



评分方式：包含應用實作，總成績預計至少105分

(1) 每個上課指派之實作作業佔4~5% (預計總計50%)

以兩人一組為原則。**超過繳交期限(一至兩周)後，每星期扣1%。有繳交作業至少給2%。**每次上課會透過實例講解原理，作業是以提供的實例為參考作變化。**出席率偏低，會依據遲交的作業酌量扣分。**

(2) 期中與期末上機測共佔40%

期中可能採組或個人上機測，期末採個人上機測。題目基本上會跟平時作業的程式有關，只要了解平時作業的程式內容，應該作一些程式段整合或修改即可完成。

(3) Arduino感測應用實作佔15~20%

●以兩組整合為一組(預計4人為一組)進行專案實作，以超音波模組、三軸加速度計模組、溫溼度感測模組、三色LED模組、無源蜂鳴器，搭配Arduino UNO R3微控制器，完成老師所展示的功能。

●額外加入微波雷達感應開關模塊，自行整合所有模組，構思一個使用情境，完成功能並展示。

评分原則: 所有感測動作符合老師展示的功能就給8~10%，整合微波雷達感應開關模塊，完成所構思的使用情境將酌量加分。

上課內容規劃:

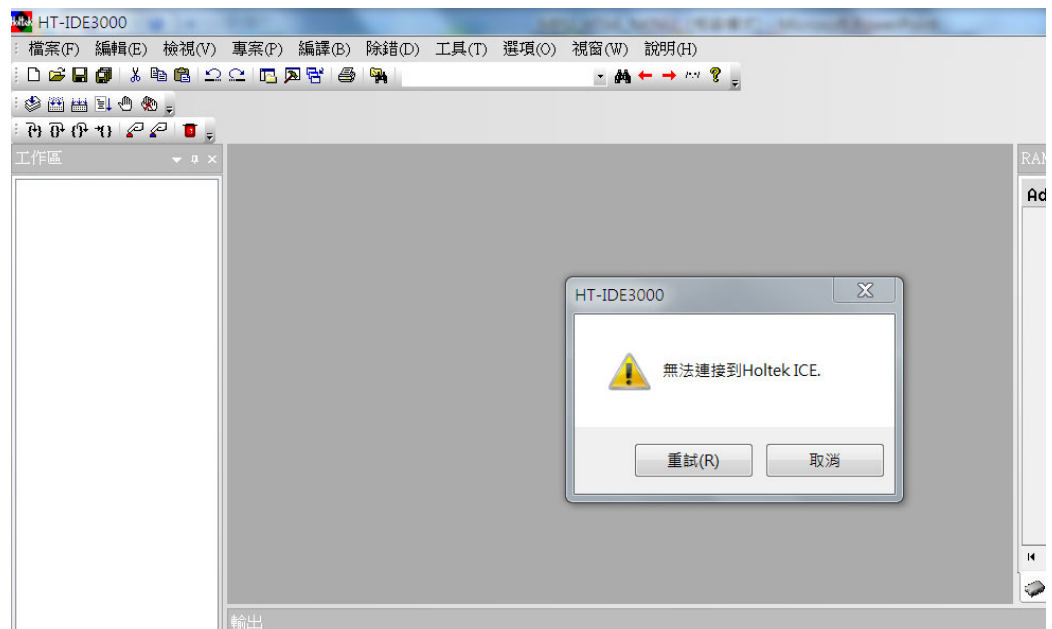
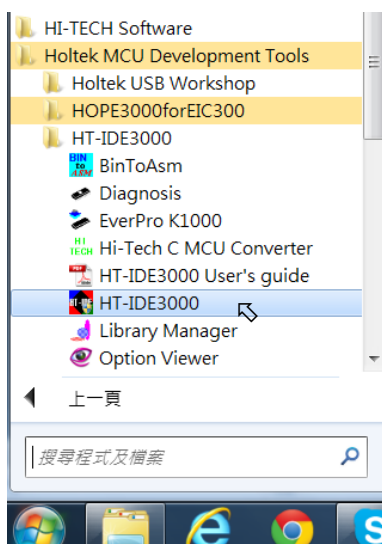
- 以Holtek微控制器講解一般微處理器及周邊整合硬體的相關原理與程式設計考量。
- Arduino由於網路可參考程式很多，而且與感測器相關的控制程式很多有被寫成函式庫或模組化，可不必完全理解感測模組元件動作的協定即可簡易撰寫程式來控制感測模組，大家可透過網路參考來進行期末專案作業。有些模組的通訊傳輸原理，課堂上會透過Holtek實驗板的實作做相關講解，甚至用感測模組透過課程撰寫協定控制程式，以了解Arduino模組化程式的原理。

