

舵機控制函數介紹

本實驗為了讓讀者更加了解舵機的控制，本章節針對 Arduino Reference 網站中的 Servo library 網頁(<http://arduino.cc/en/Reference/Servo>)，對下列舵機控制的函式庫加以解釋，讓讀者可以更容易使用。

Servo librar 有以下成員函數：

`attach()`

`write()`

`writeMicroseconds()`

`read()`

`attached()`

`detach()`

attach()

指令用法

將舵機的控制線(Signal Pin)連接到 Arduino 開發版的 Digital Pin 的設定：

語法：`servo.attach(pin)`

`servo.attach(pin, min, max)`

參數

`servo`: Servo 形態的變數

`pin`: 舵機的控制線(Signal Pin)連接到 Arduino 開發版的 Digital Pin

`min`: 選擇性參數，為輸出控制 pulse 的寬度,用 microseconds 計算, 對應舵機最小的角度(0-degree)

`max` : 選擇性參數，為輸出控制 pulse 的寬度,用 microseconds 計算, 對應舵

機最大的角度 (180-degree)

回傳：無回傳資料

範例：

```
#include <Servo.h>

Servo myservo;

void setup()
{
  myservo.attach(9);
}

void loop() {}
```

write()

指令用法

透過這個函數，將舵機的控制透過寫入 **angle** 角度的參數，將舵機轉動到 **angle** 角度的參數所指定的角度，

對連續旋轉的舵機而言，輸入 90 度，則舵機不旋轉，輸入 0 度則往逆時鐘方向全力運轉，輸入 180 度則往順時鐘方向全力運轉。

語法：servo.write(angle)

參數

servo: Servo 形態的變數

angle: 控制舵機轉動的角度，通常為 0~180 度

回傳：無回傳資料

範例：

```
#include <Servo.h>

Servo myservo;
```

```
void setup()
{
  myservo.attach(9);
  myservo.write(90); // set servo to mid-point
}

void loop() {}
```

writeMicroseconds ()

指令用法

透過這個函數，將舵機的控制透過寫入 uS 毫秒的參數，將舵機轉動到 uS 毫秒的參數所對應的角度，

對標準的舵機而言，輸入 1500 us，則舵機不旋轉，輸入 1000 us 往逆時鐘方向全力運轉，輸入 2000 us 則往順時鐘方向全力運轉

語法：servo.writeMicroseconds(uS)

參數

servo: Servo 形態的變數

uS:為輸出控制舵機轉動的角度，輸入 1500 us，則舵機不旋轉，輸入 1000 us 往逆時鐘方向全力運轉，輸入 2000 us 則往順時鐘方向全力運轉

回傳：無回傳資料

範例：

```
#include <Servo.h>

Servo myservo;

void setup()
{
```

```
myservo.attach(9);  
myservo.writeMicroseconds(1500); // set servo to mid-point  
}  
  
void loop() {}
```

read()

指令用法

讀取目前舵機的角度：

語法：servo.read()

參數

servo: Servo 形態的變數

回傳：回傳目前舵機的角度，為 0 到 180 度。

範例：

```
#include <Servo.h>  
  
Servo myservo;  
  
void setup()  
{  
  Serial.begin(9600);  
  myservo.attach(9);  
}  
  
void loop() {  
  Serial.println(myservo.read());  
}
```

attached()

指令用法

檢查舵機與控制線(Signal Pin)連接是否連接成功：

語法：servo.attached()

參數

servo: Servo 形態的變數

回傳：回傳 **true** 代表連接成功，回傳 **false** 代表連接失敗

範例：

```
#include <Servo.h>

Servo myservo;

void setup()
{
  Serial.begin(9600);
  myservo.attach(9);
}

void loop() {
  Serial.println(myservo.attached());
}
```

detach()

指令用法

將舵機與控制線(Signal Pin)連接解除：

語法：servo.detach()

參數

servo: Servo 形態的變數

回傳：無

範例：