

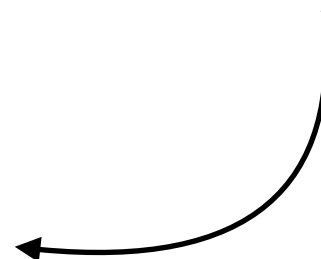
Week 4

Howard Ke

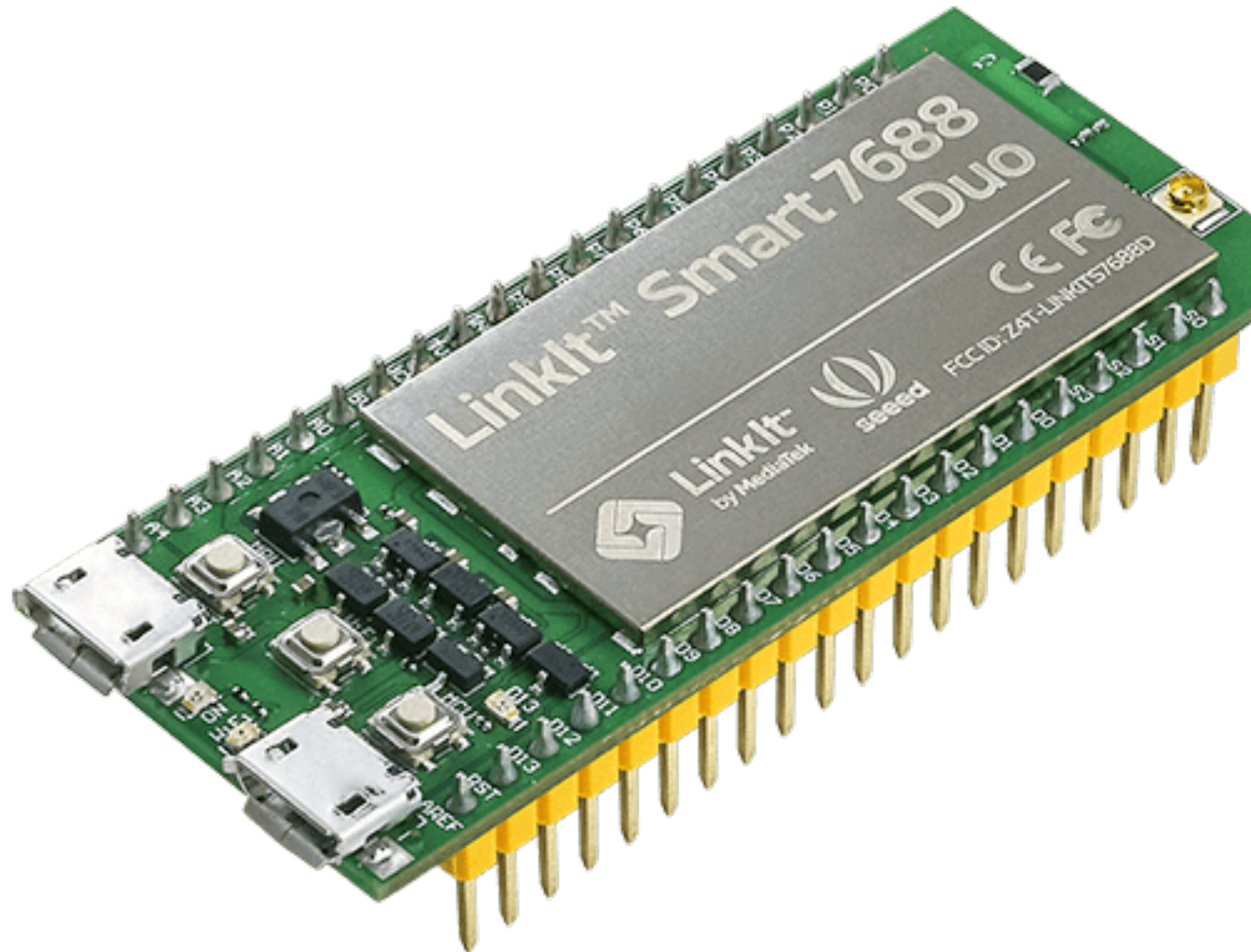
<https://goo.gl/eXaR9G>



投影片檔案



linkit 7688



安裝環境

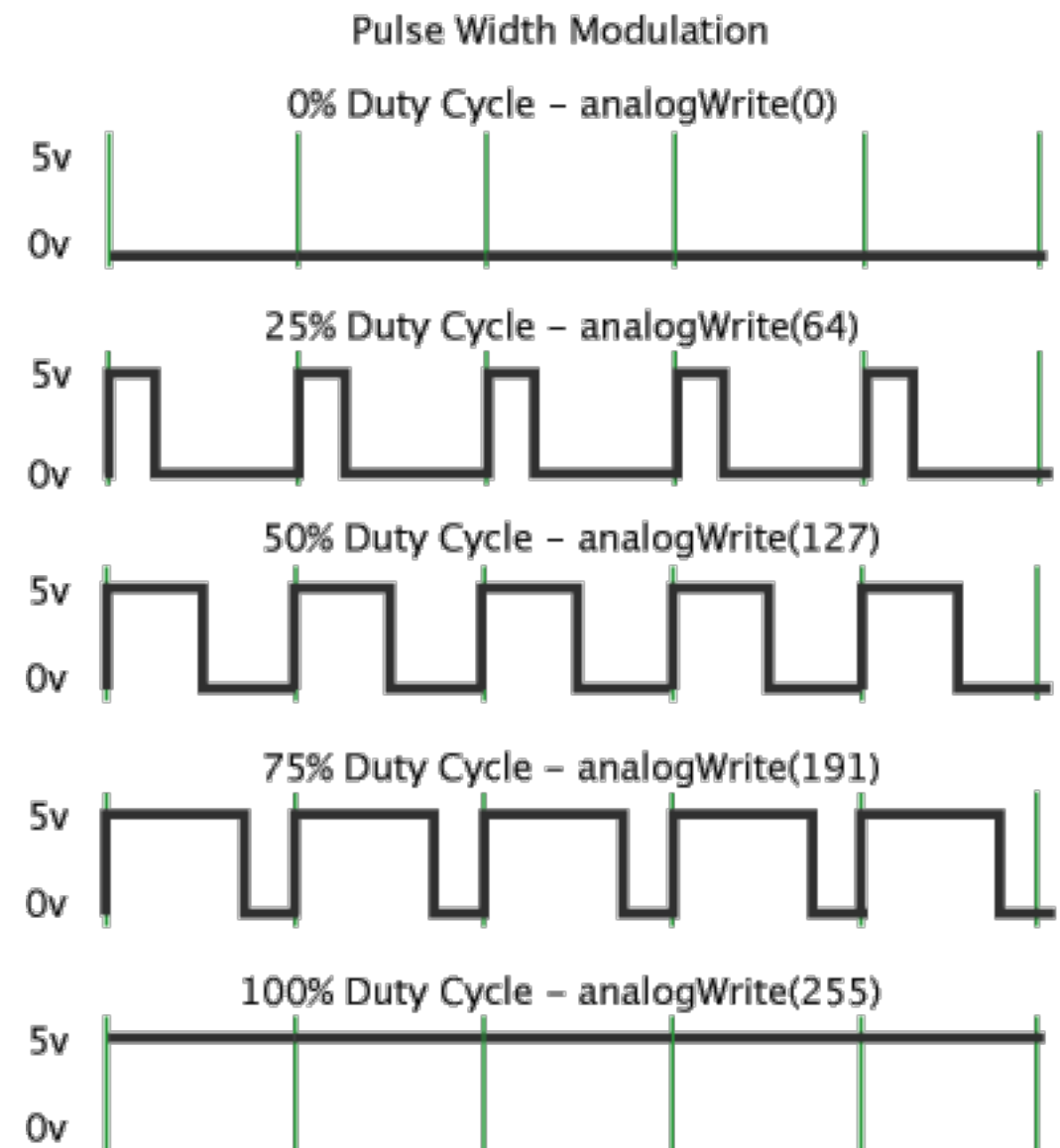
- 把以下網址加入board manager:
 - http://download.labs.mediatek.com/package_mtk_linkit_smart_7688_index.json
- 教學：<https://goo.gl/9LLvgV>