## Holtek – Using Holtek IDE-3000 & ICE ESK/啟動HT-IDE3000

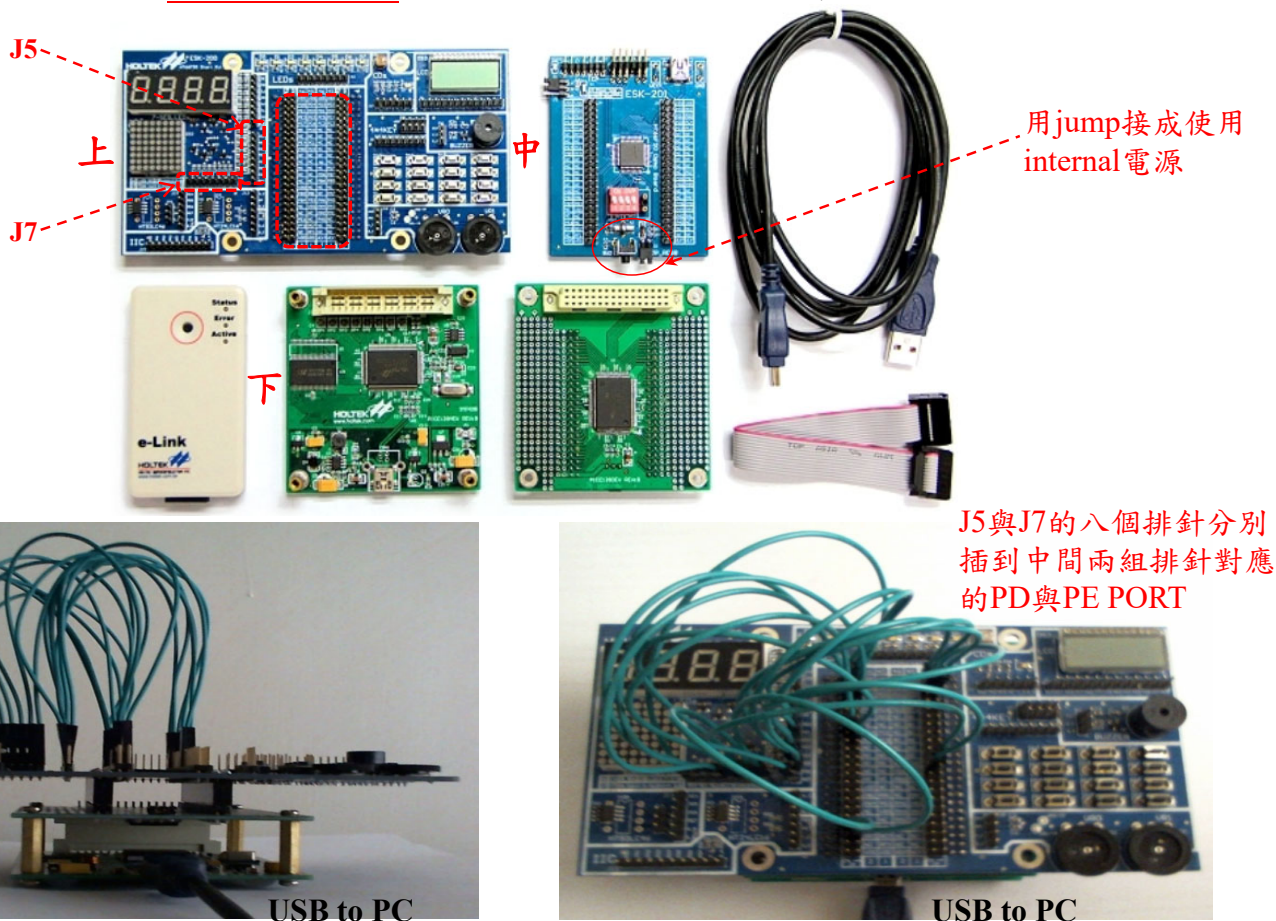


- 點選”開始/所有程式/Holtek MCU Development Tools/HT-IDE3000/HT-IDE3000”，將會出現HT-IDE3000視窗。可能視窗中間會出現”無法連接到Holtek ICE.”的警告訊息，可以**直接按”取消”**，這只是在告知Holtek ICE實驗板尚未透過USB連到電腦，待程式設計完成，再接上ICE板子，進行程式下載執行也是可以的。

Note: ICE ESK套件的接法，參考下一頁

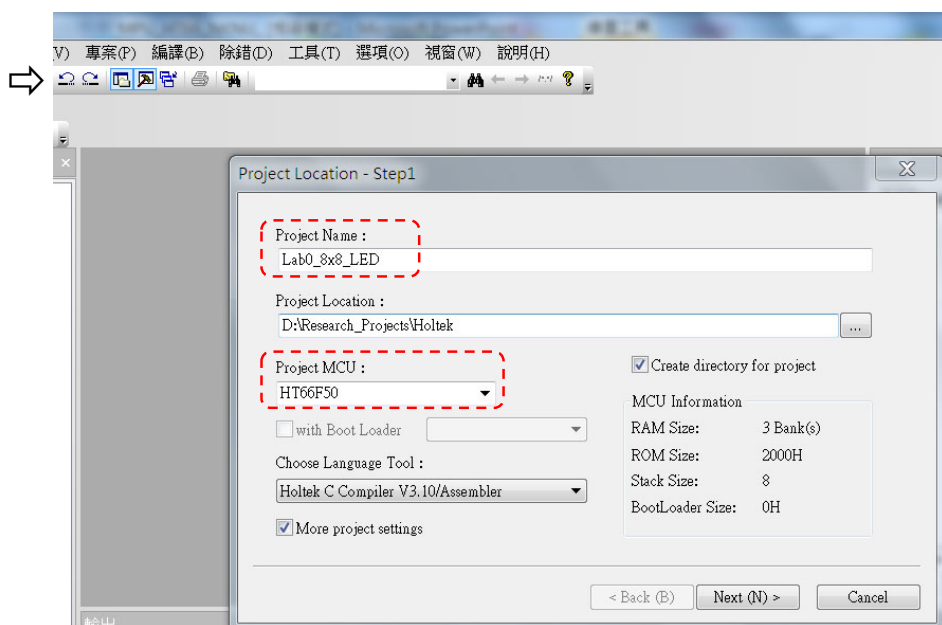
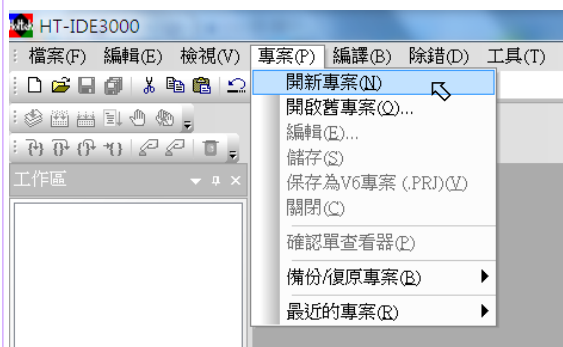


ESK-66F-A01 套件，HT66F50的開發學習板



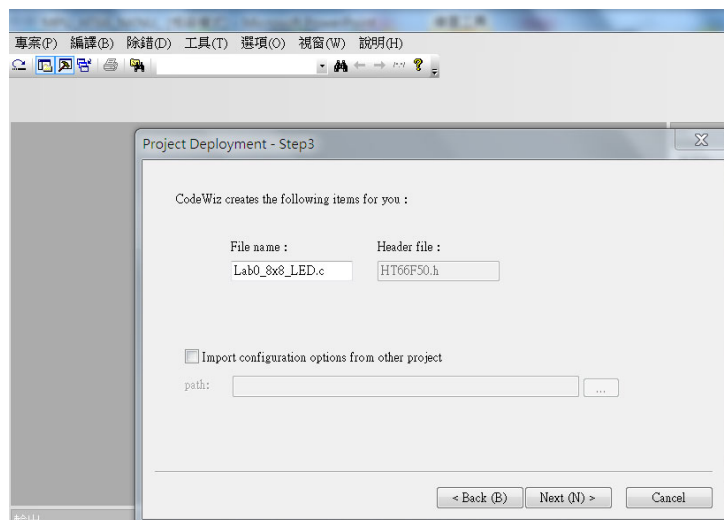
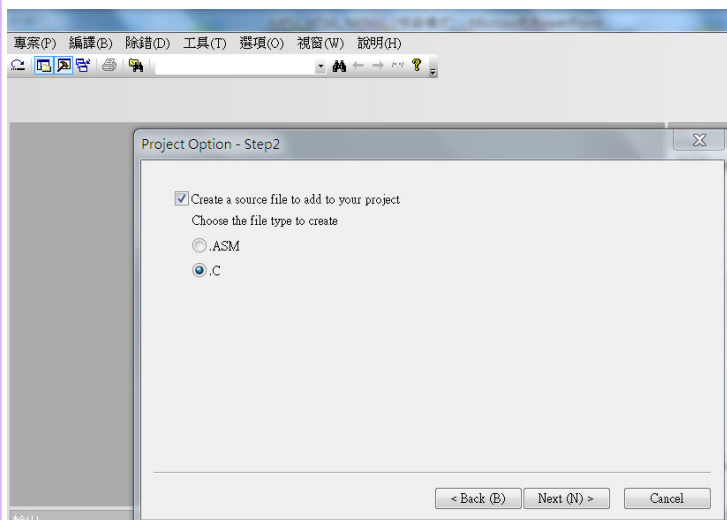
Holtek – Using Holtek IDE-3000 & ICE/建立專案、撰寫與加入程式 (I)

- 點選”專案/開新專案”，將會出現Project Location視窗，於Project Name欄位鍵入專案名稱(例如，Lab0\_8x8\_LED)，並於Project MCU欄位下拉式選單點選”HT66F50”，然後按下”Next”。

