

作業二：LED Control 設計說明

問題

何謂Memory Mapped I/O?(10%)

- 在CPU在CPU和外部設備之間執行輸入輸出的方法
- 使用相同的地址總線來尋I/O設備，前提是IO設備上的設備內存和寄存器都已經被映射到內存空間的某個地址。
- 當CPU訪問某個地址的時候，可能是要訪問某一部份物理內存，也可能是要訪問IO設備上的內存。

I/O Device 對應到之記憶體位址(可視為特定暫存器)，其功能可以分成哪三種？(10%)

- Control：下命令給裝置或設定
- Data：data in data out
- Status：紀錄裝置的處理狀態

何謂GPIO？(10%)

- General-purpose input/output
- 接腳可以供使用者由程式控制自由使用

WT58F2C932-bit Microcontroller提供幾個GPIO Ports? 每個Port各有幾根腳位？(10%)

- A、B、C、D、E、F，共6個port
- 每個port有16個腳位，除了F只有10個

ADP-WT58F2C9實驗板上LEDs接到哪一個GPIOPort？其對應到的Memory Address範圍？

- 連接到 Port B
- memory address range 是0x001F_6820 ~ 0x001F_6BFF

詳細說明以下幾個暫存器之功能及如何使用(設定)。

- 假設X=A
- PTX_GPIO
 - output data to portA[15:0]
 - 0:output low
 - 1:output high
- PTX_PADIN
 - input data from PortA[15:0]
- PTX_DIR
 - PortA[15:0] direction
 - 0:output mode
 - 1:input mode
- PTX_CFG
 - Reserved
 - PortA[15:0] configuration

	PTA_DIR = 0	PTA_DIR = 1
0	CMOS output	input without pull-up
1	open-drain output	input with pull-up

- PTX_PADINSEL
 - digital input buffer off

- 0:set PortA[x] as digital I/O
 - 1:set PortA[x] as analog input(AIN)
- PTX_FS
 - PortA digital special function selection

bit	PTA_DIR = 0	PTA_DIR = 1
15	GPIO	TMR7
14	GPIO	TMR7
13	GPIO	TMR7
12	GPIO	TMR7
11	GPIO	TMR2
10	GPIO	TMR2
9	GPIO	TMR2
8	GPIO	TMR2
7	GPIO	TMR9
6	GPIO	TMR9
5	GPIO	TMR9
4	GPIO	TMR9
3	GPIO	SPI3_NSS
2	GPIO	SPI3_SCK
1	GPIO	SPI3_MOSI
0	GPIO	SPI3_MISO

LED 燈號亮滅如何控制 ? (10%)

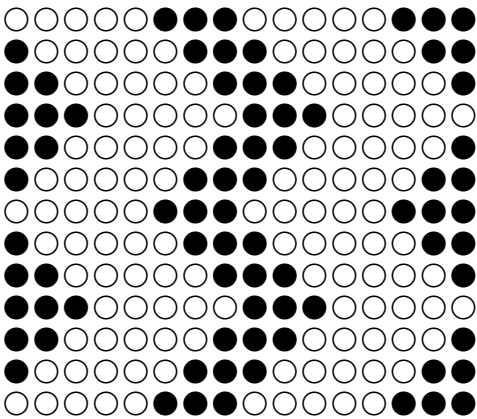
- 0 => 讓LED燈亮
- 1 => 讓LED燈滅

LED Control 設計說明

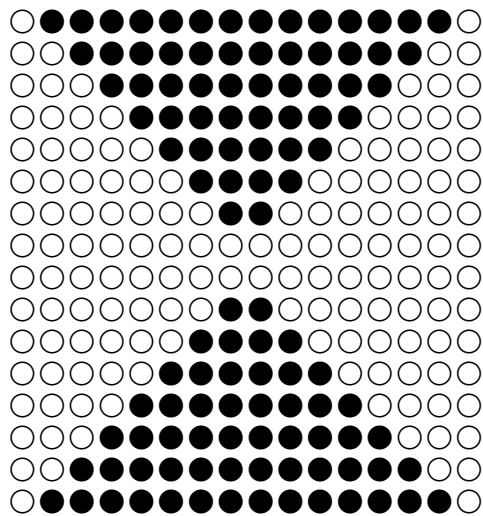
圖案解釋

- => 亮
- => 暗

- 第一種圖形
 - 33號
- 設計理念
 - 大一的回憶代表
- LED 燈號變化說明
 - 向右閃爍 接著向左 重複兩次
- 示意圖



- 第二種圖形
 - Join 運算子
- 設計理念
 - 資料庫很重要，要好好學
- LED 燈號變化說明
 - 由外向內，持續亮燈
- 示意圖



LED Control C程式

```

1  int main()
2  {
3
4
5      OS_PowerOnDriverInitial();
6
7
8      DRV_Printf("=====\r\n", 0);
9      DRV_Printf("  ADP-WT58F2C9 LED demo program  \r\n", 0);
10     DRV_Printf("=====\r\n", 0);
11
12
13     DRV_Printf("LED testing ...\r\n", 0);
14
15     GPIO_PTB_DIR = 0x0000; //for GPIO
16     GPIO_PTB_CFG = 0xFFFF; //for push pull
17     GPIO_PTB_PADIN = 0x00; //for Output
18
19
20     int i,j,w;
21     for(w=0 ; w<2 ; w++) // 兩個圖案 各跑兩次
22     {
23         // 圖案一 : 33
24         int a = 0xFFFF;
25         int b = 0x1F1F;
26         for(i=0 ; i<2 ; i++)
27         {
28             for(j=0 ; j<3 ; j++) // 33向右平移的部分
29             {
30                 GPIO_PTB_GPIO = a ^ b;
31                 delay1(600000);
32
33                 b = b << 1;
34             }
35             for(j=0 ; j<3 ; j++) // 33 向左平移的部分
36             {
37                 GPIO_PTB_GPIO = a ^ b;
38                 delay1(600000);
39
40                 b = b >> 1;
41             }
42
43             GPIO_PTB_GPIO = a ^ b;
44             delay1(600000);
45
46         }
47
48         // 圖案二 : join 運算子
49         a = 0x7FFF;
50         b = 0xFFFE;
51
52         for(i=0 ; i<8 ; i++) // 由外向內 持續亮燈
53         {
54             GPIO_PTB_GPIO = a & b;
55             delay1(600000);
56
57             a = a >> 1;
58             b = b << 1;
59         }
60
61         a = 0xFE00;
62         b = 0x007F;
63
64         for(i=0 ; i<8 ; i++) // 由內向外 將燈光熄滅
65         {
66             GPIO_PTB_GPIO = ~(a | b);
67             delay1(600000);
68
69             a = a << 1;
70             b = b >> 1;
71         }
72     }
73
74
75     DRV_Printf("=====\r\n", 0);
76
77
78     return 0;
79 }
80

```