PWM

我要怎麼控制亮度？

PWM (Pulse Width Modulation)

- 用數位輸出模擬類比輸出
- 迅速的開關，用通電的時間比例控制輸出能量
- Arduino - `analogWrite`



PWM (Pulse Width Modulation)

- 腳位
 - 旁邊有 ~ 符號
 - Arduino UNO: 3, 5, 6, 9, 10, 11
- `analogWrite(pin, value)`
 - 輸入範圍 0(全暗) ~ 255(全亮)
 - Duty cycle: $\text{<input>} / 256 * 100\%$

PWM

```
int led_pin = 9;

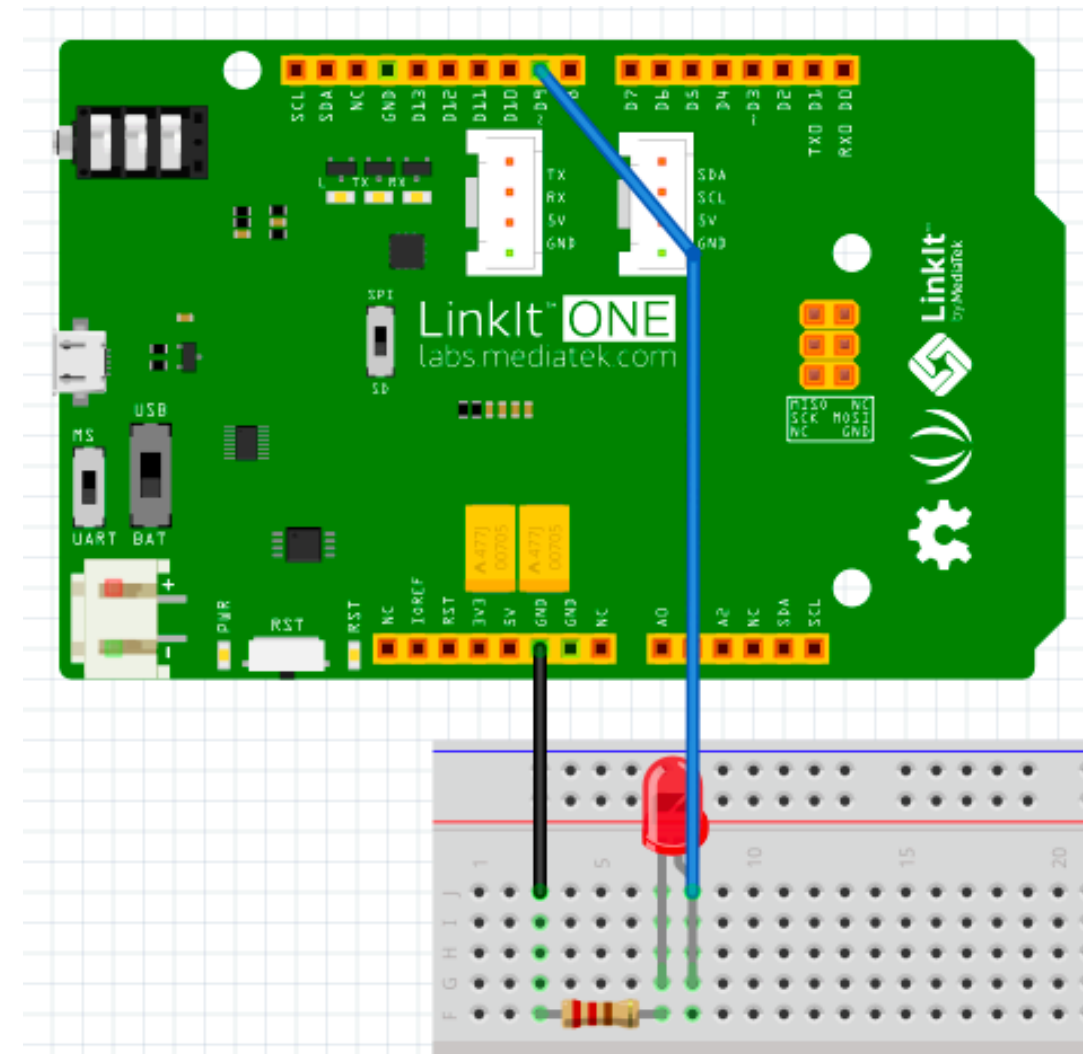
void setup() {
    pinMode(led_pin, OUTPUT);
}

void loop() {
    analogWrite(led_pin, 64);
    delay(1000);
    analogWrite(led_pin, 255);
    delay(1000);
}
```