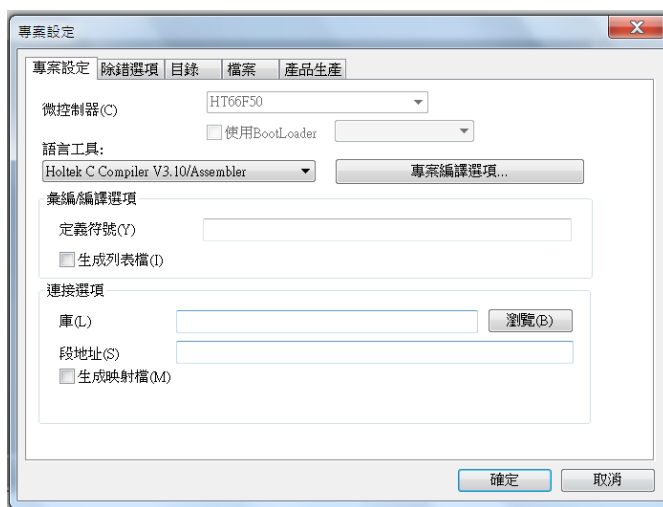
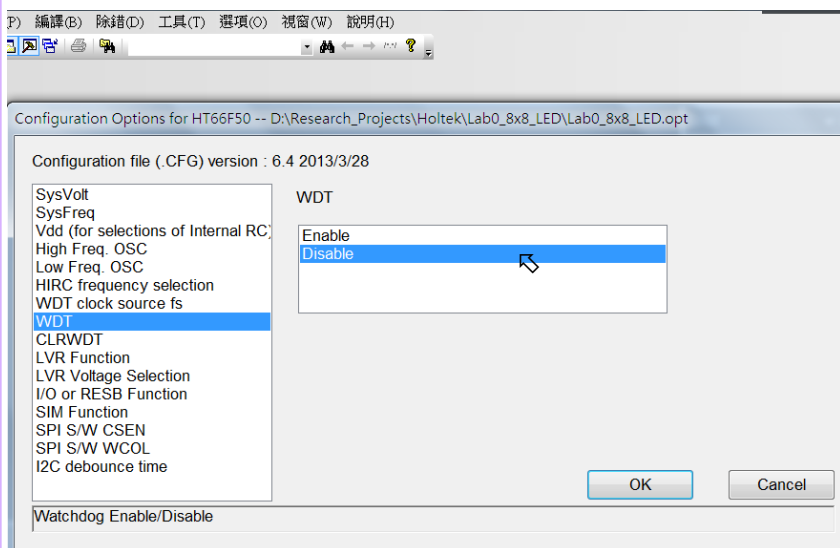




- 於Project Option-Step 2視窗，點選”C”，這是設定程式是以C語言撰寫，然後按下”Next”。於出現的Project Deployment-Step 3視窗，File name欄位會出現與專案名稱一樣的檔名，直接按下”Next”。若出現警告視窗，直接按“OK”。



- 於出現的Configuration Options for HT66F50...視窗，點選”WDT/Disable”，然後按下”OK”。於出現的專案設定視窗，直接按下”確定”。



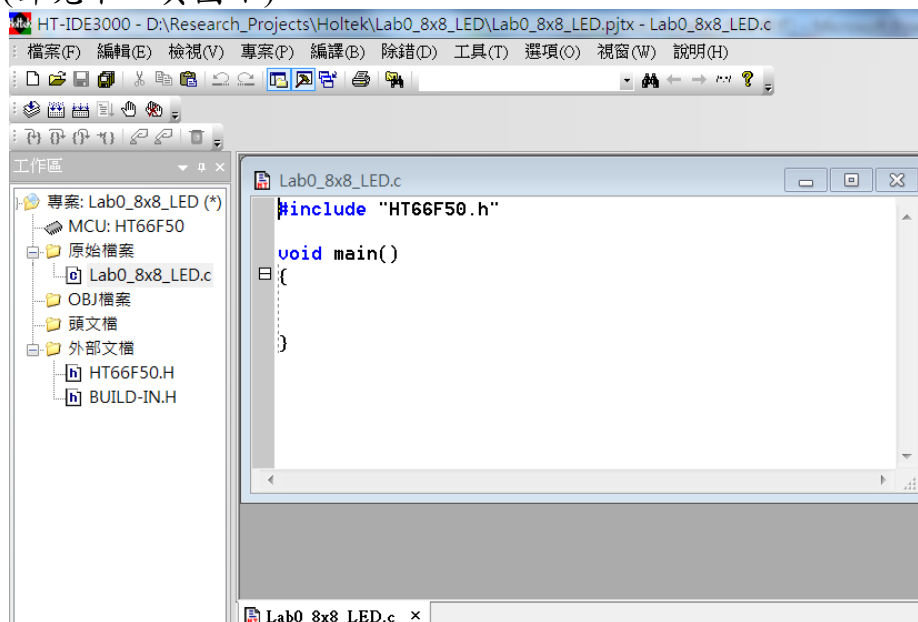




• 接著，就開始編寫程式。

方法一：於出現的Lab0\_8x8\_LED.c視窗，**撰寫程式**。或者

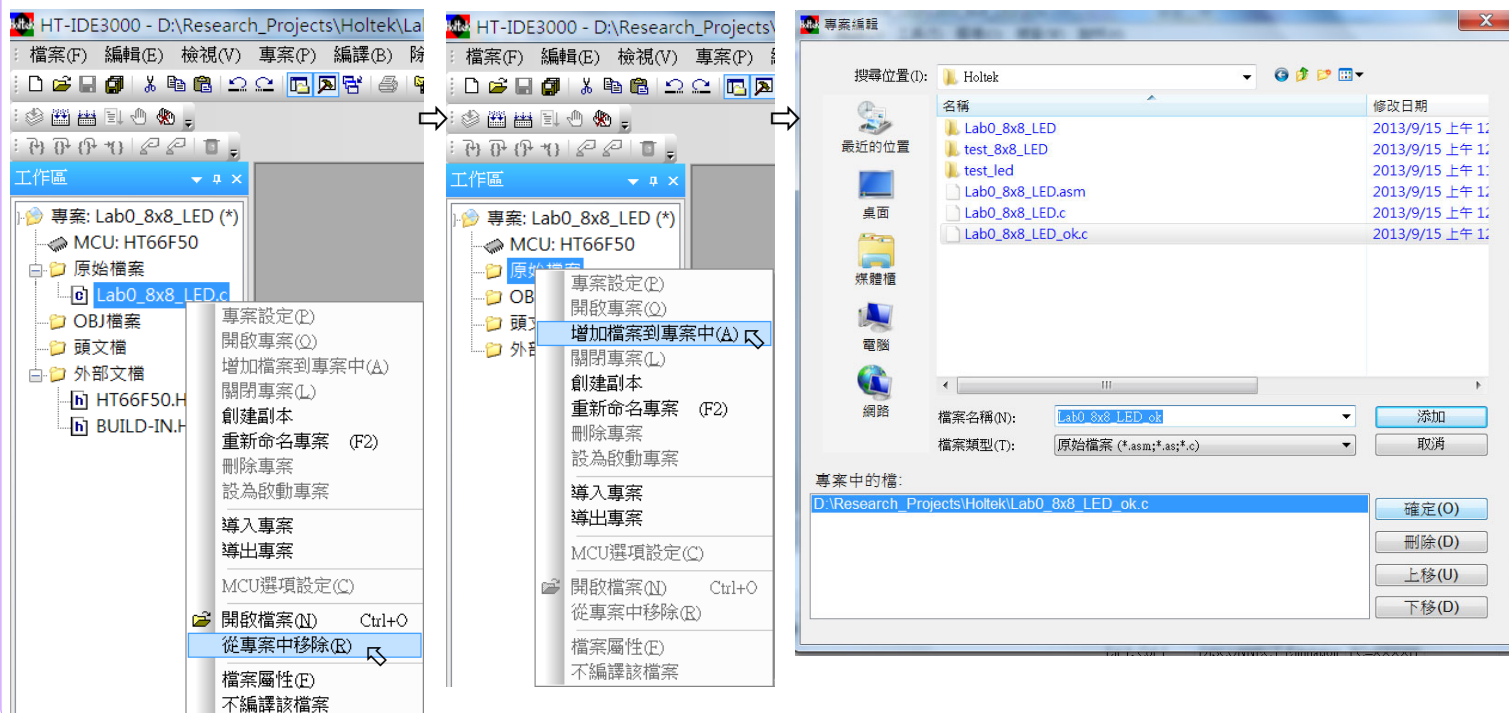
方法二：直接Lab0\_8x8\_LED.c視窗关掉(按視窗右上角的"x")，然後點選**工作區/原始檔案/Lab0\_8x8\_LED.c**並按滑鼠右鍵，於出現的子視窗點選**從專案中移除**，將此原始檔案移除。然後再點選**工作區/原始檔案**並按滑鼠右鍵，於出現的子視窗點選**增加檔案到專案中**。於出現的專案編輯視窗中，搜尋位置欄位下拉式選單，選取欲加入的程式檔，然後按下**添加**，確定檔案名稱與其路徑出現在專案中的檔欄位後，再按下**確定**。(詳見下一頁圖示)



