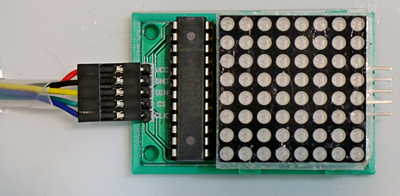


積木用途：以10進制數字設定MAX7219 8\*8 LED矩陣模組中指定行的8顆LED亮或滅的狀態。

說明：

1. 「公版教具(4060)」中的8\*8 LED 矩陣模組線路分成兩組，第一組黃、黑、紅線接在積木預設的D12，第二組綠、藍線接在I2C插槽的A4與A5。
2. 「公版教具(5012、5016A/B)」中的8\*8LED矩陣模組，將CS接腳改為D10，CLK改為D11。



1. 此積木沒有設定8\*8 LED 矩陣模組連接的腳位，使用之前必須先執行底下這個積木進行初始化。

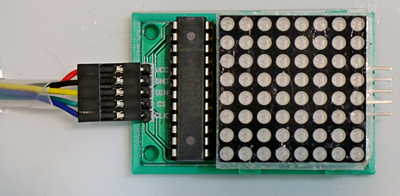


使用「公版教具(5012、5016A/B)」時需更改CS與CLK腳位。



1. 8\*8 LED 矩陣模組中LED的排列由左至右共8行，每行8顆燈。

行 1 2 3 4 5 6 7 8



1. 模組的每一顆LED由1個位元控制亮(1)或滅(0)，如果把第1行LED狀態設定成11111111，代表該行8顆燈全亮，10101111代表該行除第2、4顆燈熄滅以外其餘全亮。不過這個積木必須使用十進制數字來設定，所以必須將位元狀態轉換為十進制數字才可使用，轉換方法請參考底下表格。

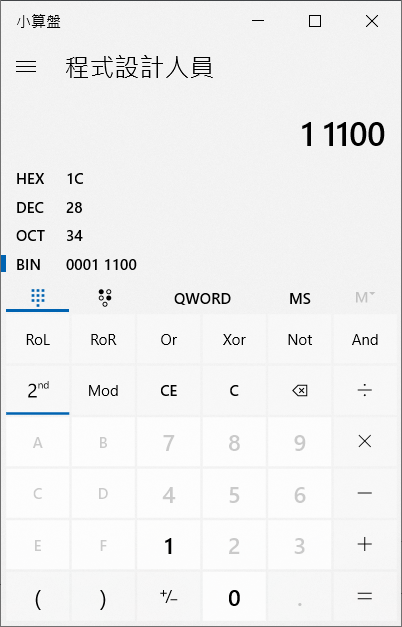
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 燈號 | 狀態 | 十進制數字 |  | 燈號 | 狀態 | 十進制數字 |  | 燈號 | 狀態 | 十進制數字 |
|  | 1 | 27=128 |  | 1 | 27=128 |  | 0 |  |
| 1 | 26=64 | 1 | 26=64 | 0 |  |
| 1 | 25=32 | 1 | 25=32 | 0 |  |
| 1 | 24=16 | 1 | 24=16 | 0 |  |
| 1 | 23=8 | 0 |  | 1 | 23=8 |
| 1 | 22=4 | 1 | 22=4 | 1 | 22=4 |
| 1 | 21=2 | 0 |  | 1 | 21=2 |
| 1 | 20=1 | 1 | 20=1 | 1 | 20=1 |
| 合計 | | 255 | 合計 | | 245 | 合計 | | 15 |

**程式範例：簡易動畫**

說明：在8\*8 LED 矩陣模組設計一個簡單的圖樣，利用此積木製作讓圖案由左向右平移的動畫。

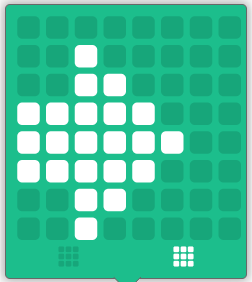
步驟：

1. 設計圖案，並算出圖案每一行的十進制數值，下圖中的箭頭圖案共6行，由左至右10進制數值分別是：28、28、127、62、28、8。
2. 計算有困難的話，可以叫出小算盤，切換到程式設計人員模式。
3. 點擊BIN切換到2進位輸入模式。
4. 由上而下依序輸入該行每個點的狀態，例如第一行要輸入00011100，不過開頭的3個0在輸入時不會顯示，可以直接輸入11100即可。
5. DEC數字就是要記下來的10進制數值。
6. 點擊CE清除目前的數字，繼續輸入下一行狀態。