Lab 01

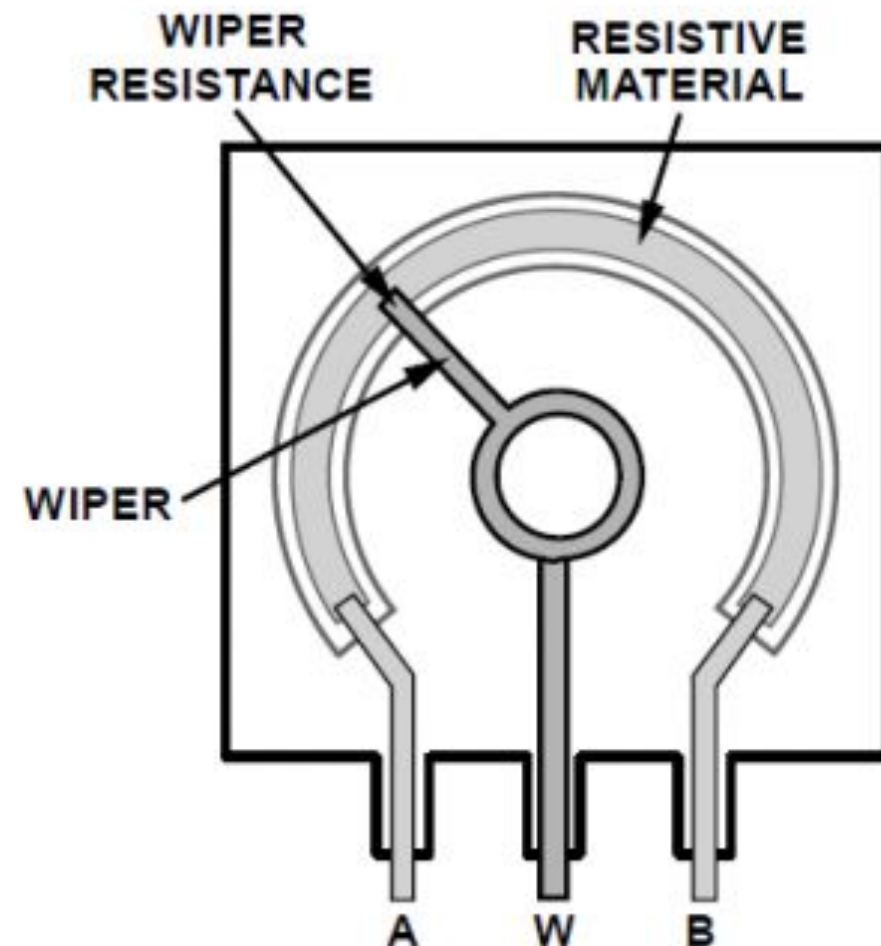
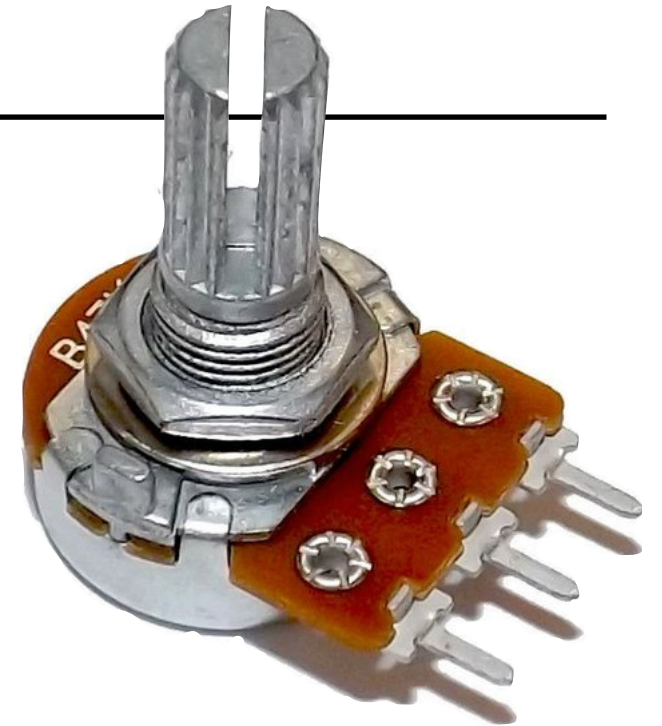
Goal: 呼吸燈

1. 讓LED從暗漸漸變亮



可變電阻

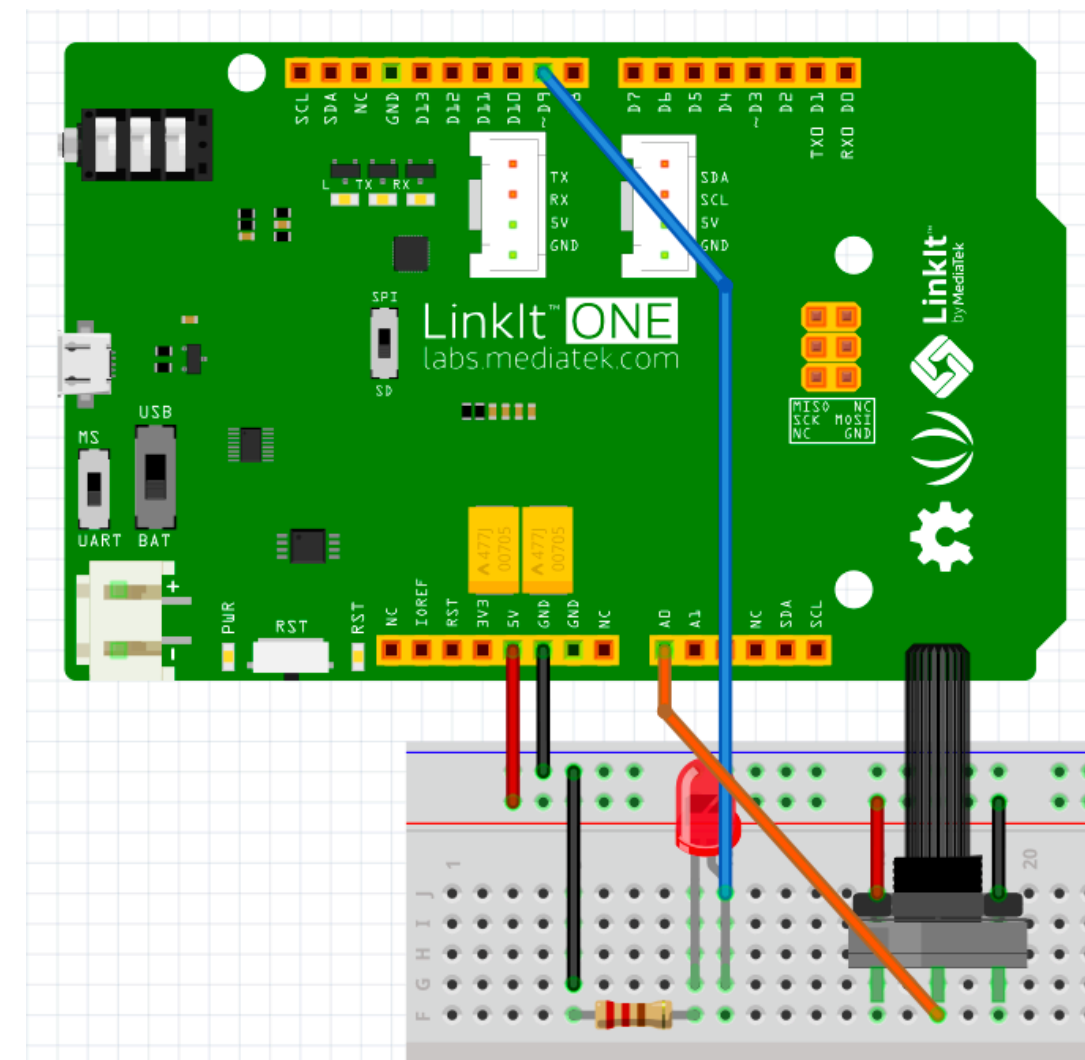
- 兩個固定接點+一個滑動接點
- 透過改變滑動端和固定端間距離改變電阻值
- 用途：
 - 音量控制
 - 位置、距離傳感器