NKNU\_EE\_MM50C\_RLWang



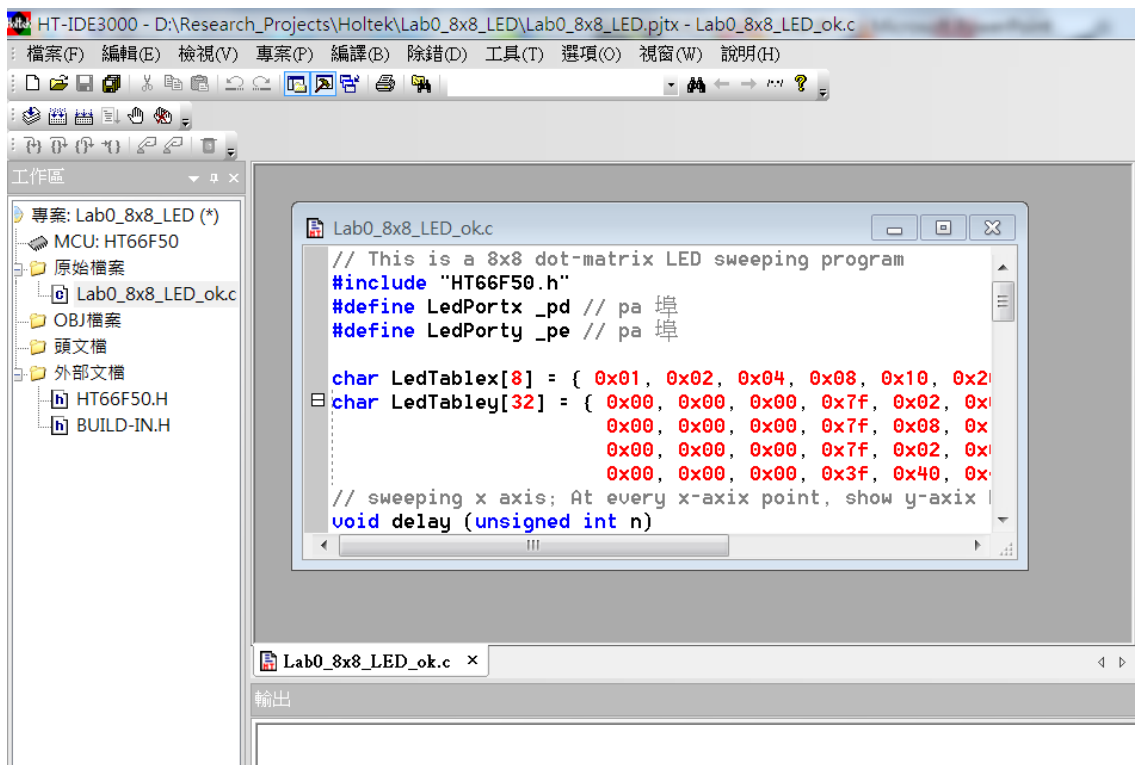
方法二：直接Lab0\_8x8\_LED.c視窗关掉(按視窗右上角的"x")，然後點選**工作區/原始檔案/Lab0\_8x8\_LED.c**並按滑鼠右鍵，於出現的子視窗點選**從專案中移除**，將此原始檔案移除。然後再點選**工作區/原始檔案**並按滑鼠右鍵，於出現的子視窗點選**增加檔案到專案中**。於出現的專案編輯視窗中，搜尋位置欄位下拉式選單，選取欲加入的程式檔，然後按下**添加**，確定檔案名稱與其路徑出現在專案中的檔欄位後，再按下**確定**。可點選**工作區/原始檔案/Lab0\_8x8\_LED\_ok.c**，開啟檔案。



NKNU\_EE\_MM50C\_RLWang



可點選”工作區/原始檔案/Lab0\_8x8\_LED.c”，開啟檔案。



## Holtek – Using Holtek IDE-3000 & ICE/程式內容(1/2)

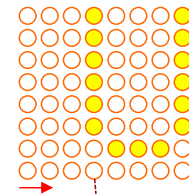
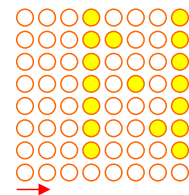
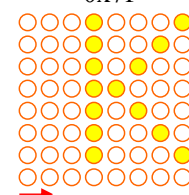


```
// This is a 8x8 dot-matrix LED sweeping program : Display “N”, “K”, “N”, “U” in turn;
// function delay(n) : Adjust the switching rate of LED column by changing the index n value.
#include "HT66F50.h"
#define LedPortx _pd // pd 埠
#define LedPorty _pe // pe 埠
unsigned char LedTablex[8] = { 0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x80 };
unsigned char LedTabley[32] = { 0x00, 0x00, 0x00, 0x7f, 0x02, 0x08, 0x20, 0x7f,
                                0x00, 0x00, 0x00, 0x7f, 0x08, 0x14, 0x22, 0x41,
                                0x00, 0x00, 0x00, 0x7f, 0x02, 0x08, 0x20, 0x7f,
                                0x00, 0x00, 0x00, 0x3f, 0x40, 0x40, 0x40, 0x3f };
// sweeping x axis; At every x-axis point, show y-axis LEDs for a short time
void delay (unsigned int n) //Adjust light intensity of LEDs
{
```

```
    unsigned int idy1, idy2;
    for(idy1 = 0; idy1 < n ; idy1++) //
```

```
        GCC_NOP(); //no operation, delay a instruction period
        GCC_NOP();
        for(idy2=0; idy2<4 ; idy2++)
        {
            GCC_NOP();
            GCC_NOP();
            GCC_NOP();
        }
    }
```