1

2



合計28 28 127 62 28 8

0 0 1 0 0 0

0 0 0 0 0 0

0 0 1 1 0 0

1 1 1 1 1 0

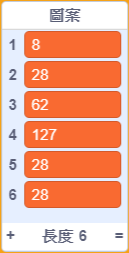
1 1 1 1 1 1

1 1 1 1 1 0

0 0 1 1 0 0

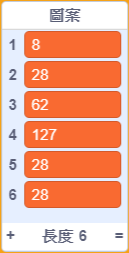
0 0 1 0 0 0

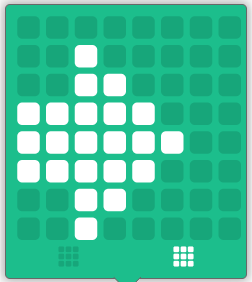
1. 建立一個清單來存放圖案的十進制數值，清單的第一項放置圖案最右邊行的數值，也就是箭頭的頂尖，第六項放置圖案最左邊行的數值。



1. 建立3個變數：指標行、繪圖行、數值項。
2. 指標行用來指示圖案最右邊，也就是箭頭頂尖的位置。
3. 繪圖行用來指示圖案每一行的位置，由指標行的位置開始，往左6行就是這個圖案的繪圖範圍。
4. 數值項用來指示清單中的項目位置，用來提取每一行繪圖時的10進制數值，預設值為1，範圍1-6。

指標行





數值項

繪圖行範圍

1. 先把指標行設為6，繪圖行與指標行相同，數值項設為1，將數值「畫」在模組上，就可看到箭頭的頂尖，接著將繪圖行減1，數值項加1，「畫」出圖案第二行，如此重複6次，就可以畫出完整圖形，繪圖的步驟與變數的變化如下圖所示。

指標行=6

繪圖行=6

數值項=1

指標行=6

繪圖行=5

數值項=2

指標行=6

繪圖行=4

數值項=3

指標行=6

繪圖行=3

數值項=4

指標行=6

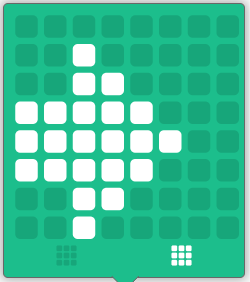
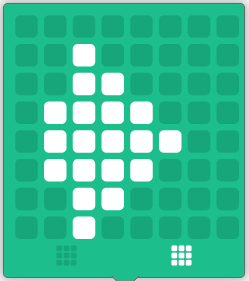
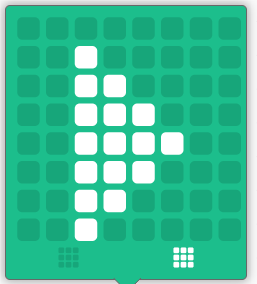
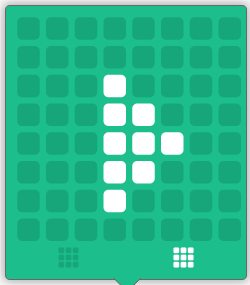
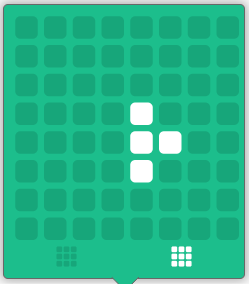
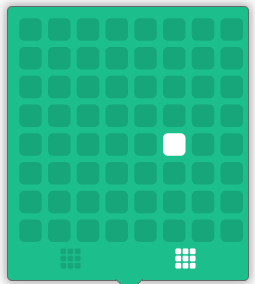
繪圖行=2

