

用 Arduino 製作一個感覺在呼吸的燈，運用 PWM(模擬類比輸出)控制 LED

- 實驗材料

1. 第一題檔案：

Name	Quantity	Component
U1	1	Arduino Uno R3
R1	1	220 $\Omega$ Resistor
D1	1	Red LED

2. 第二題檔案：

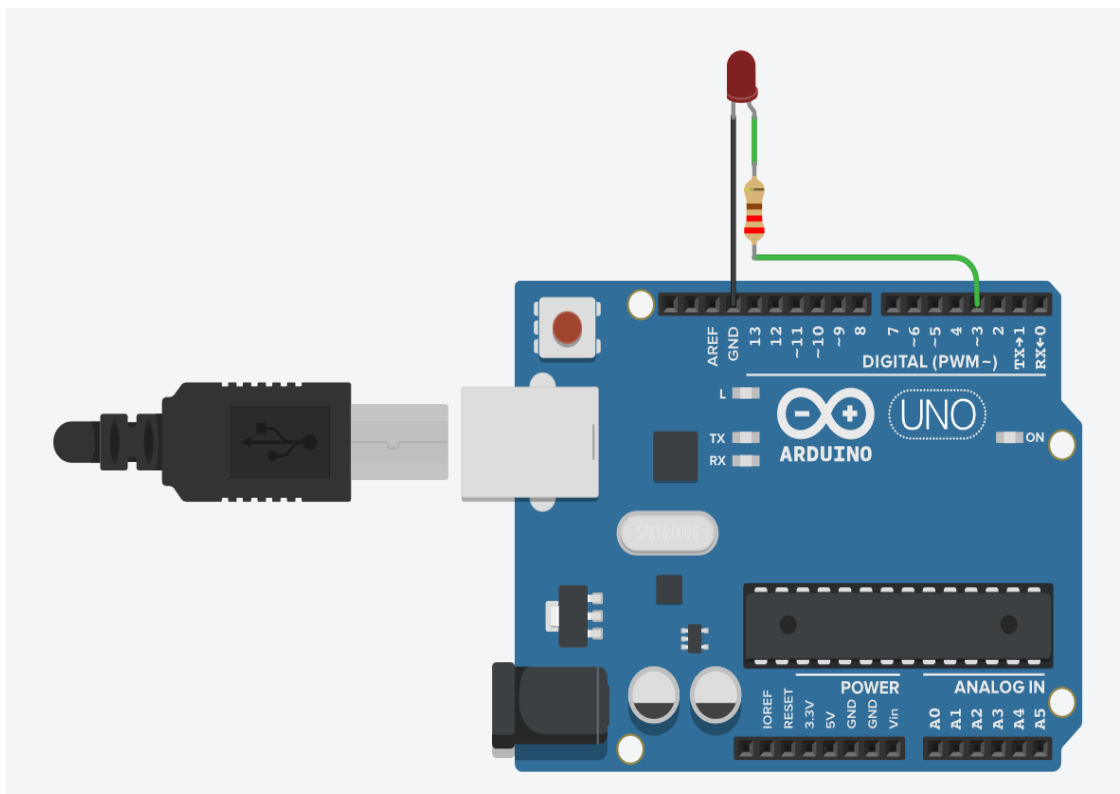
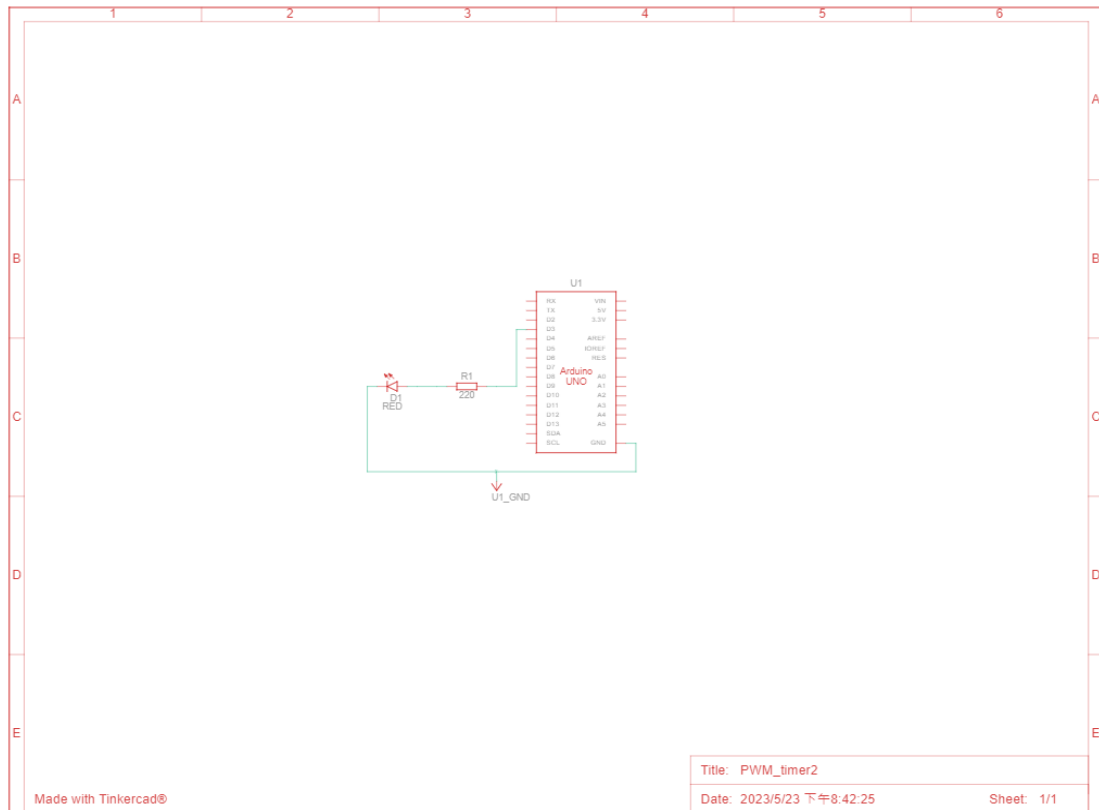
Name	Quantity	Component
U1	1	Arduino Uno R3
R1	1	220 $\Omega$ Resistor
D1	1	Red LED