共陽極LED: PD<sub>n</sub>為”1”，選擇第n行的LEDs，由PE port的8個bits決定那些LED亮。



0x3f



```
void main(void) // 主函式
```

```
{
```

```
    unsigned int k, j, i, char_no ;
```

```
    _smod=0b11000000; // [7:5]=100->fH/16,110-->fH/4
```

```
    _pdc = 0 ; // 設定 PD 埠為輸出
```

```
    LedPortx = 0x00 ; // 8x8 LED X-axis (PD) OFF
```

```
    _pec=0; // 設定 PE 埠為輸出
```

```
    LedPorty = 0x00; // 8x8 LED Y-axis (PE) ON
```

```
    while(1)
```

```
    {
```

```
        for(char_no = 0 ; char_no < 4; char_no++) <----- 顯示字元總數目
```

```
        {
```

```
            for(j = 0 ; j < 100; j++) <----- 控制字元顯示時間
```

```
            {
```

```
                for( k = 0 ; k < 8 ; k++) <----- 迴圈執行總時間不得超過
```

```
                //The total execution time of the for-loop must be less than 1/16 seconds
```

```
                {
```

```
                    i=char_no*8+k;
```

```
                    LedPortx = 0x00 ; // 輸出 LED 控制碼, 關掉LEDx
```

```
                    LedPorty = 0x00 ; // 輸出 LED 控制碼, 關掉LEDx
```

```
                    LedPortx = LedTablex[k] ; // 輸出 LED 控制碼, 點亮 LEDx
```

```
                    LedPorty = LedTabley[i] ; // 輸出 LED 控制碼, 點亮 LEDx
```

```
                    delay(10); //delay can not be too long
```

```
                }
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

在切換到另一行LED顯示前，必須將LED全關暗，不然會因為視覺暫留現象，下一個將顯示的LED行，與前一行亮燈相同的位置之LED也會亮，也就是會產生類似殘影的現象。

NKNU\_EE\_MM50C\_RLWang

## Holtek –程式內容 / 8x8 LED Display 硬體架構與動作原理(行或列掃描)



```
#define LedPortx _pd // pd 埠
```

```
#define LedPorty _pe // pe 埠
```

```
LedPortx[0]=0x01
```

```
LedPortx[1]=0x02
```

```
LedPortx[2]=0x04
```

```
LedPortx[3]=0x08
```

```
LedPortx[4]=0x10
```

```
LedPortx[5]=0x20
```

```
LedPortx[6]=0x40
```

```
LedPorty[0]=0x00
```

```
LedPorty[1]=0x00
```

```
LedPorty[2]=0x00
```

```
LedPorty[3]=0x7f
```

```
LedPorty[4]=0x02
```

```
LedPorty[5]=0x08
```

```
LedPorty[6]=0x20
```

```
    k=0;
```

```
    PD
```

```
    0 1 2 3 4 5 6 7PE
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

```
LedPortx[7]=0x80
```

```
LedPorty[7]=0x7f
```

八行LED顯示掃描的總時間<1/16秒，視覺暫留效應導致眼睛感受到整個8x8LED顯示的圖案為”N”

```
unsigned char LedTablex[8] = { 0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x80 } ;
```

```
unsigned char LedTabley[32] = { 0x00, 0x00, 0x00, 0x7f, 0x02, 0x08, 0x20, 0x7f,
```

```
0x00, 0x00, 0x00, 0x7f, 0x08, 0x14, 0x22, 0x41,
```

```
0x00, 0x00, 0x00, 0x7f, 0x02, 0x08, 0x20, 0x7f,
```

```
0x00, 0x00, 0x00, 0x3f, 0x40, 0x40, 0x40, 0x3f} ;
```

← N

← K

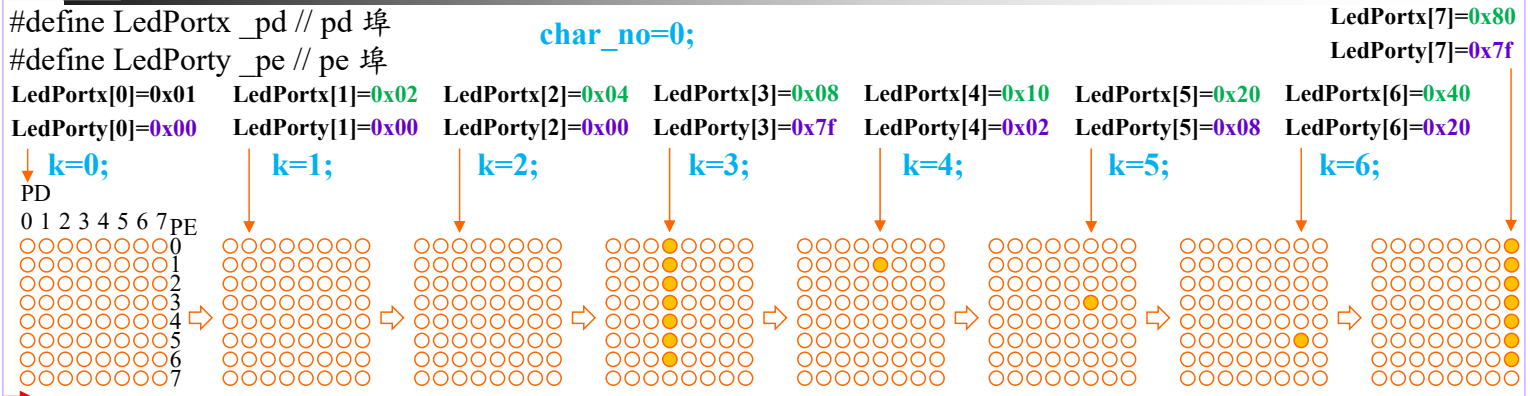
← N

← U

```
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
    0 1 2 3 4 5 6 7
```

NKNU\_EE\_MM50C\_RLWang



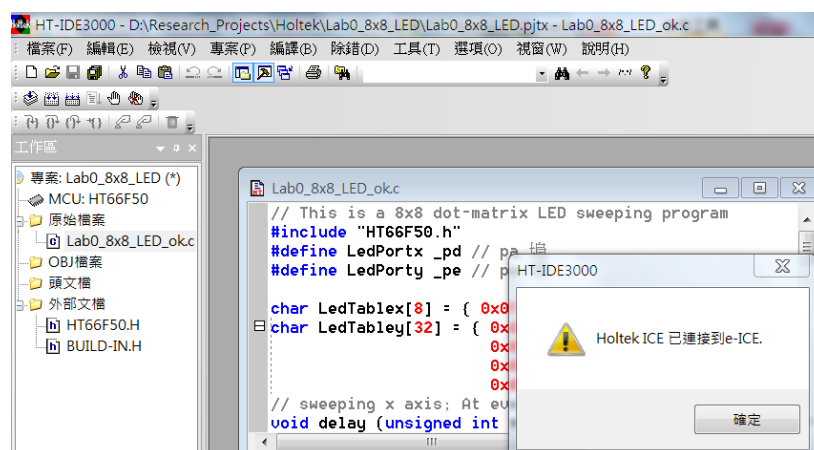
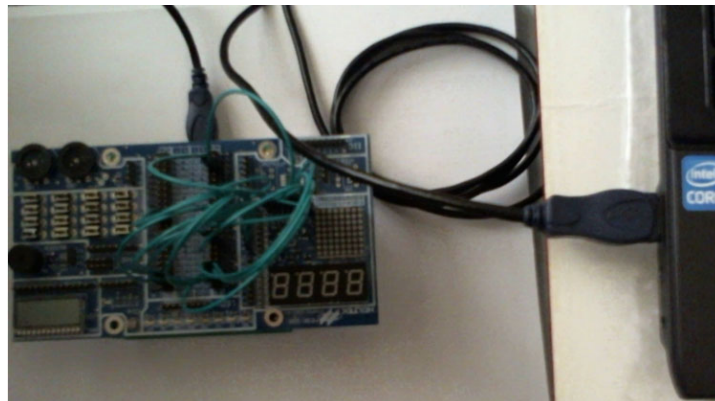


