





Arduino程式設計

二十四小時制 七段顯示器



電子二忠 3號 王昱智

目錄

- 介紹
- 程式架構
- 成果
- 心得感想





介紹

Arduino

- **Arduino**是一個開源的電子開發平台，讓我們能夠實現創意和想法，學習程式設計和電子電路控制，並建立各種有趣的互動裝置和原型。它簡單易用，提供豐富的函式庫和範例，讓我們能夠快速入門並進行自己的電子創作。





程式架構

Set up():

- Setup() 函式是一個特殊的函式，在程式開始執行時被自動呼叫，用於初始化設定。以下是setup() 函式的功能：
 1. 設定數字顯示器的腳位模式為輸出：使用 pinMode() 函式將 sega 和 seg4 陣列中的腳位設定為輸出模式。這些腳位用於控制數字顯示器的段和位元。
 2. 設定變數的初始值：將變數 d0、d1、d2、d3 和 start 初始化為特定的數值。這些變數用於控制要顯示的數字。





程式架構

Loop():

- Loop() 函式則是 Arduino 程式的主要執行函式，它會不斷地重複執行。以下是 Loop() 函式的功能：
 1. 呼叫 scan() 函式：scan() 函式用於控制數字顯示器的腳位，根據變數的值來顯示對應的數字。
 2. 更新變數的值：透過變數 d0、d1、d2、d3 和 aaa 的增加，來控制要顯示的數字。在每次迴圈中，變數 aaa 會增加，當其值超過 10 時，會重置為 0，並更新其他變數的值以達到數字的變化。

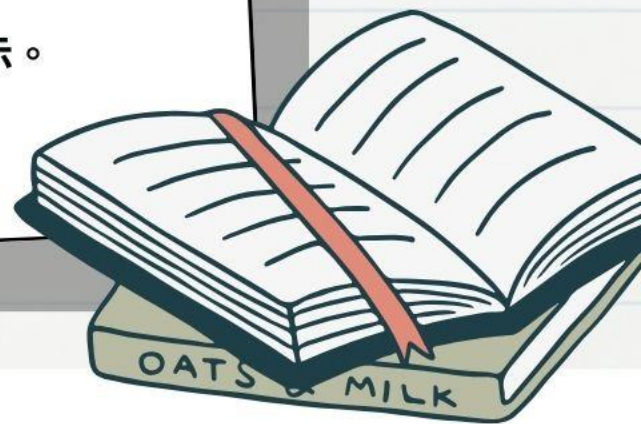




程式架構

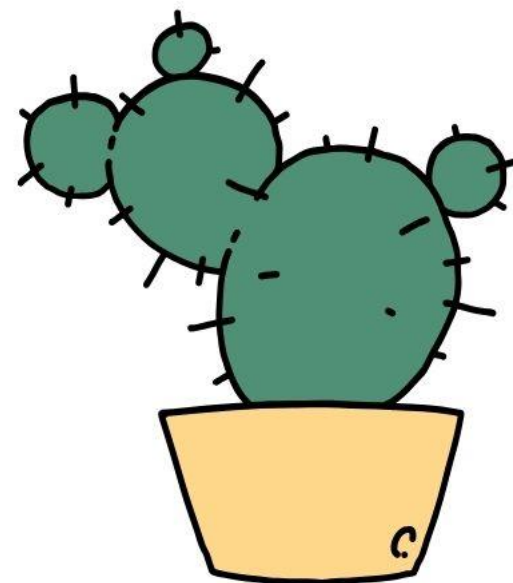
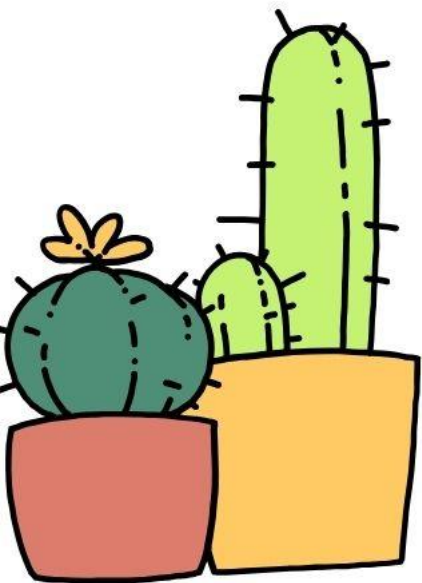
二十四小時制:

1. 數字變化控制：透過 d0、d1、d2 和 d3 四個變數的遞增和進位操作，實現數字的變化和連續顯示。
 2. 進位機制：使用 if 條件語句檢查各個位數的值是否超過指定範圍（9、5、3、2），並進行進位操作，將進位的數字傳遞到高一位的變數中。
 3. 變數重置：當某一位數達到指定範圍時，將其重置為 0，以實現進位操作。
 4. 循環執行：這段程式碼通常位於 loop() 函式中，表示它會不斷地重複執行，以實現連續的數字變化。
 5. 數字範圍限制：透過 if 條件語句限制各個位數的數字範圍，確保它們在指定範圍內循環變化，例如 d0 的範圍是 0 到 9，d1 的範圍是 0 到 5，以此類推。
- 這些要素共同作用，使得數字能夠在指定範圍內循環變化，並實現連續的數字顯示。