3. 第三題檔案：

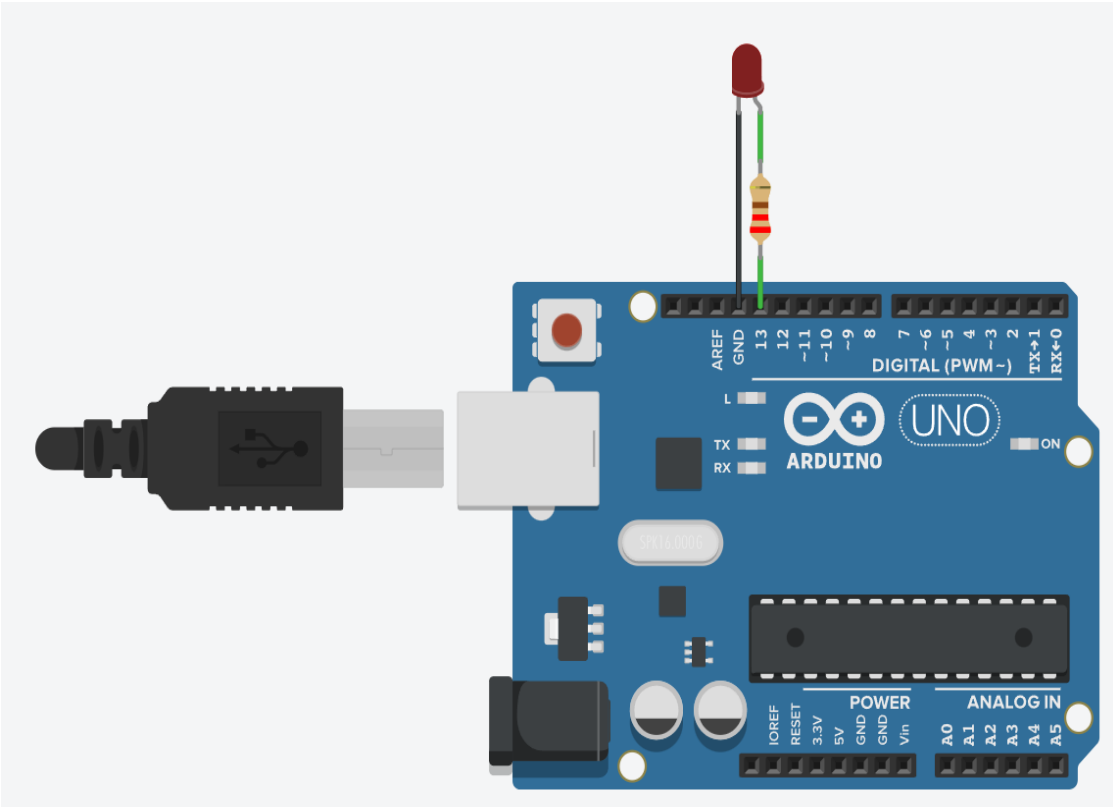
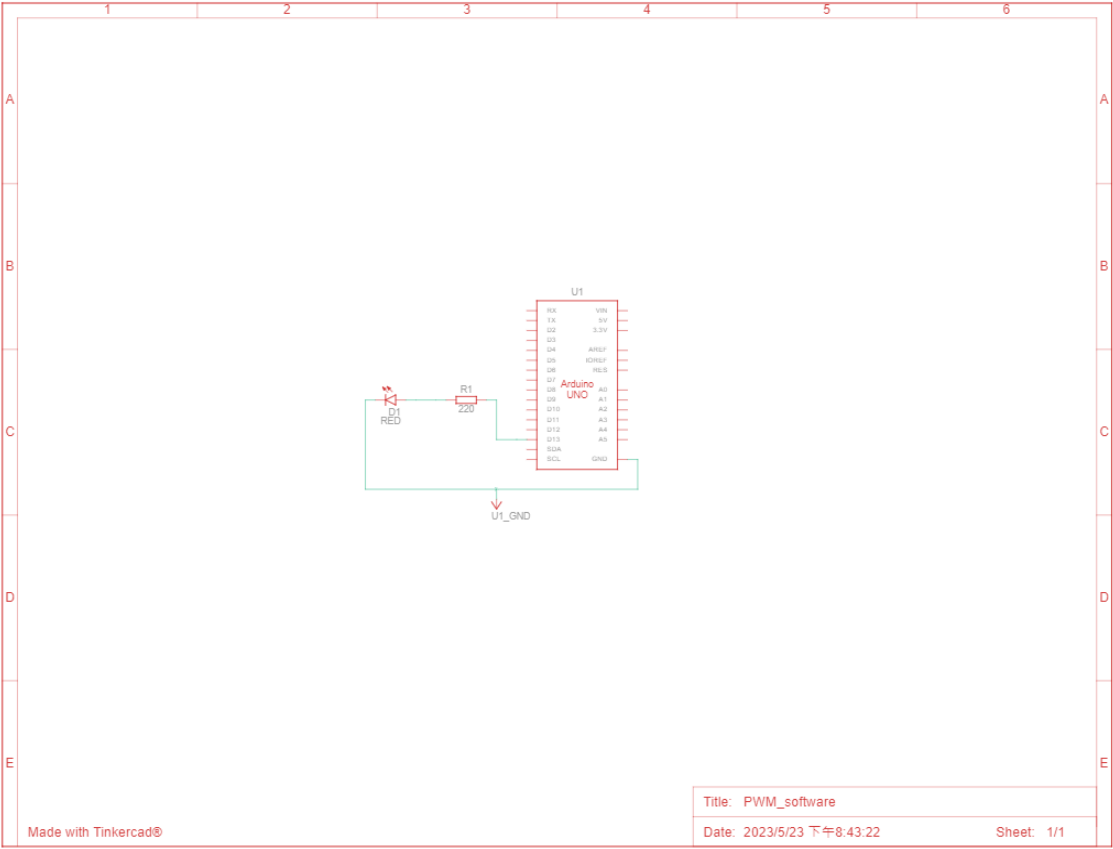
Name	Quantity	Component
U1	1	Arduino Uno R3
D1	1	Red LED
R1	1	1 k $\Omega$ Resistor