若未加入 `LedPortx = 0x00` ; `LedPorty = 0x00` ; 以先關掉LED，之後在設定下一行，將有可能產生殘影現象：

```
k=3;
i=char_no*8+k; //i=3
LedPortx = LedTablex[3]; // 0x08; 輸出 LED 控制碼, 點亮 LEDx
LedPorty = LedTabley[3]; // 0x7f; 輸出 LED 控制碼, 點亮 LEDx
delay(10);
k=4;
i=char_no*8+k; //i=4
LedPortx = LedTablex[4]; // 0x10; 輸出 LED 控制碼, 點亮 LEDx
LedPorty = LedTabley[4]; // 0x02; 輸出 LED 控制碼, 點亮 LEDx
delay(10);
k=5;
i=char_no*8+k; //i=5
LedPortx = LedTablex[5]; // 0x20; 輸出 LED 控制碼, 點亮 LEDx
LedPorty = LedTabley[5]; // 0x08; 輸出 LED 控制碼, 點亮 LEDx
delay(10);
```



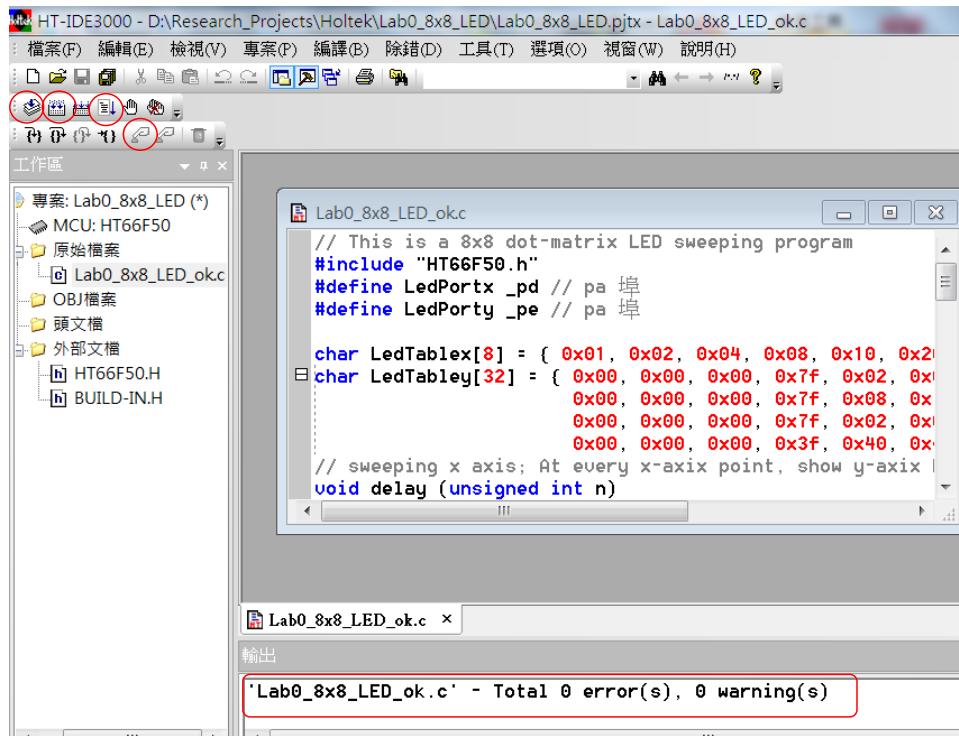
- 將ESK套件透過USB與電腦連接。連接成功會出現訊息"Holtek ICE已連接到e-ICE"，按下"確定"。



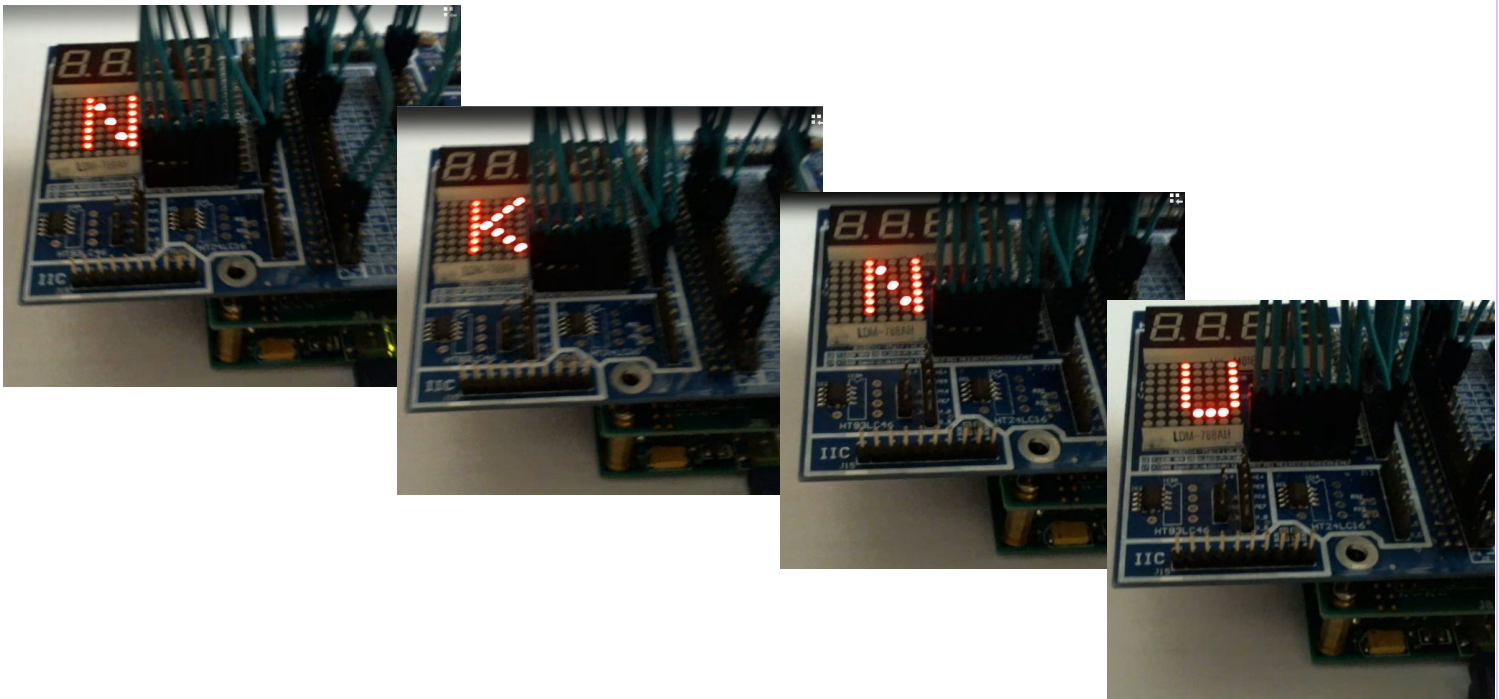


- 按下 icon(Assemble)，進行編譯、組譯，成功後，下方輸出訊息視窗會出現”0 error(s), 0 warning(0)”訊息。
- 按下 icon(Build)，進行硬體建置，成功後，下方輸出訊息視窗會出現”0 error(s), 0 warning(0)”訊息。  
若程式已經過Build，因需要而修改程式，經過Assemble後，可以按 icon(Rebuild All)，重新進行硬體建置。
- 按下 icon，執行程式。

