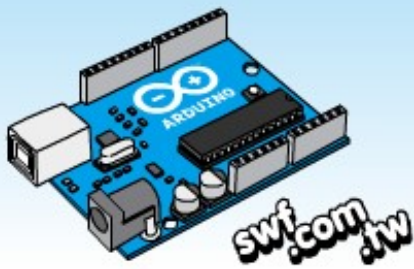


DHT11語法範例：

```
#include <dht11.h>

dht11 DHT11;

int chk = DHT11.read(2);    讀取數位2腳的DHT11資料，
                             如果傳回0，代表讀取成功。
float temp = DHT11.temperature;  讀取帶小數點的溫度值
float humi = DHT11.humidity;     讀取帶小數點的濕度值
```



用LCD顯示器呈現溫濕度值

透過LCD程式庫的setCursor()函數，可任意排列文字：

在第4行第0列，顯示"Temp"。

```
lcd.setCursor(4, 0);  
lcd.print("Temp");
```

```
lcd.setCursor(9, 0);
```

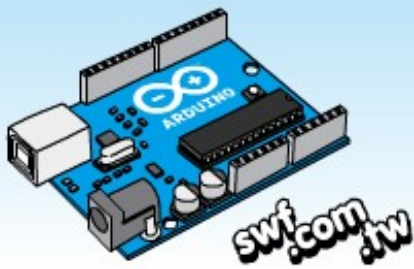
```
lcd.print("20.50");  
lcd.print((char) 0xDF);  
lcd.print("C");
```

顯示溫度符號

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	行
0					T	e	m	p		2	0	.	5	0	°	C	
1	H	u	m	i	d	i	t	y		6	2	.	8	0	%		

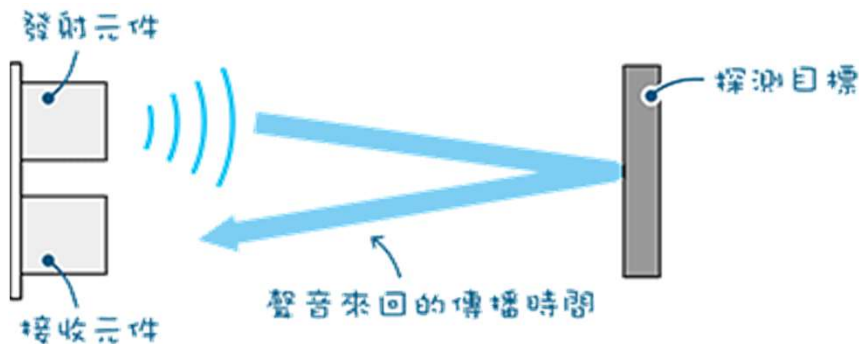
```
lcd.setCursor(0, 1);  
lcd.print("Humidity");
```

```
lcd.setCursor(9, 1);  
lcd.print("62.80");  
lcd.print("%");
```