- 電路圖

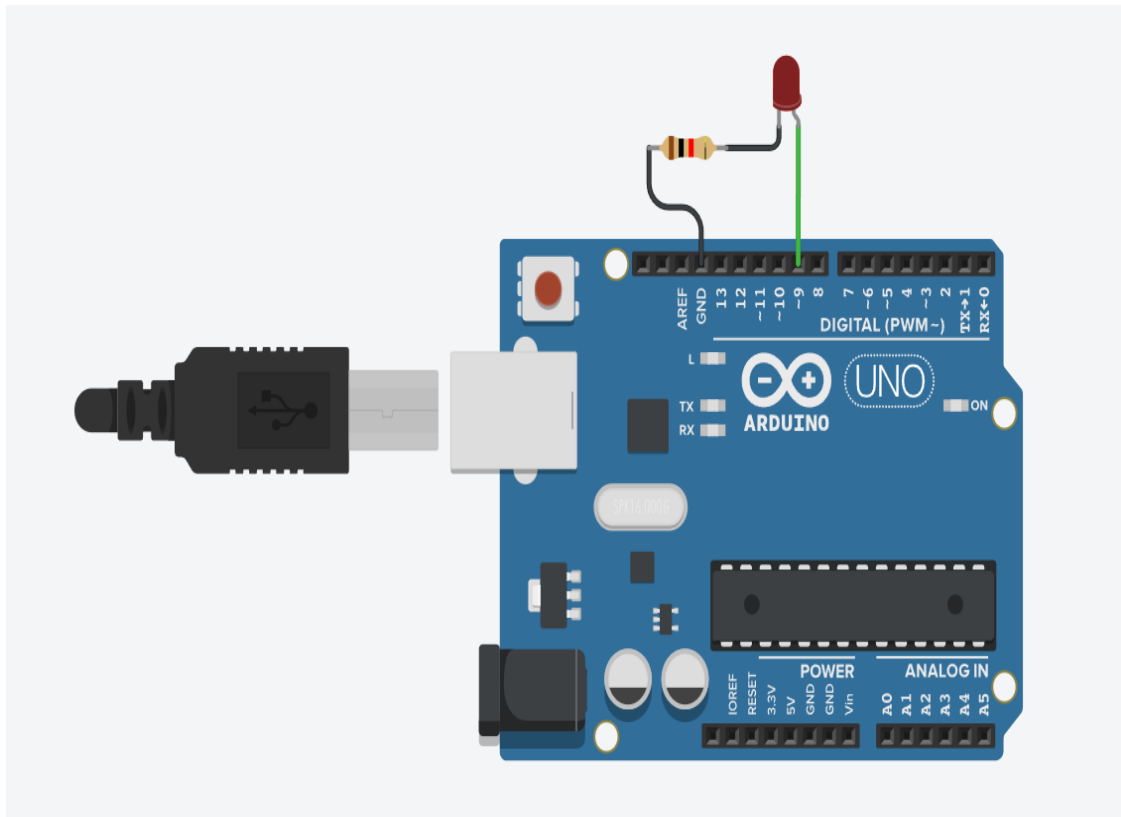
1. 第一題檔案



2. 第二題檔案



### 3. 第三題檔案



- 程式碼

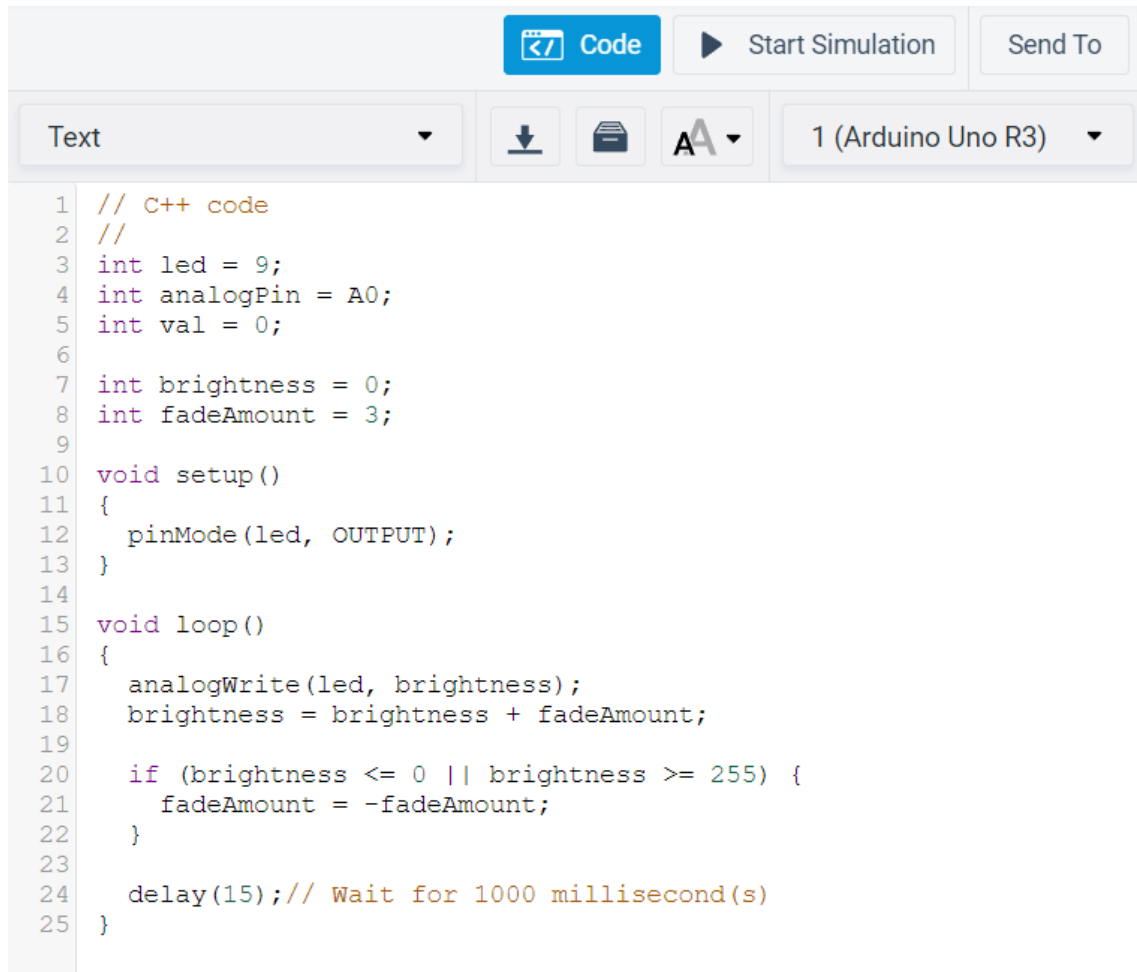
1. 第一題檔案：

```
Text [Download] [Save] [Font Size] 1 (Arduino Uno R3)
1  /** 範例9.5.1(PWM範例-1) ***/
2  #include <wiring_private.h>      //暫存器位元名稱的定義
3
4  /*****
5   * setup & loop
6   *****/
7  void setup() {}
8  void loop() {
9      pin3PWM(31,5);      //freq=31Hz, duty cycle=5%
10     delay(3000);
11     pin3PWM(2000,50);    //freq=2000Hz, duty cycle=90%
12     delay(3000);
13 }
14
15 /*****
16 * pin3PWM
17 *****/
18 int pin3PWM(int freq, int dc)
19 {
20     float TOP;
21     int i, pre[6]={8,32,64,128,256,1024};
22
23     pinMode(3, OUTPUT);
24     if(dc==0) {
25         digitalWrite(3,LOW); //若duty cycle=0%直接改用digitalWrite(3,LOW)
26         return 1;
27     }
28     if(dc==100) {
29         digitalWrite(3,HIGH); //若duty cycle=100%直接改用digitalWrite(3,HIGH)
30         return 1;
31     }
32     //根據目標頻率，計算適用的除頻倍率，並且設定OCR2A
33     for(i=0;i<6;i++) {
34         TOP=16000000/pre[i]/freq/2;      //頻率公式