Lab 02

Goal: 可變電阻控制LED

1. 用Arduino讀取可變電阻電壓
2. 輸出對應的PWM訊號控制LED燈亮度
3. bonus: 反向



Lab 02 - code

```
int led_pin = 9;
void setup() {
    pinMode(led_pin, OUTPUT);
}

void loop() {
    int sensorValue = analogRead(A0);

    analogWrite(led_pin, sensorValue);
    delay(150);
}
```

digitalRead

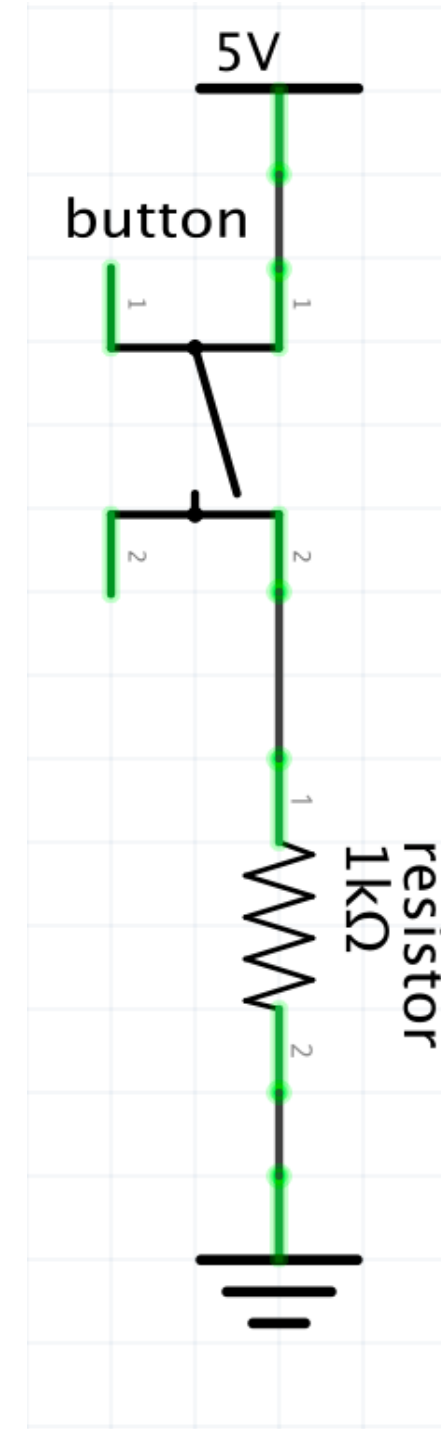
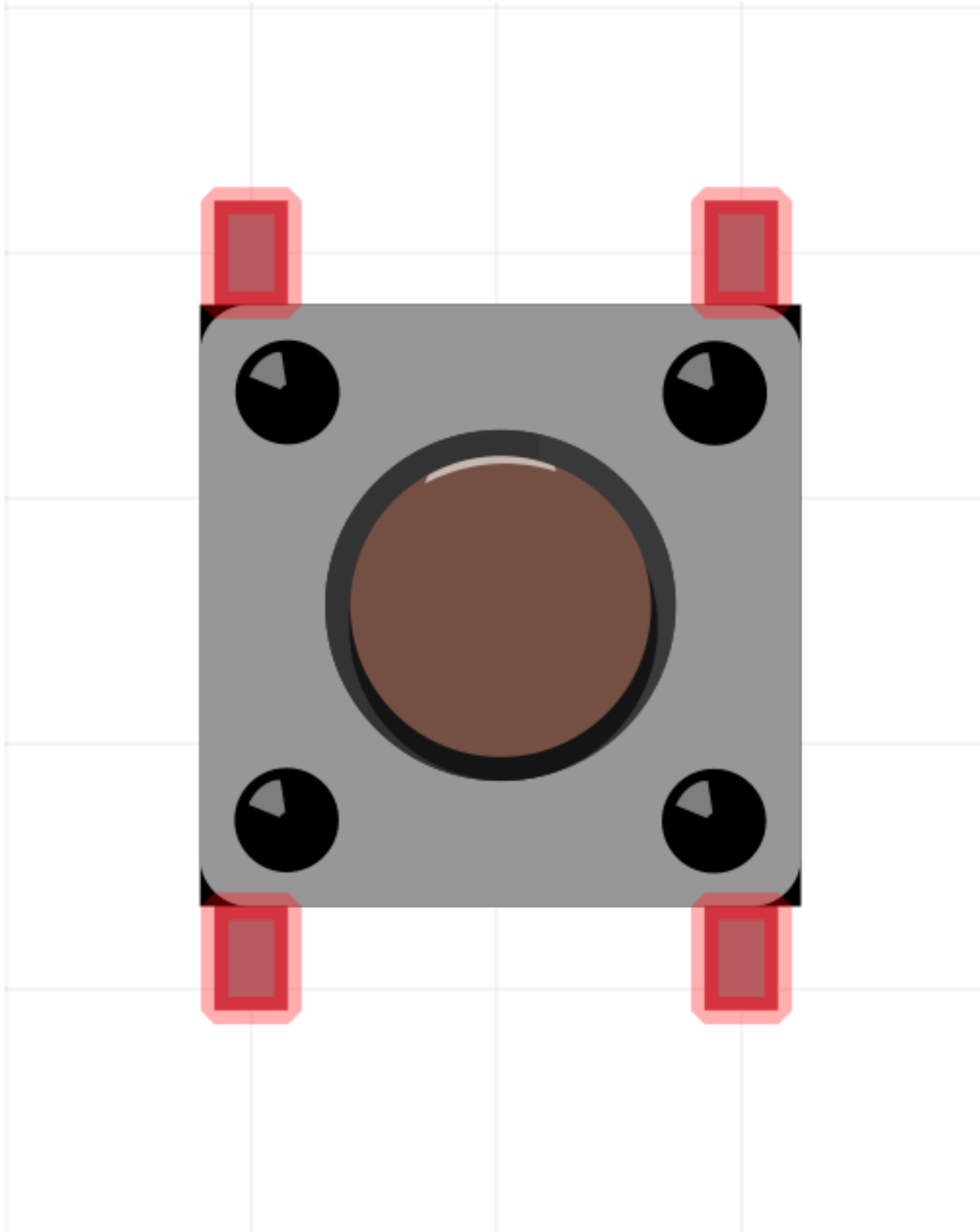
有analogWrite