認識超音波

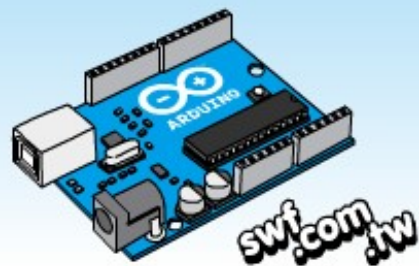
高於人耳可聽見的最高頻率以上的聲波，稱為超音波。



從聲音的傳播速度和傳播時間，可求出距離，而物體的實際距離是傳播時間的一半，從此可求得1公分距離的聲波傳遞時間約為58 μ s（微秒）

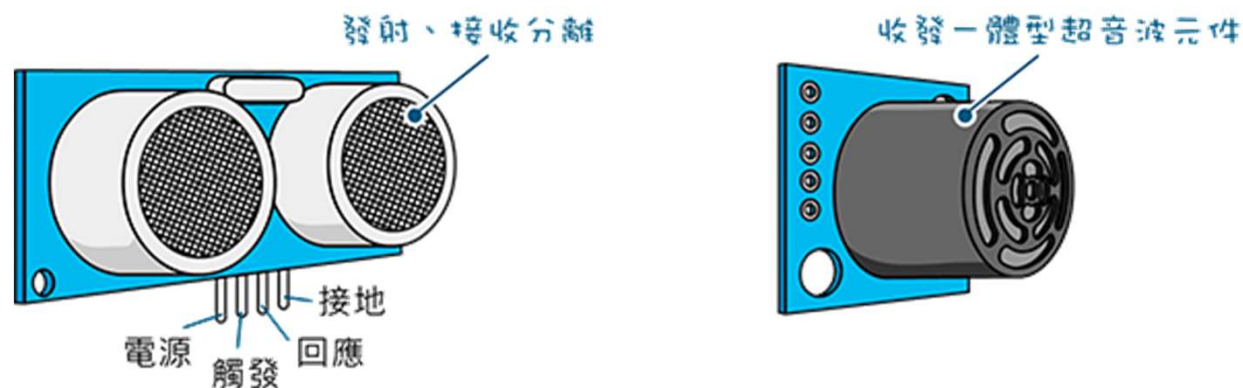
:

$$\begin{aligned} \text{距離} &= 344 \text{公尺/秒} \times \frac{\text{傳播時間}}{2} && \text{計算聲波前進1公分所需的時間} \\ &\uparrow && \\ &\text{聲波在室溫下，} && \\ &\text{空氣中的傳播速度} && \\ &\Rightarrow 0.01 \text{公尺} = \text{時間} \times 172 \text{公尺/秒} && \\ &\Rightarrow \text{時間} = \frac{0.01 \text{公尺}}{172 \text{公尺/秒}} && \\ &\Rightarrow \text{時間} \approx 58.1 \times 10^{-6} \text{秒} && \text{前進1公分所需的時間} \\ &&& \text{(單趟): } 58.1 \mu\text{s} \end{aligned}$$