可按 停止執行，按 重置程式。要修改程式前，一定要停止程式執行，才能進行程式修改。



- 執行結果：



### Assignment 1:

假設同一組的兩位同學的學號末兩碼是”03”, ”12”，設計程式讓8x8 LEDs依序顯示”N”, ”K”, ”N”, ”U”, ”E”, ”E”, ”0”, ”3”, ”|”, ”1”, ”2”，持續循環顯示。

可以的話，讓每個字停留約0.5秒。

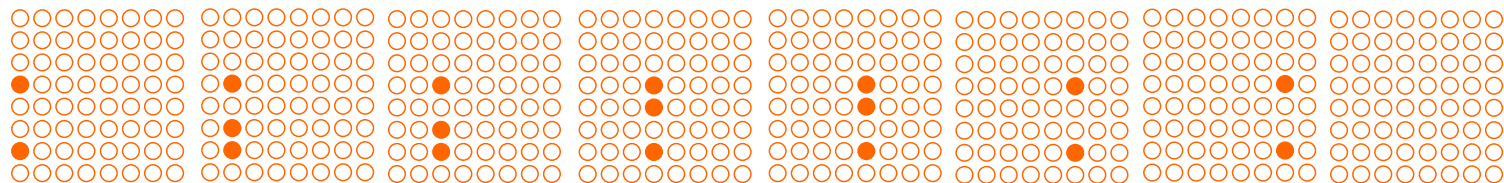




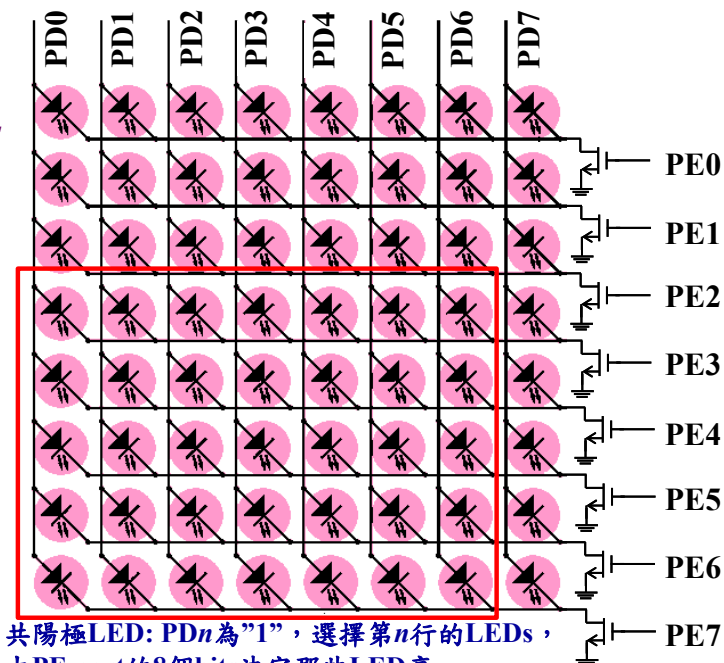
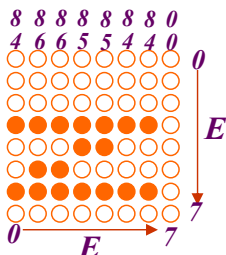
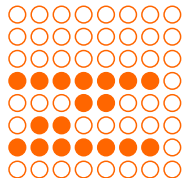
瞬間顯示:

透過迴圈執行八次“行”的顯示，迴圈執行總時間不得超過人眼視覺暫留時間(1/16秒)

PD: 0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x08  
PE: 0x48, 0x68, 0x68, 0x58, 0x58, 0x48, 0x48, 0x00

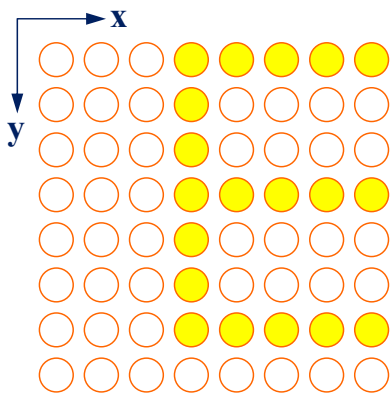


眼睛實際看到的畫面:



共陽極LED: PDn為“1”，選擇第n行的LEDs，由PE port的8個bits決定那些LED亮。

NKXNU\_EE\_MM50C\_RLWang



PD	PE
0x01	0x00
0x02	0x00
0x04	0x00
0x08	0x7F
0x10	0x49
0x20	0x49
0x40	0x49
0x80	0x49