35         if(TOP<255 && TOP>0) { OCR2A=TOP; break; } //合法的TOP值
36     }
37
38     //設定正確的 CS22-CS20
39     switch(i) {
40         case 0: //prescaler=8, CS=0102
41             sbi(TCCR2B,CS21); break;
42         case 1: //prescaler=32, CS=0112
43             sbi(TCCR2B,CS21); sbi(TCCR2B,CS20); break;
44         case 2: //prescaler=64, CS=1002
45             sbi(TCCR2B,CS22); break;
46         case 3: //prescaler=128, CS=1012
47             sbi(TCCR2B,CS22); sbi(TCCR2B,CS20); break;
48         case 4: //prescaler=256, CS=1102
49             sbi(TCCR2B,CS22); sbi(TCCR2B,CS21); break;
50         case 5: //prescaler=1024, CS=1112
51             sbi(TCCR2B,CS22); sbi(TCCR2B,CS21); sbi(TCCR2B,CS20); break;
52         default: //無可用的除頻倍率，設定失敗，傳回0
53             return 0;
54     }
55     //設定 Phase correct PWM (mode 5) 模式，WGM2[2:0]=1012
56     sbi(TCCR2B,WGM22); sbi(TCCR2A,WGM20);
57     OCR2B=OCR2A*dc/100; //根據duty cycle，設定正確的OCR2B
58     sbi(TCCR2A,COM2B1); //設定OC2B的比較輸出模式=102
59     return 1;          //設定成功，傳回1
60 }
```