實習成果

```
1  int sega[]={0,1,2,3,12,5,6,7};
2  int seg4[]={8,9,10,11};
3  int i,j,aaa,start;
4  int d0,d1,d2,d3;
5  byte num[]={B11000000,B11111001,B10100100,B10110000,
6  B10011001,B10010010,B10000010,B11111000,B10000000,B10010000};
7  void setup(){
8      for(i=0;i<8;i++)
9      {
10         pinMode(sega[i],OUTPUT);
11         pinMode(seg4[i],OUTPUT);
12     }
13     d0=8,d1=5,d2=3,d3=2,start=0;
14 }
```

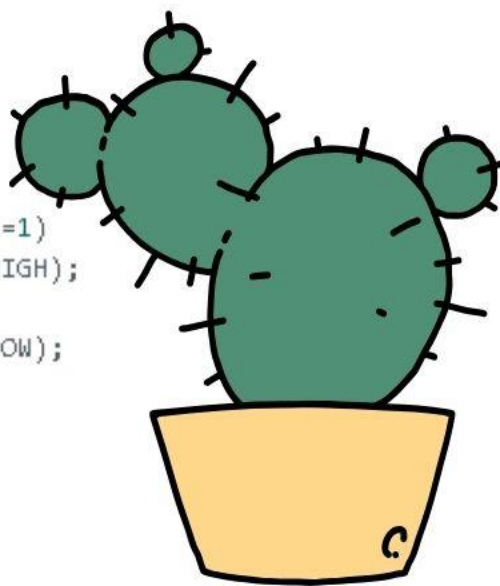


實習成果

```
15 void loop()  
16 {  
17   scan();  
18   aaa++;  
19  
20   while(aaa>10)  
21   {  
22     aaa=0;  
23  
24  
25  
26     d0++;  
27     if(d0>9)  
28     {  
29       d1++;  
30       d0=0;  
31     }
```

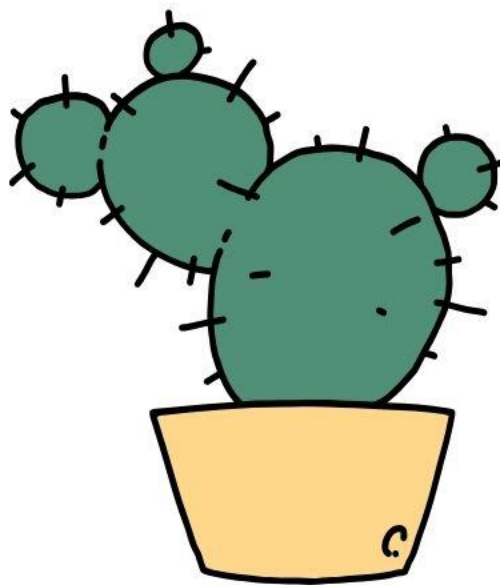
```
32   if(d1>5)  
33   {  
34     d2++;  
35     d1=0;  
36   }  
37   if(d2>3)  
38   {  
39     d3++;  
40     d2=0;  
41   }  
42   if(d3>2)  
43   {  
44     d3=0;  
45   }  
46  
47  
48 }
```

```
52 void scan()  
53 {  
54   digitalWrite(8,1);  
55   for(j=0;j<8;j++)  
56   {  
57     digitalWrite(8,1);  
58     for(j=0;j<8;j++){  
59       if(bitRead(num[d0],j)==1)  
60         digitalWrite(sega[j],HIGH);  
61       else  
62         digitalWrite(sega[j],LOW);  
63     }  
64     delay(5);  
65     digitalWrite(8,0);  
66  
67     digitalWrite(9,1);  
68     for(j=0;j<8;j++){  
69       if(bitRead(num[d1],j)==1)  
70         digitalWrite(sega[j],HIGH);  
71       else  
72         digitalWrite(sega[j],LOW);  
73     }  
74   }
```



實習成果

```
52 void scan()  
53 {  
54     digitalWrite(8,1);  
55     for(j=0;j<8;j++)  
56     {  
57         digitalWrite(8,1);  
58         for(j=0;j<8;j++){  
59             if(bitRead(num[d0],j)==1)  
60                 digitalWrite(sega[j],HIGH);  
61             else  
62                 digitalWrite(sega[j],LOW);  
63         }  
64         delay(5);  
65         digitalWrite(8,0);  
66  
67         digitalWrite(9,1);  
68         for(j=0;j<8;j++){  
69             if(bitRead(num[d1],j)==1)  
70                 digitalWrite(sega[j],HIGH);  
71             else  
72                 digitalWrite(sega[j],LOW);  
73         }  
74     }
```





心得感想

這段程式碼讓我對數字顯示和控制有了更深入的了解。透過控制不同位數的變化和進位條件，我們可以實現數字的循環顯示效果。這簡單的程式碼展示了程式設計的基本概念，如變數、迴圈和條件語句的應用。這段程式碼的編寫過程讓我更加細心和耐心，並學會了如何解決問題和優化程式。儘管這只是一個簡單的例子，但它激發了我對程式設計的興趣。我期待能夠進一步學習和探索更複雜的程式設計概念和技巧，並應用於實際的專案中。這段程式碼讓我明白了程式設計的潛力和靈活性，並且讓我對於未來的學習和發展充滿了期待。



Thank's For Watching