有digitalWrite

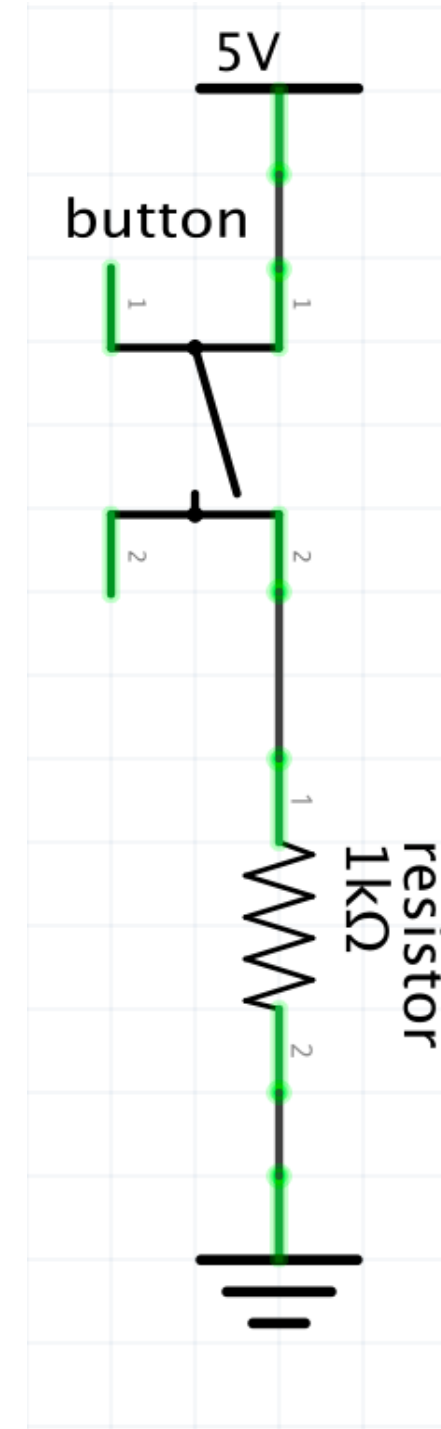
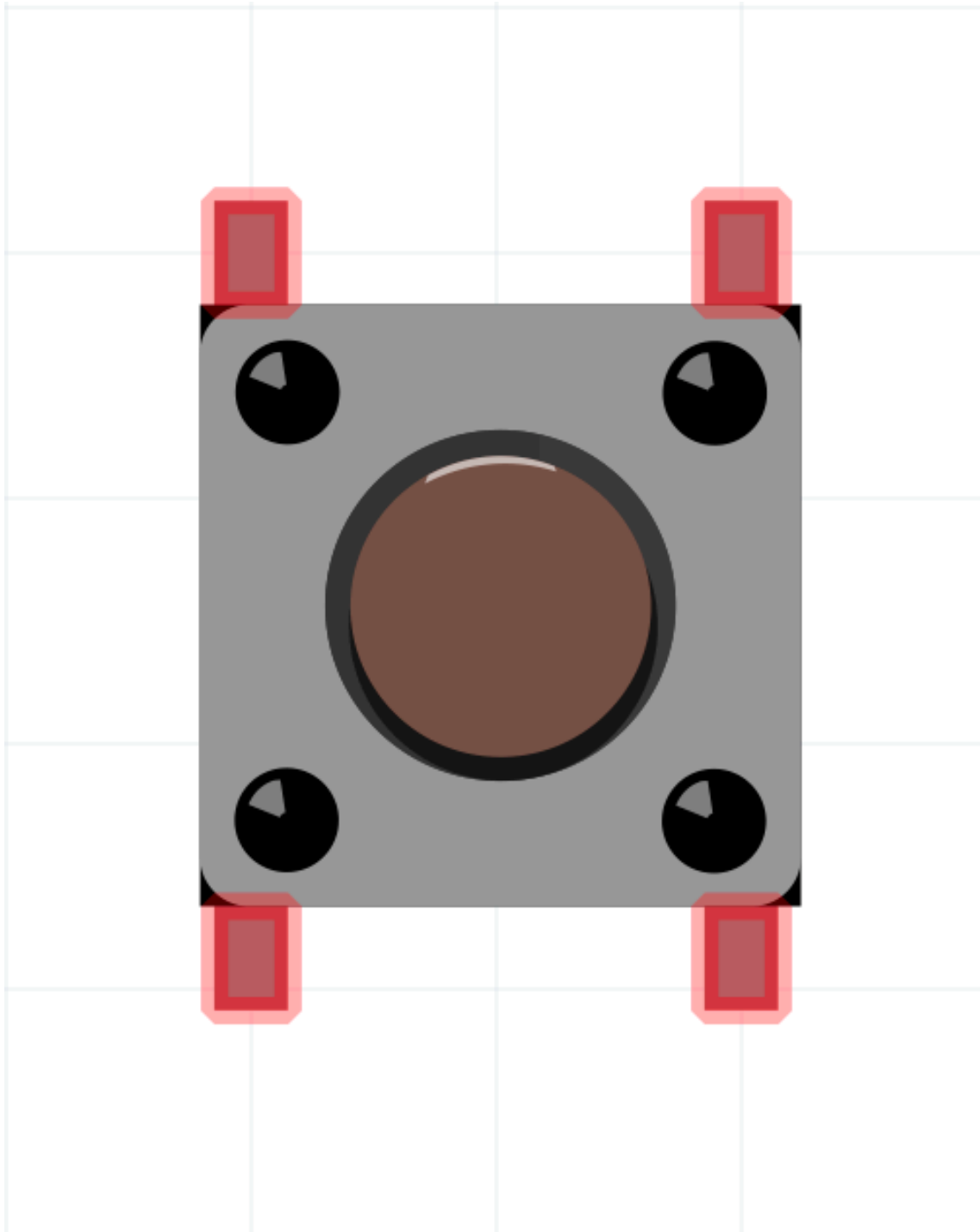
有analogRead

