微處理機 LAB 2.1

Due： 兩周後 早上 8:00

PART 1. (22%)

1. 如果使得 GPIOB PIN8 符合以下所有條件
   1) Enable 2) 設定為輸出 3) Pulldown 4) Open-drain 5) Medium speed
   請列出位址及數值的參數為何?

1. Enable:
   a. 位址：RCC_AHB2ENR, 0x4002104C
   b. 參數：#0x6 (將 bit1 和 bit2 設為 1)
2. Output mode:
   a. 位址：GPIOB_MODER, 0x48000400
   b. 參數：0x10000
3. Pull down:
   a. 位址：GPIOB_PUPDR, 0x4800040C
   b. 參數：0x20000
4. Open-drain:
   a. 位址：GPIOB_OTYPER, 0x48000404
   b. 參數：0x80
5. Medium speed:
   a. 位址：GPIOB_OSPEEDR, 0x48000408
   b. 參數：0x10000

```
/*
 * part1.s
 *
 * Created on : 3/24 Fri
 *      Author: alfonso
 */

.syntax unified
.cpu cortex-m4
.thumb

.data
    Leds: .byte 0

.text
    .global Init_PB
    .global main

    // PB setup
    .equ GPIOB_MODER,   0x48000400
    .equ GPIOB_OTYPER,  0x48000404
    .equ GPIOB_OSPEEDR, 0x48000408
    .equ GPIOB_PUPDR,   0x4800040C
    .equ GPIOB_ODR,     0x48000414

    // Clock setup
    .equ RCC_AHB2ENR,   0x4002104C

```

```
Init_PB:

    // Enable AHB2 clock
    movs r0, #0x6
    ldr r1, =RCC_AHB2ENR
    str r0, [r1]

    // Set PB8 as output mode
    ldr r1, =GPIOB_MODER
    ldr r0, [r1]
    ldr r3, =0xFFFCFFFF // 1111 1111 1111 1100 1111 1111 1111 1111
    ldr r4, =0x10000    // 0000 0000 0000 0001 0000 0000 0000 0000
    and r0, r3
    orr r0, r4
    str r0, [r1]

    // Set PB8 as pull-down
    ldr r1, =GPIOB_PUPDR
    ldr r0, [r1]
    ldr r3, =0xFFFCFFFF // 1111 1111 1111 1100 1111 1111 1111 1111
    ldr r4, =0x20000    // 0000 0000 0000 0010 0000 0000 0000 0000
    and r0, r3
    orr r0, r4
    str r0, [r1]

    // Set PB8 as open-drain
    ldr r1, =GPIOB_OTYPER
    ldr r0, [r1]
    ldr r3, =0xFFFFFF7F // 1111 1111 1111 1111 1111 1111 0111 1111
    ldr r4, =0x80       // 0000 0000 0000 0000 0000 0000 1000 0000
    and r0, r3
    orr r0, r4
    str r0, [r1]

    // Set PB8 as medium speed
    ldr r1, =GPIOB_OSPEEDR
    ldr r0, [r1]
    ldr r3, =0xFFFCFFFF // 1111 1111 1111 1100 1111 1111 1111 1111
    ldr r4, =0x10000    // 0000 0000 0000 0001 0000 0000 0000 0000
    and r0, r3
    orr r0, r4
    str r0, [r1]

    ldr r1, =GPIOB_ODR
    bx lr

main:
    Init_PB
    /* Do something with PB8 */
L: b L
```

2. 請將學號最後一碼轉換為 2 進制 (2%)
   My ID: 110611063
   dec(3) = bin(11)