X : PD7~PD0

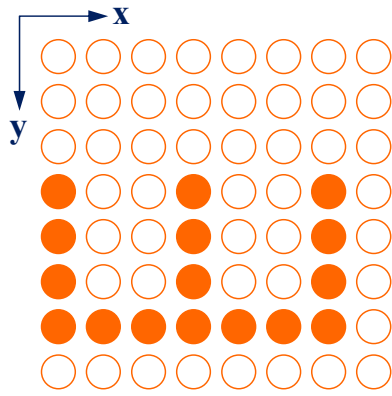
PDx=1

PDx=1, PEy=1 → LEDxy ON

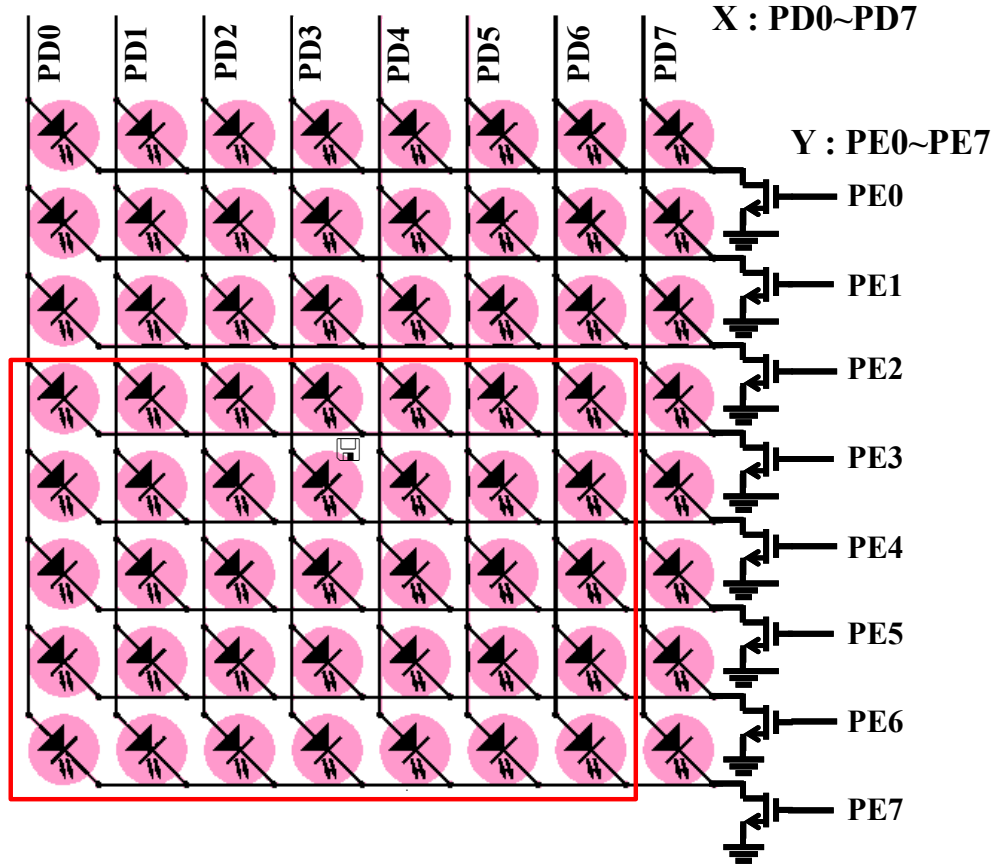


參考範例是用此版面(p.14)

NKXNU\_EE\_MM50C\_RLWang



PE	PD
0x78	0x01
0x40	0x02
0x40	0x04
0x70	0x08
0x40	0x10
0x40	0x20
0x78	0x40
0x00	0x80



0x78,0x40,0x40,0x70,0x40,0x40,0x78, // 0x45, [E]

利用5x7點矩陣字型編輯器產生字元組碼 → 實際展示範例是用此版面

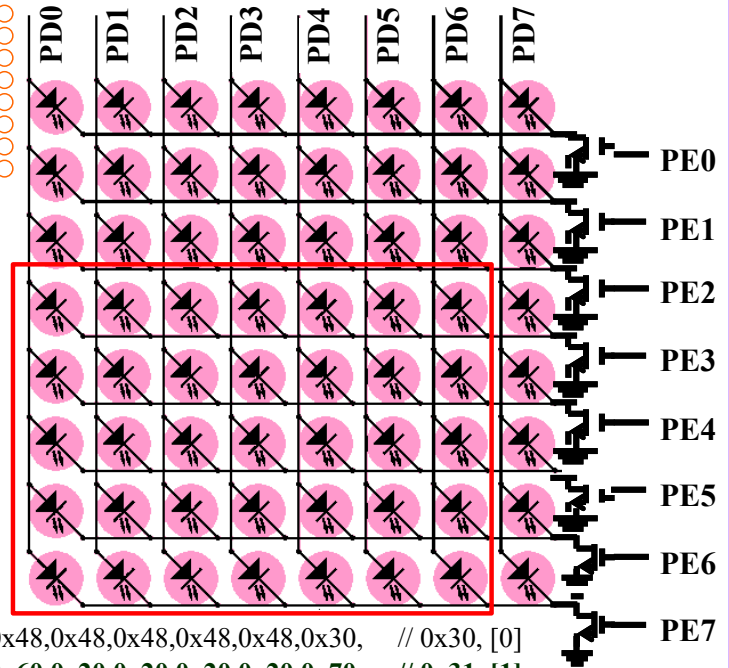
NK9NU\_EE\_MM50C\_RLWang



PE PORT :

0x30,0x48,0x48,0x48,0x78,0x48,0x48, // 0x41, [A]  
 0x70,0x48,0x48,0x70,0x48,0x48,0x70, // 0x42, [B]  
 0x30,0x48,0x40,0x40,0x40,0x48,0x30, // 0x43, [C]  
 0x70,0x48,0x48,0x48,0x48,0x48,0x70, // 0x44, [D]  
 0x78,0x40,0x40,0x70,0x40,0x40,0x78, // 0x45, [E]  
 0x78,0x40,0x40,0x70,0x40,0x40,0x40, // 0x46, [F]  
 0x30,0x48,0x40,0x58,0x48,0x48,0x30, // 0x47, [G]  
 0x48,0x48,0x48,0x78,0x48,0x48,0x48, // 0x48, [H]  
 0x70,0x20,0x20,0x20,0x20,0x20,0x70, // 0x49, [I]  
 0x08,0x08,0x08,0x08,0x48,0x48,0x30, // 0x4A, [J]  
 0x48,0x48,0x50,0x60,0x50,0x48,0x48, // 0x4B, [K]  
 0x40,0x40,0x40,0x40,0x40,0x40,0x78, // 0x4C, [L]  
 0x48,0x78,0x78,0x48,0x48,0x48,0x48, // 0x4D, [M]  
 0x48,0x68,0x68,0x58,0x58,0x48,0x48, // 0x4E, [N]  
 0x78,0x48,0x48,0x48,0x48,0x48,0x78, // 0x4F, [O]  
 0x70,0x48,0x48,0x70,0x40,0x40,0x40, // 0x50, [P]  
 0x30,0x48,0x48,0x48,0x68,0x58,0x38, // 0x51, [Q]  
 0x70,0x48,0x48,0x70,0x60,0x50,0x48, // 0x52, [R]  
 0x30,0x48,0x40,0x30,0x08,0x48,0x30, // 0x53, [S]  
 0x78,0x20,0x20,0x20,0x20,0x20,0x20, // 0x54, [T]  
 0x48,0x48,0x48,0x48,0x48,0x48,0x30, // 0x55, [U]  
 0x48,0x48,0x48,0x48,0x30,0x30,0x30, // 0x56, [V]  
 0x48,0x48,0x48,0x48,0x78,0x78,0x48, // 0x57, [W]  
 0x48,0x48,0x30,0x30,0x30,0x48,0x48, // 0x58, [X]  
 0x88,0x88,0x50,0x20,0x20,0x20,0x20, // 0x59, [Y]  
 0x78,0x08,0x10,0x30,0x20,0x40,0x78, // 0x5A, [Z]

這些數據  
是用軟體  
程式產生



0x30,0x48,0x48,0x48,0x48,0x48,0x30, // 0x30, [0]  
 0x20,0x60,0x20,0x20,0x20,0x20,0x70, // 0x31, [1]  
 0x30,0x48,0x08,0x10,0x20,0x40,0x78, // 0x32, [2]  
 0x30,0x48,0x08,0x30,0x08,0x48,0x30, // 0x33, [3]  
 0x10,0x30,0x30,0x50,0x50,0x78,0x10, // 0x34, [4]  
 0x78,0x40,0x40,0x70,0x08,0x08,0x70, // 0x35, [5]  
 0x30,0x48,0x40,0x70,0x48,0x48,0x30, // 0x36, [6]  
 0x78,0x48,0x08,0x10,0x20,0x20,0x20, // 0x37, [7]  
 0x30,0x48,0x48,0x30,0x48,0x48,0x30, // 0x38, [8]  
 0x30,0x48,0x48,0x38,0x08,0x48,0x30, // 0x39, [9]

NK9NU\_EE\_MM50C\_RLWang