那有沒有digitalWrite？

digitalRead



digitalRead



digitalRead

```
int led_pin = 9;
int button_pin = 7;

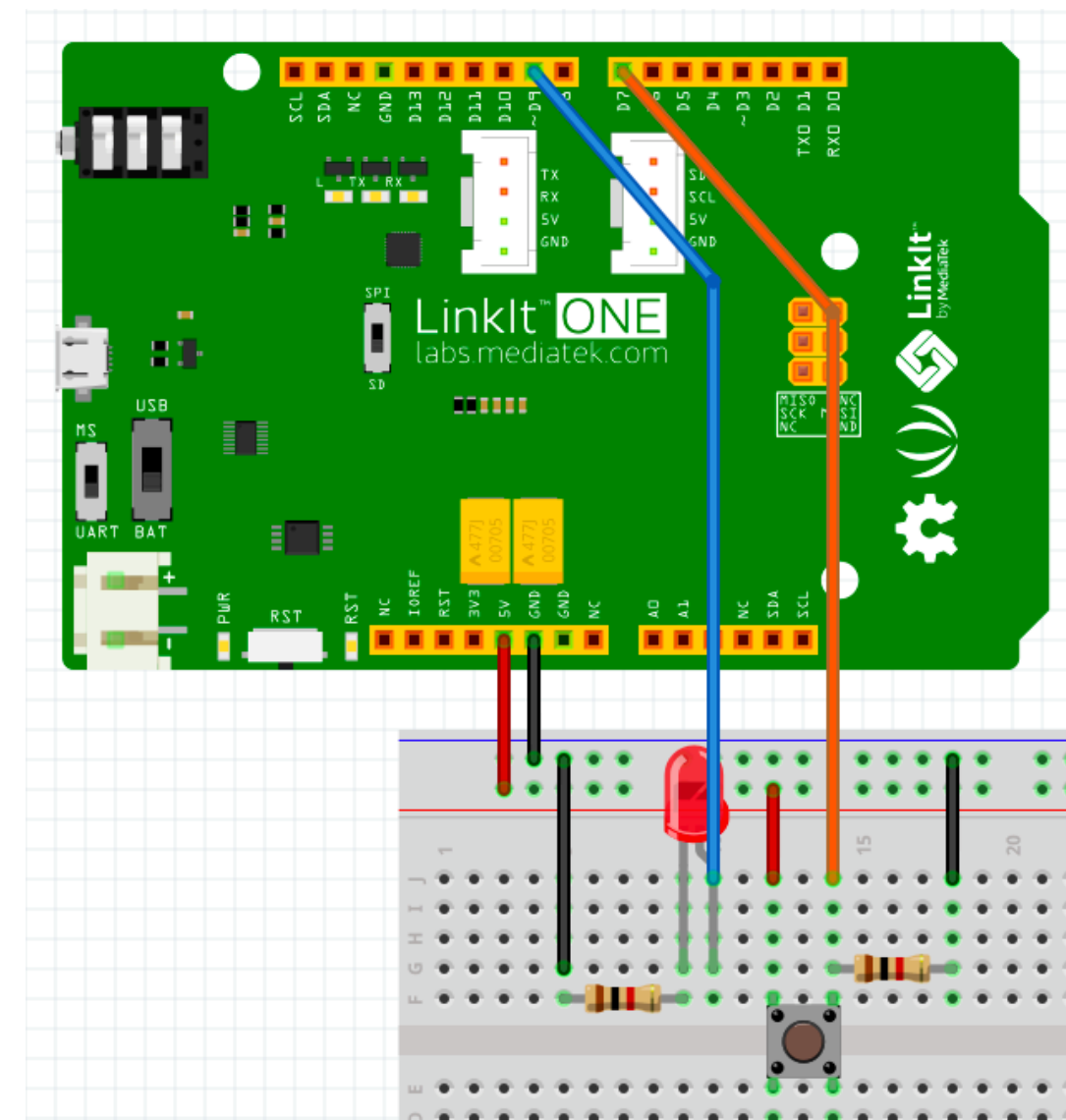
void setup() {
    pinMode(led_pin, OUTPUT);
    pinMode(button_pin, INPUT);
}

void loop() {
    int value = digitalRead(button_pin);
    digitalWrite(led_pin, value);
    delay(10);
}
```

Lab 03

Goal: 按鈕控制LED

1. bonus: sticky LED
(按下切換狀態)