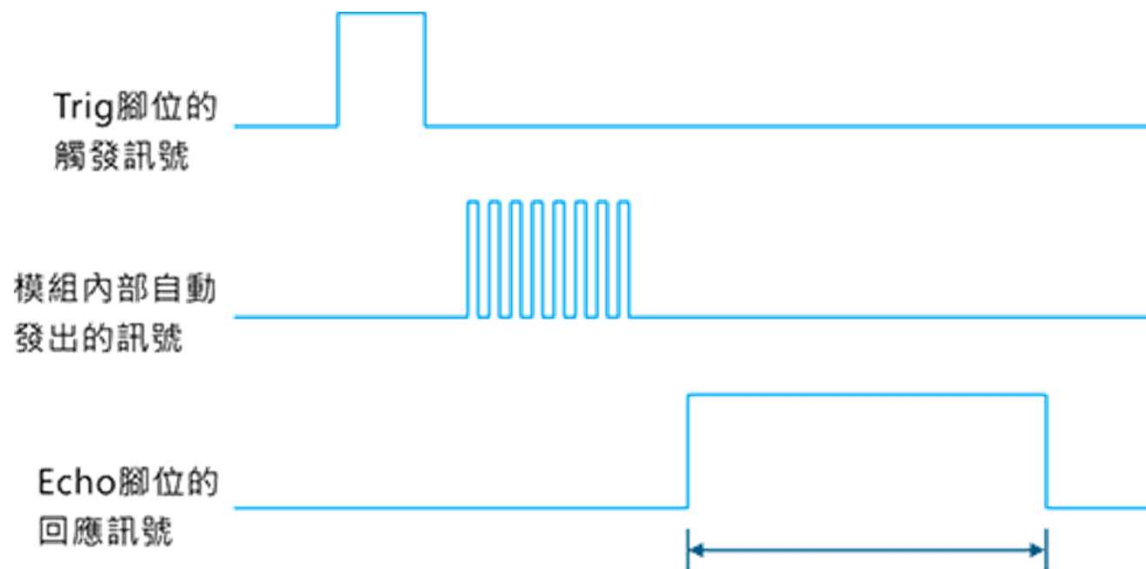
超音波感測器元件簡介

超音波感測器模組上面通常有兩個超音波元件，一個用於發射，一個用於接收；也有收發一體型。

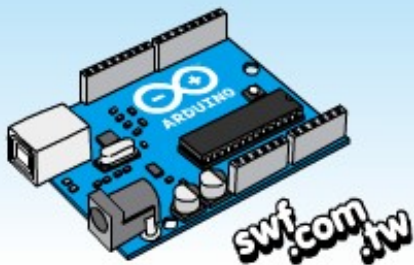


在「觸發」腳位輸入10微秒以上的高電位，即可發射超音波；

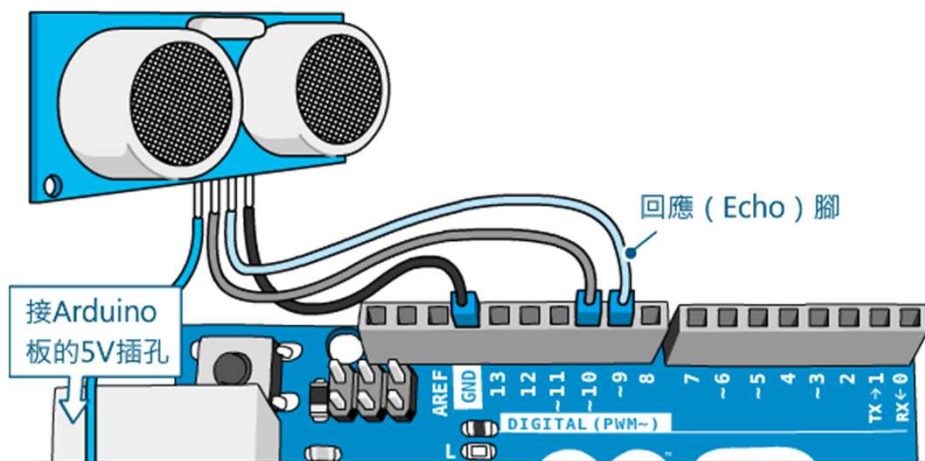
發射之後，接收到傳回的超音波之前，「回應」腳位將呈現高電位。



測量此脈衝的長度，即可得知被測物的距離。



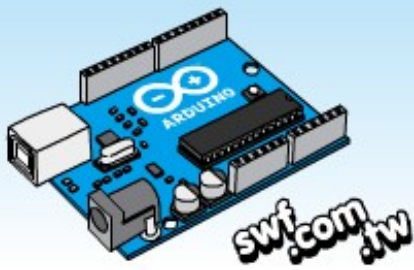
超音波數位量尺



藉由測量脈衝時間長度的pulseIn()函數，可得知超音波的回應時間並藉此計算距離。

```
// 將第9腳的高脈衝時間存入變數d  
unsigned long d = pulseIn(9, HIGH);
```

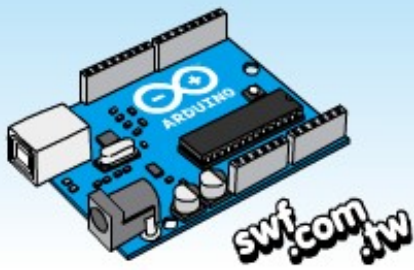




```
const int trig = 3;
const int echo = 4;
const int inter_time = 1000;
int time = 0;

void setup() {
  Serial.begin(9600);
  pinMode (trig, OUTPUT);
  pinMode (echo, INPUT);
}

void loop() {
  float duration, distance;
  digitalWrite(trig, HIGH);
  delayMicroseconds(1000);
  digitalWrite(trig, LOW);
  duration = pulseIn (echo, HIGH);
  distance = (duration/2)/29; //換成公分
  Serial.print("Data:");
  Serial.print (time/1000); // 第幾次(間隔一秒)
  Serial.print(", d = ");
  Serial.print(distance);
  Serial.println(" cm");
  time = time + inter_time;
  delay(inter_time);
}
```



- 一分鐘回饋:
- <https://goo.gl/forms/0C6jWOW5MTX9paos1>