## 2. 第二題檔案：

```
Code Start Simulation Send To
Text 1 (Arduino Uno R3)
1  /** 範例9.5.2 (PWM範例-2)  **/
2  #include <wiring_private.h>
3  volatile int ISR_cnt, Soft_pin, Soft_dc;
4
5  /*****
6   * setup & loop
7   *****/
8  void setup() {
9      softPWM_init();
10 }
11
12 void loop() {
13     int i;
14     for(i=0; i<=45; i+=2)
15         { softPWM(13,i); delay(200); }
16     for(i=45; i>=0; i-=2)
17         { softPWM(13,i); delay(200); }
18 }
19
20 /*****
21 * softPWM_init
22 *****/
23 void softPWM_init()
24 {
25     //---step1:初始暫存器
26     TCCR2A=0; TCCR2B=0; TCNT2=0;
27     //---step2:設定CTC模式
28     sbi(TCCR2A,WGM21);
29     //---step3:設定預先除頻倍率prescaler與TOP值
30     sbi(TCCR2B,CS21); sbi(TCCR2B,CS20); //prescaler=32
31     OCR2A=64; //設定TOP=64
32     //---step4:致能對應的中斷
33     sbi(TIMSK2, OCIE2A);
34     //---其它變數
35     ISR_cnt=99; Soft_pin=-1; Soft_dc=0;
36 }
37
38 /*****
39 * softPWM
40 *****/
41 void softPWM(int pin, int dc)
42 {
43     pinMode(pin,OUTPUT);
44     Soft_pin=pin; Soft_dc=dc; //設定pin腳與duty cycle
45     ISR_cnt=0; //ISR_cnt計數值重置為0
46 }
47
48 /*****
49 * Timer2的COMPA中斷服務程式
50 *****/
51 ISR(TIMER2_COMPA_vect)
52 {
53     ISR_cnt=++ISR_cnt%100;
54     //前dc個中斷輸出HIGH，後(100-dc)個中斷輸出LOW
55     if(ISR_cnt==0) digitalWrite(Soft_pin,HIGH);
56     if(ISR_cnt==Soft_dc) digitalWrite(Soft_pin,LOW);
57 }
58
```