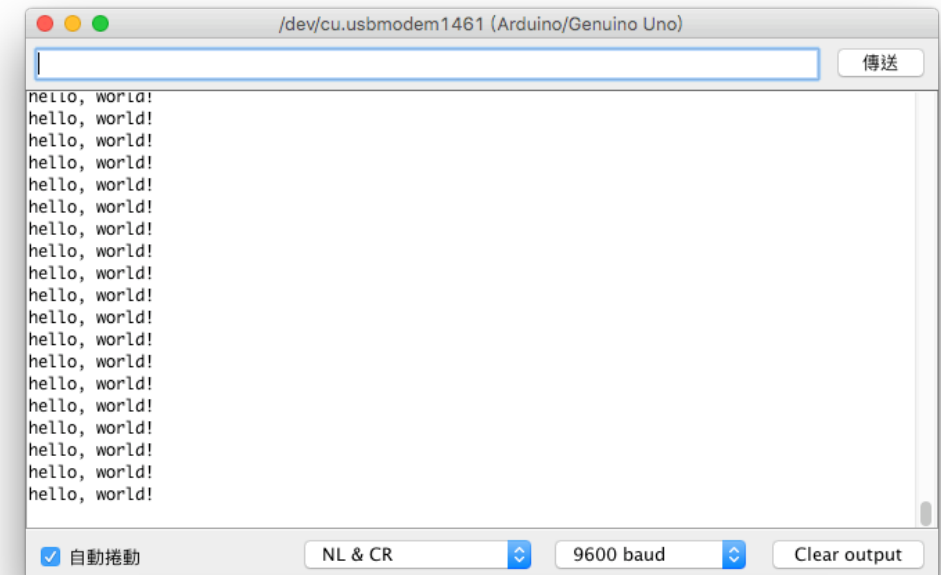


Serial Monitor

要怎麼看到值？

Serial Monitor

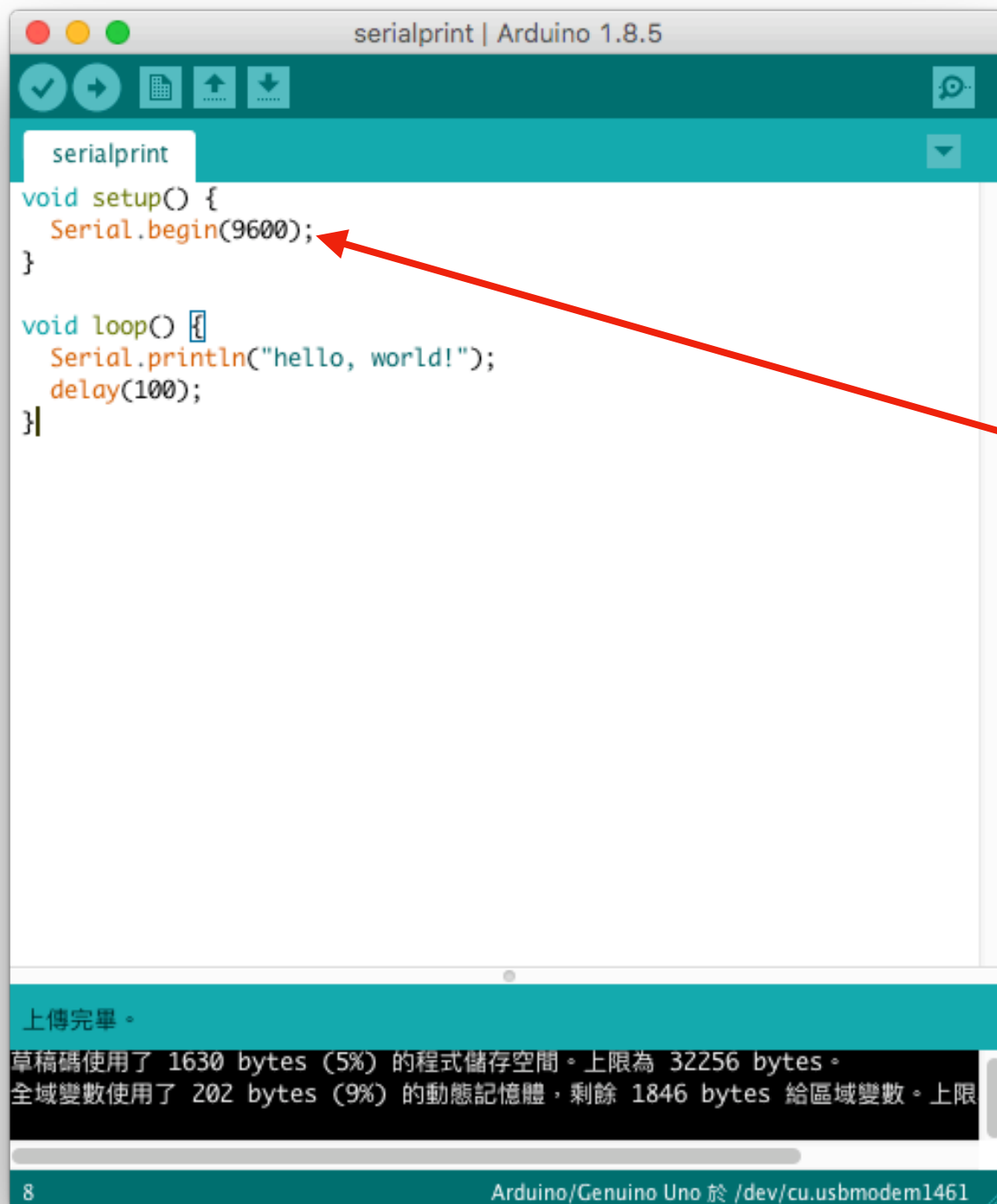
Arduino 透過 USB 作串列傳輸，與電腦互傳資料
可以透過 Arduino IDE 的 Serial Monitor 進行操作



Serial Monitor

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    Serial.println("Hello, world ");  
    delay(1000);  
}
```

Serial Monitor



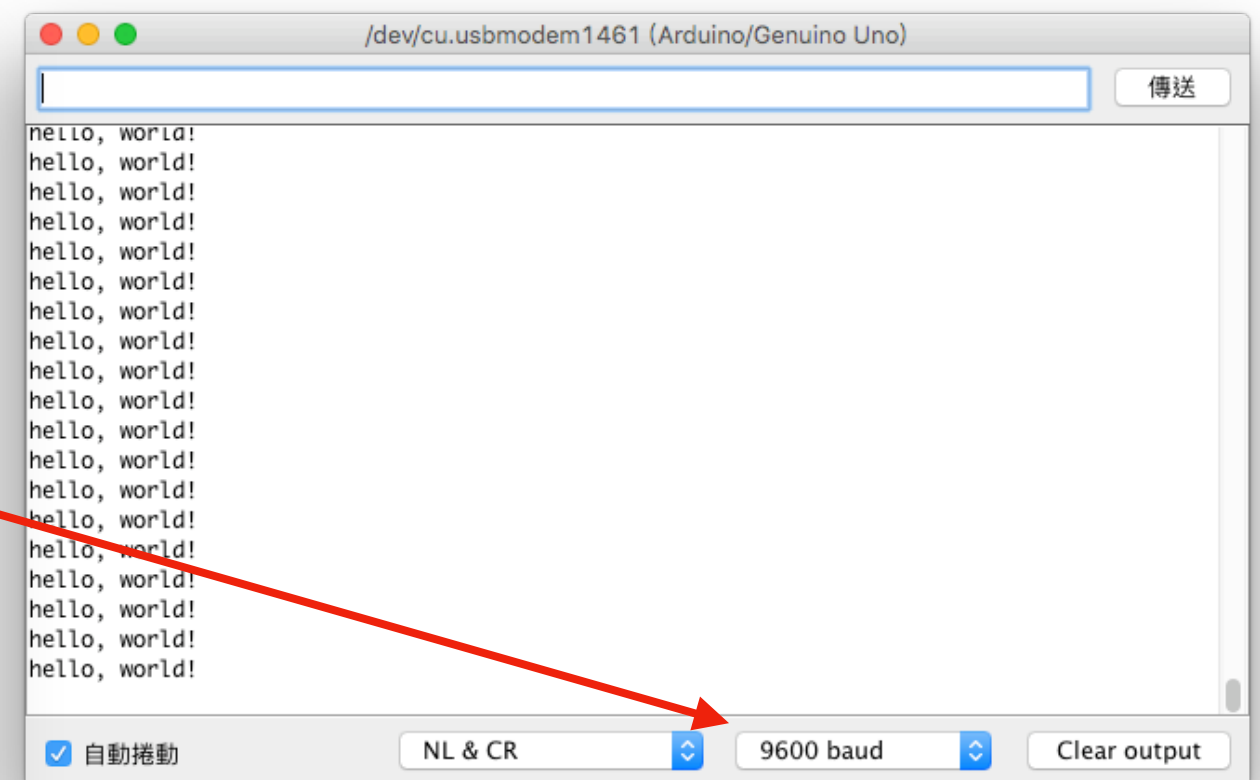
```
serialprint | Arduino 1.8.5  
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  Serial.println("hello, world!");  
  delay(100);  
}
```

上傳完畢。

草稿碼使用了 1630 bytes (5%) 的程式儲存空間。上限為 32256 bytes。

全域變數使用了 202 bytes (9%) 的動態記憶體，剩餘 1846 bytes 給區域變數。上限

8 Arduino/Genuino Uno 於 /dev/cu.usbmodem1461



baud rate 要一樣

Serial.print / Serial.println

Serial.print 不換行
Serial.println 會換行



The screenshot shows the Arduino IDE interface with a sketch named "serialprint" open. The code defines a variable `worldCount` and uses `Serial.print` and `Serial.println` in the `loop` function. A status bar at the bottom indicates the upload is complete and shows memory usage statistics.