數值項=5

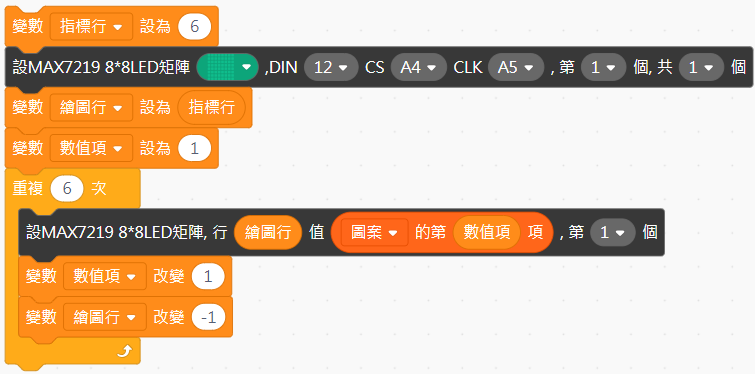
指標行=6

繪圖行=1

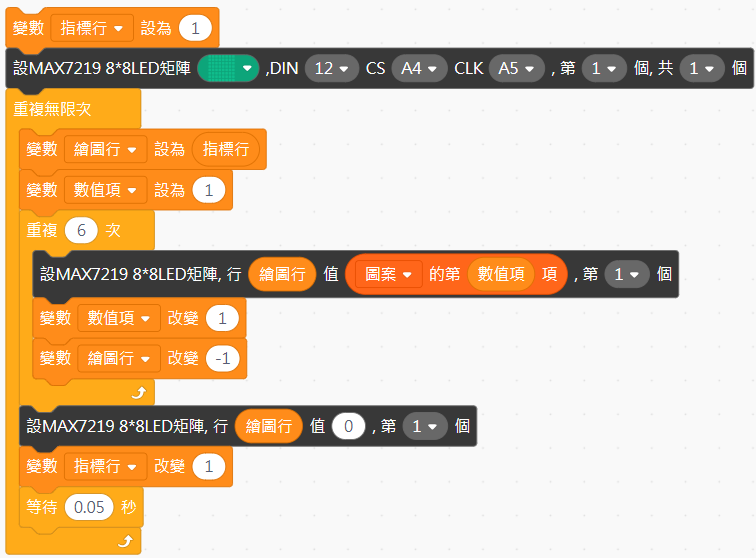
數值項=6



底下程式可以完成繪圖的工作。



1. 接下來就開始做動畫了，動畫效果是讓圖案由模組左邊平移到右邊，所以變數指標行的預設值要改為1，每畫完一個完整的圖案時就將指標行加1，圖案就會往右偏移，不過要記得將上一次所畫圖案最左邊行清除，否則會留下一條長長的尾巴。

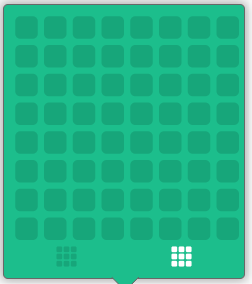
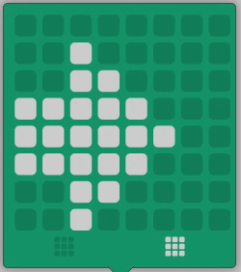


清除上一次繪圖最左邊的部分，以免留下尾巴

1. 可是這麼一來，圖案往右移就一去不回頭了，怎麼讓它重新回到模組左邊呢？這裡可以運用餘數運算來幫我們做出循環效果。首先，先決定圖案跑到何處要回頭？例如當讓箭頭完全移出模組時就回頭，這時的指標行等於14(模組8行+圖案6行)，也就是指標行的變動範圍是1~14，那麼任何數除以14的餘數範圍為0~13，將餘數加1的範圍就是1~14，所以只要在畫完一個圖案後，將指標行設定為(指標行除以14的餘數)再加1就大功告成了。

指標行=14

1 2 3 4 5 6 7 8 9 10 11 12 13 14



在模組的右側延伸一個假想的8\*8矩陣