### 3. 第三題檔案：



```
1 // C++ code
2 //
3 int led = 9;
4 int analogPin = A0;
5 int val = 0;
6
7 int brightness = 0;
8 int fadeAmount = 3;
9
10 void setup()
11 {
12   pinMode(led, OUTPUT);
13 }
14
15 void loop()
16 {
17   analogWrite(led, brightness);
18   brightness = brightness + fadeAmount;
19
20   if (brightness <= 0 || brightness >= 255) {
21     fadeAmount = -fadeAmount;
22   }
23
24   delay(15); // Wait for 1000 millisecond(s)
25 }
```

#### ● 反思

在我進行 **Arduino** 的製作專案時，我選擇了一個有趣且具有美感的專案：製作一個感覺在呼吸的燈。這個專案利用 **Arduino** 的 **PWM**（模擬類比輸出）功能來控制 **LED** 的亮度，讓燈光仿佛在呼吸一般漸強漸弱，給人一種柔和且舒適的感覺。

我首先瞭解到 **Arduino** 是一個開源的單片機平台，擁有廣泛的應用和豐富的開發資源。對於一個初學者來說，**Arduino** 提供了簡單易用的開發環境和豐富的函式庫，讓我能夠輕鬆上手並進行各種有趣的實驗。

在這個專案中，我首先準備了一個 **Arduino UNO** 開發板、一個 **LED** 和幾條連接線。我將 **LED** 連接到開發板的數位腳位上，利用 **PWM** 功能來控制 **LED** 的亮度。**PWM** 是一種模擬類比輸出的技術，通過改變脈衝寬度的長短來控制電源的平均電壓，從而改變 **LED** 的亮度。

接下來，我在 **Arduino** 的開發環境中編寫了程式碼。我使用了 `analogWrite()` 函式來設置 **PWM** 輸出，並通過改變脈衝寬度的數值來調整 **LED** 的亮度。為了實現呼吸的效果，我使用了一個迴圈，逐漸增加和減少 **PWM** 輸出的數值，從而使 **LED** 的亮度逐漸變化。

經過一番調試和測試，我成功地將 **Arduino** 連接到 **LED**，並製作出了一個感覺在呼吸的燈。當我啟動程式後，**LED** 的亮度會逐漸增加，然後再逐漸減弱，就像是燈光在呼吸一樣。這個效果非常迷人，讓人感到放鬆和舒適。

這個專案讓我深入瞭解了 **Arduino** 的 **PWM** 功能和模擬類比輸出的原理。通過這個專案，我不僅學會了如何使用 **PWM** 控制 **LED** 的亮度，還體驗到了 **Arduino** 這個平台的強大和靈活性。

在完成這個專案的過程中，我遇到了一些困難和挑戰，但通過閱讀文檔、查找資源和實驗不斷嘗試，我最終成功地解決了這些問題。這個過程讓我學會了如何克服困難，並培養了我在 **Arduino** 開發中的問題解決能力。

總的來說，這個 **Arduino** 的製作專案是一次寶貴的經驗。我學到了很多關於 **PWM** 和模擬類比輸出的知識，同時也提升了我的電子製作和程式編寫能力。我對於 **Arduino** 的應用和開發充滿了興趣，並期待在未來的專案中繼續深入探索和學習。