```
serialprint | Arduino 1.8.5  
  
int worldCount;  
  
void setup() {  
  worldCount = 0;  
  Serial.begin(9600);  
}  
  
void loop() {  
  Serial.print("Hello, world ");  
  Serial.println(worldCount++);  
  delay(1000);  
}
```

上傳完畢。

草稿碼使用了 1914 bytes (5%) 的程式儲存空間。上限為 32256 bytes。

全域變數使用了 204 bytes (9%) 的動態記憶體，剩餘 1844 bytes 給區域變數。上限

9 Arduino/Genuino Uno 於 /dev/cu.usbmodem1461



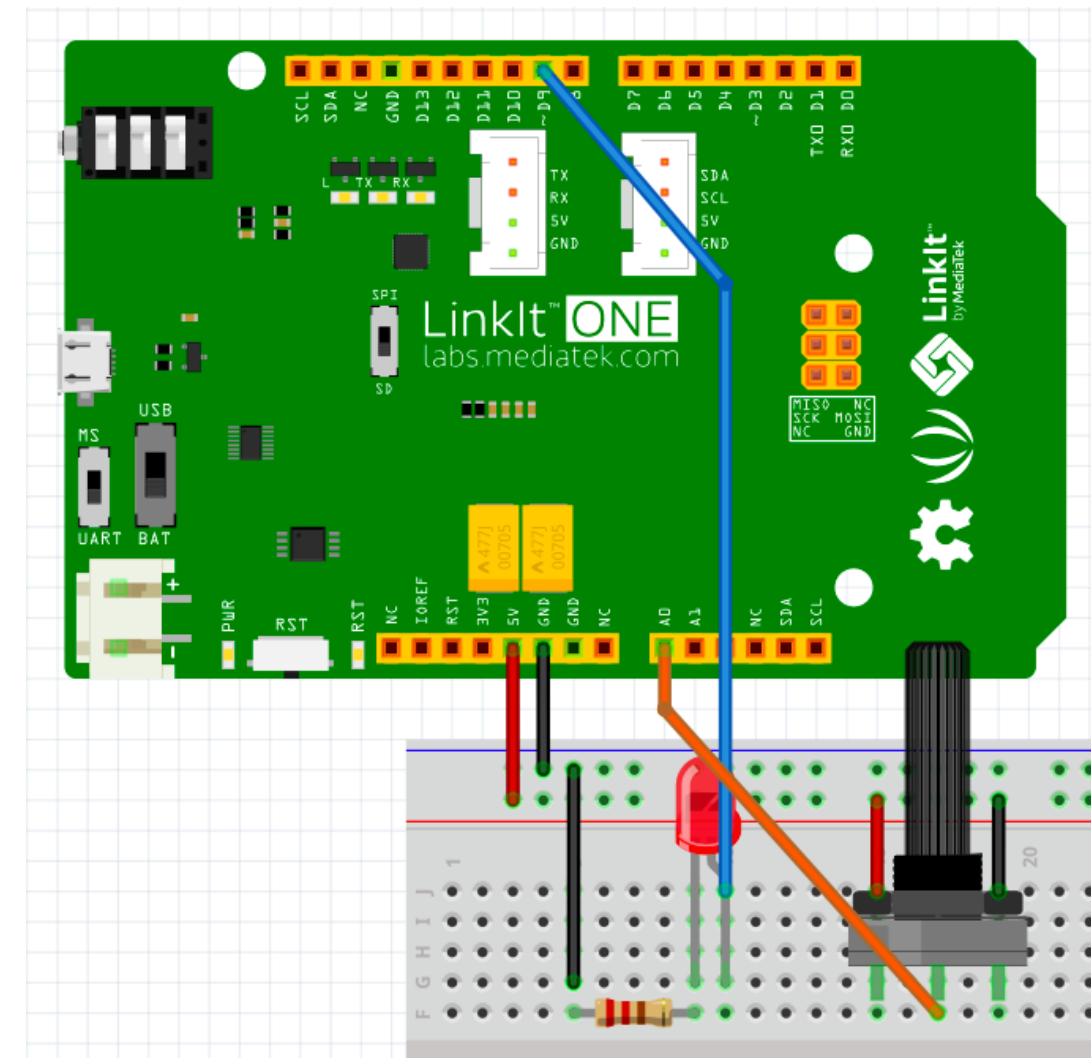
The serial monitor window shows the output of the sketch, displaying "Hello, world" followed by the counter value from 0 to 4, each on a new line.

```
/dev/cu.usbmodem1461 (/dev/cu.usbmodem1461)  
  
Hello, world 0  
Hello, world 1  
Hello, world 2  
Hello, world 3  
Hello, world 4
```

Lab 04

Goal: 可變電阻控制LED

跟lab02一樣，但要在 Serial Monitor 顯示讀到的值



function

我覺得你們的 code 很亂。

function

```
void setup() {  
    Serial.begin(9600);  
}
```

```
void loop() {  
    sayHello();  
    delay(1000);  
}
```

```
void sayHello () {  
    Serial.println("Hello, world ");  
}
```


bonus lab

Goal: 用可變電阻與 function 控制七段顯示器