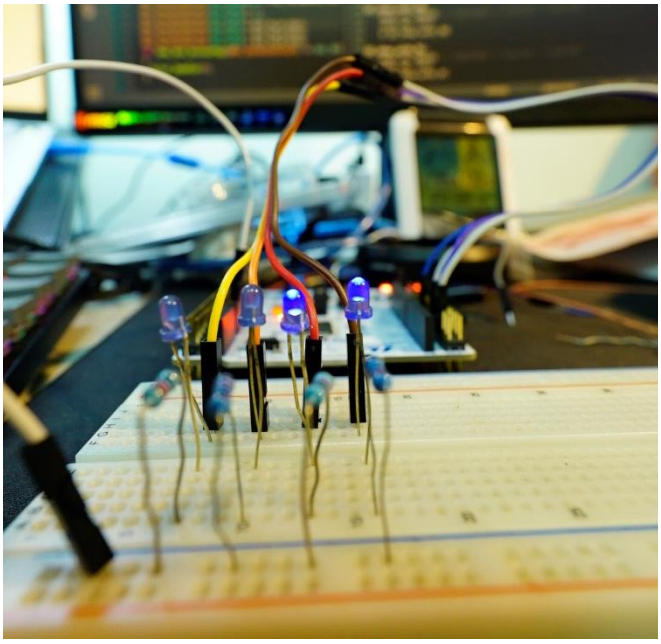
PART 2. (50%) 實作題

請完成實驗 拍照記錄實驗結果 截圖紀錄實驗結果並附上程式碼(main.s 及 include 之 pin.s 檔案)

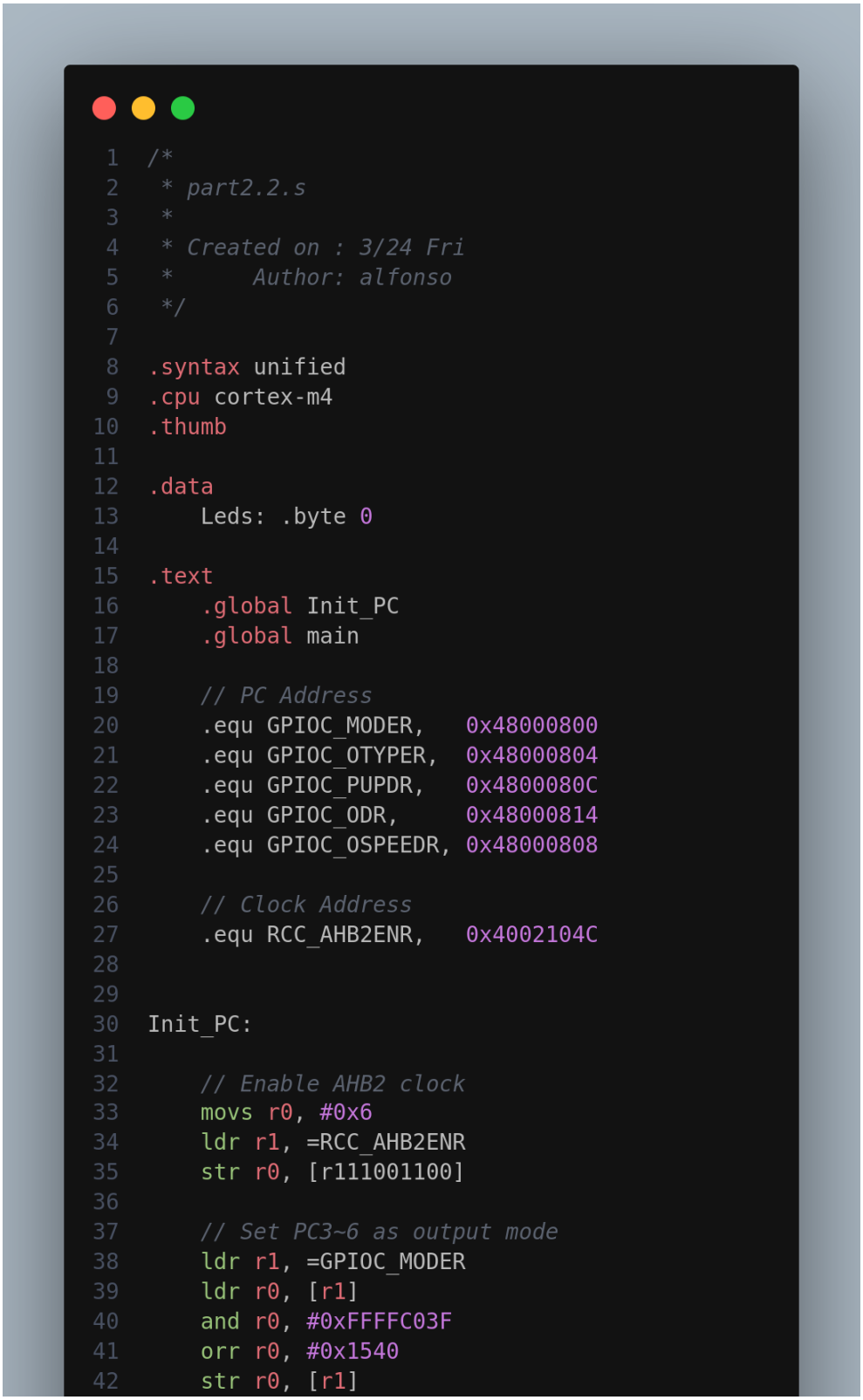
1. 參照課堂練習，將四顆 LED 分別接於 PB3-PB6 四個腳位，控制使四個 LED 亮起(Active Low)，燈號為學號最後一碼之 2 進制結果 (38%)

```
/*
 * part2.1.s
 *
 * Created on : 3/24 Fri
 *      Author: alfonso
 */

.syntax unified
.cpu cortex-m4
.thumb

.data
    Leds: .byte 0

.text
    .global Init_PB
    .global main

    // PB Address
    .equ GPIOB_MODER,   0x48000400
    .equ GPIOB_OTYPER,  0x48000404
    .equ GPIOB_OSPEEDR, 0x48000408
    .equ GPIOB_PUPDR,   0x4800040C
    .equ GPIOB_ODR,     0x48000414

    // Clock Address
    .equ RCC_AHB2ENR,   0x4002104C

Init_PB:

    // Enable AHB2 clock
    movs r0, #0x6
    ldr r1, =RCC_AHB2ENR
    str r0, [r111001100]

    // Set PB3~6 as output mode
    ldr r1, =GPIOB_MODER
    ldr r0, [r1]
    and r0, #0xFFFFC03F
    orr r0, #0x1540
    str r0, [r1]
```

```
43      // Set PB3~6 as pull-up
44      ldr r1, =GPIOB_PUPDR
45      ldr r0, [r1]
46      and r0, #0xFFFFC03F
47      orr r0, #0x1540
48      str r0, [r1]
49
50      // Set PB3~6 as high speed
51      ldr r1, =GPIOB_OSPEEDR
52      ldr r0, [r1]
53      and r0, #0xFFFFC03F
54      orr r0, #0x2A80
55      str r0, [r1]
56
57      ldr r1, =GPIOB_ODR
58      bx lr
59
60  main:
61
62      // Init
63      bl Init_PB
64
65      // My ID is 110611063
66      // dec(3) -> bin(11) -> show pattern 0011
67      // activate PB3 and PB4 -> write bin(1100)
68      // bin(1100) -> hex(18)
69      movs r0, #0x18
70      strh r0, [r1]
71  L: b L
```

