

# 海水水族箱 鹽度自動調節系統 三

經過硬件接駁，以及理解海水水族箱裡的鹽度知識後，今期餘下程式設計的部分。大家不要以為此部分很艱難，因為傳感器的基礎程式已有，大家只要稍作修改就可使用。

我們會先到傳感器的官方網站下載程式碼，然後啟動Arduino加以編譯就可使用。

**Step 1** 利用Arduino編程以使用Vernier傳感器的各種Sketches程式碼，可於Vernier公司的網站下載。

① <https://www.vernier.com/engineering/arduino>



```
char Sensor[]="Salinity Sensor";
float Intercept = 0;
float Slope = 16.3;
int ReadingNumber=0;
```

**1** 首先下載傳感器適用的Sketch程式碼，也就是Arduino的程式。再利用Arduino IDE開發環境修改。因每種傳感器的「Intercept」（攔截）及「Slope」（斜率）的值皆不一樣，需根據傳感器規格中列明的值作出修改。

```
void setup()
{
  Serial.begin(9600);
  pinMode(12, OUTPUT);
}
```

**2** 把繼電器連接到Arduino UNO的數位腳是12（D12），於程式的設置（setup）中把數位腳12定為輸出模式。

**Step 2** 程式碼並不複雜，僅34行左右。同學們只要理解當中每一個原則，就可按需要而設計及改動。

SalinityAutoBalance | Arduino 1.8.3

File Edit Sketch Tools Help

SalinityAutoBalance

```
char Sensor[]="Salinity Sensor";
float Intercept = 0;
float Slope = 16.3;
int ReadingNumber=0;
```

void setup()

```
{
  Serial.begin(9600);
  pinMode(12, OUTPUT);
}
```

void loop()

```
{
  float Count = analogRead(A0);
  float Voltage = Count / 1023 * 5.0;
  float SensorReading= Intercept + Voltage * Slope;
```

```
Serial.println(SensorReading);
freshwaterpump();
}
```

void freshwaterpump()

```
{
  float Count = analogRead(A0);
  float Voltage = Count / 1023 * 5.0;
  float SensorReading= Intercept + Voltage * Slope;
```

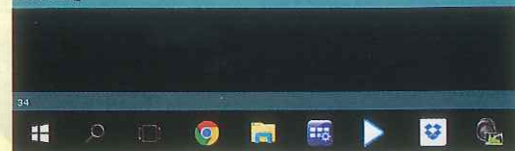
```
if(SensorReading >= 36){
  digitalWrite(12, HIGH);
  delay(10000);
  digitalWrite(12, LOW);
  delay(30000);
}
```

else

```
{
  digitalWrite(12, LOW);
}
```

```
delay(1000);
}
```

Done Saving.





## 陳璋麟

仁濟醫院靚次伯紀念中學  
綜合科學科主任

任教綜合科學科及生物科，畢業於香港大學生物技術學。喜發掘新奇有趣的科學新知與同學分享，以多元化模式讓學生接觸科學。

```

void loop()
{
  float Count = analogRead(A0);
  float Voltage = Count / 1023 * 5.0;
  float SensorReading= Intercept + Voltage * Slope;

  Serial.println(SensorReading);
  freshwaterpump();
}

```

```

void freshwaterpump()
{
  float Count = analogRead(A0);
  float Voltage = Count / 1023 * 5.0;
  float SensorReading= Intercept + Voltage * Slope;

  if(SensorReading >= 36){
    digitalWrite(12,HIGH);
    delay(10000);
    digitalWrite(12,LOW)
    delay(30000);
  }
  else
  {
    digitalWrite(12,LOW);
  }
  delay(1000);
}

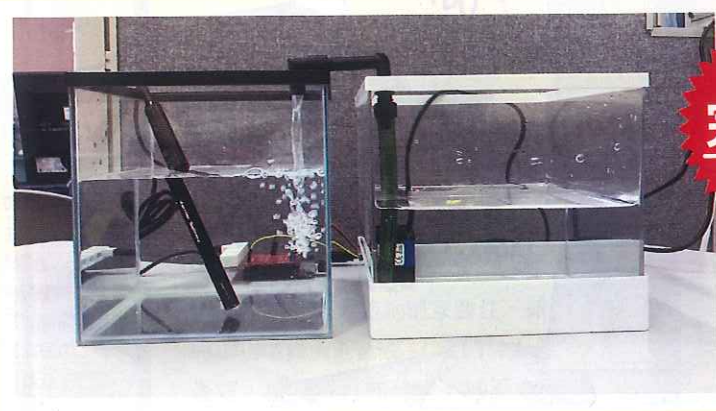
```

**3** 接著到程式運作，主要不斷地呼叫的部份 (loop)，當中分為兩部分：

- ① Sketch中已編好Arduino UNO底板會從傳感器，傳來的原始數據換算成正確讀數 (鹽度)；設定「SensorReading」變量回傳所測試出的鹽度。
- ② 「freshwaterpump ()」是我們自行編寫的一個程式碼模組，亦是這程式主要運行的部分之一。

**4** 我們利用了「if...else」的編程概念，去決定水泵的開關。如果 (if) 鹽度感測器測出的數值附合預設條件，程式便指示水泵開動；但在所有其他情況下 (else)，程式均會示水泵停止。

- ① 設定的條件是當鹽度讀數大於或等於36ppt時，程式指示水泵開動。程式會指示Arduino Uno的數位腳12 (D12) 輸出高電壓 (HIGH)，如此會令繼電器把水泵的電源接通，水泵就會開動。
- ② 當淡水加進水族箱後，需要一定時間才能與海水完全混合，所以我們於加入一定分量的淡水後，必須讓海水的鹽度有足夠的時間穩定下來，然後才作第二次的量度及操作。我們會利用「delay」這個延時指令去處理這情境，然後用「digitalWrite (12,HIGH) ;」、「delay (10000) ;」和「digitalWrite (12,LOW) ;」三組指令，讓水泵開動加入淡水10,000毫秒 (即10秒) 後停止。
- ③ 接著我們利用「delay (30000) ;」指令，提供30,000毫秒 (即30秒) 的時間，讓淡水與海水有充份時間完全混合，才開始下一次的量度及操作。(註：水泵開動和等待下一次量度的時間，需因應水族箱內水體的大小作調節。)



## 完成圖

利用此裝置就能時刻確保海洋生態水族箱內的海水鹽度，維持在一個穩定的水平，這對養殖海洋生物會十分便利及安全。📌