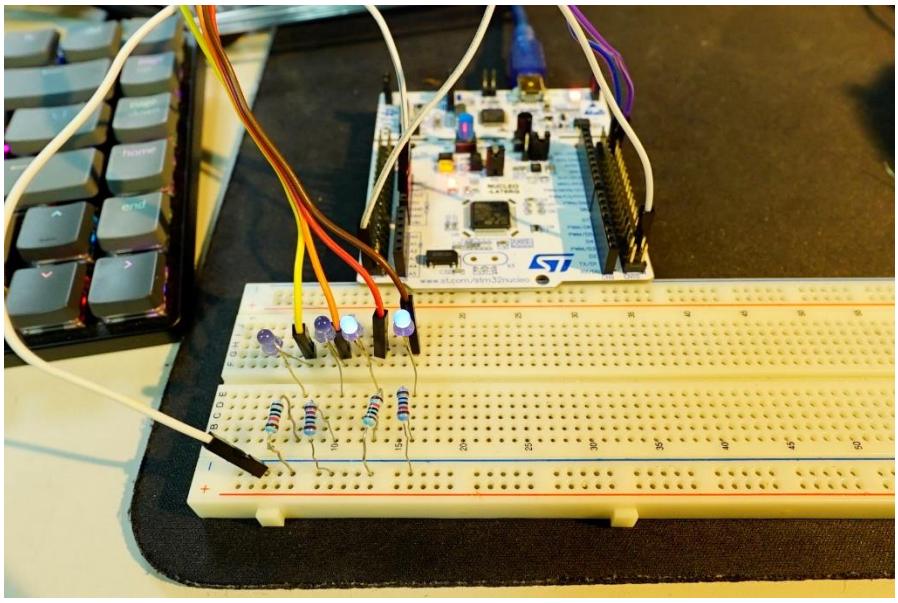


2. 將 PIN 腳轉換為 PC2-PC5 重複 PART2-1 實驗(40%)



```
/*
 * part2.2.s
 *
 * Created on : 3/24 Fri
 *      Author: alfonso
 */

.syntax unified
.cpu cortex-m4
.thumb

.data
    Leds: .byte 0

.text
    .global Init_PC
    .global main

    // PC Address
    .equ GPIOC_MODER,   0x48000800
    .equ GPIOC_OTYPER,  0x48000804
    .equ GPIOC_PUPDR,   0x4800080C
    .equ GPIOC_ODR,     0x48000814
    .equ GPIOC_OSPEEDR, 0x48000808

    // Clock Address
    .equ RCC_AHB2ENR,   0x4002104C


Init_PC:

    // Enable AHB2 clock
    movs r0, #0x6
    ldr r1, =RCC_AHB2ENR
    str r0, [r111001100]

    // Set PC3~6 as output mode
    ldr r1, =GPIOC_MODER
    ldr r0, [r1]
    and r0, #0xFFFFC03F
    orr r0, #0x1540
    str r0, [r1]
```

```
44      // Set PC3~6 as pull-up
45      ldr r1, =GPIOC_PUPDR
46      ldr r0, [r1]
47      and r0, #0xFFFFC03F
48      orr r0, #0x1540
49      str r0, [r1]
50
51      // Set PC3~6 as high speed
52      ldr r1, =GPIOC_OSPEEDR
53      ldr r0, [r1]
54      and r0, #0xFFFFC03F
55      orr r0, #0x2A80
56      str r0, [r1]
57
58      ldr r1, =GPIOC_ODR
59      bx lr
60
61  main:
62
63      // Init
64      bl Init_PC
65
66      // My ID is 110611063
67      // dec(3) -> bin(11) -> show pattern 0011
68      // activate PB3 and PB4 -> write bin(1100)
69      // bin(1100) -> hex(18)
70      movs r0, #0x18
71      strh r0, [r1]
72  L: b L
```

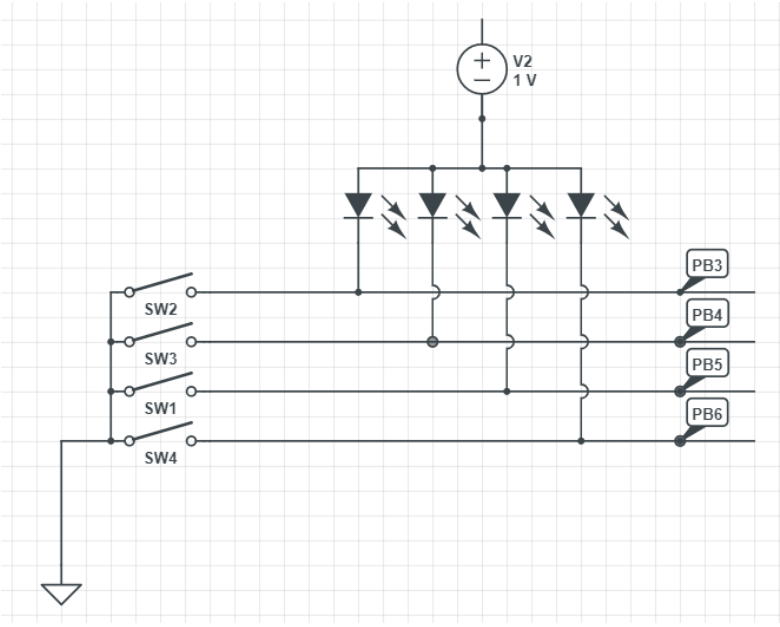


PART 3. 加分練習，不計入平常成績

隱藏密碼 ： 利用 DIP switch(紅色開關)與 PART2-1 的實驗組合成一個隱藏著密碼的燈號。若在開關上撥出正確的密碼時 LED 燈號會全部亮起。

密碼設定必須為 4 位數的 2 進制數字，其中必須包含 1 跟 0。(Ex: 1100、1010、0110，不可為 0000 或 1111)

參考電路設計: (不只一種方法 只要完成題目所述目的皆可)



我們的電路設計:

密碼為 0011，將兩個正解為關閉的開關與 LED 並聯，當這兩個中任一個被打開時電流會從被開啟的開關流向接地，此時沒有電流通過 LED 燈；將兩個正解為開啟的開關與四個 LED 燈並聯，當這兩個當中任一個被關閉時，所有 LED 燈會呈現斷路狀態