

一、【實驗目的】：

What was your design? What were the concepts you have used for your design?

- 實驗目的: 操作 LCD 並結合七段顯示器和 LED 和蜂鳴器，整合從開學學到現在的所有元件，熟悉使用，了解使用 LCD 的好處。
- 設計理念: 利用 LCD 會自己處理顯示的好處來節約在寫程式上面的困難度，並且整合所有學過的元件。

二、【遭遇的問題】：

What problems you faced during design and implementation?

1. 在使用 LCD 印出文字時，七段顯示器會閃爍。
2. 在格式化輸入 LCD 每行文字並讓它顯示時，結果不如預期，如果想控制在第一行顯示文字，但是第二行或第三行也會跑出不規則黑點。

三、【解決方法】：

How did you solve the problems?

1. 不要在每次迴圈都做 `clear_LCD`，並重新顯示 LCD，因為如果沒有做文字的更新就這樣一直清除並顯示會太花資源，導致 `cpu` 的延遲太高讓七段顯示器沒辦法好好刷新。改成有更動文字內容再重新顯示 LCD 就好。
2. 發現是最一開始的 `char` 陣列輸入的時候就 `over flow` 了，還有在 `sprintf` 時給的文字內容大小超過字元陣列大小也會導致 `over flow`，如果 `over flow` 就會導致使用 `print_Line` 時印出的文字再不如預期的位子。

四、【未能解決的問題】：

Was there any problem that you were unable to solve? Why was it unsolvable?

- 按 keypad 輸入密碼時，每次做 LCD 更新時七段顯示器還是會快速閃一下
- 原因: `print_Line` 會把 LCD 切成 4 行，並將想更新的那行作全部更新，但是這樣程式裡的迴圈會做太多次，並且有很多都是沒意義的，所以如果有時間可以考慮用 `putc` 寫應該會節省掉時間，就可以讓七段顯示器比較不會閃爍。

五、【問題】：

1. 為甚麼 `MCU_init.h` 要加入 `Define MCU Interfaces`?

A: 在 `MCU Interfaces` 中有定義 `PIN_SPI3`，這是 LCD 會使用到的 PIN 腳，所以必須設定不然不能使用 LCD。

```
#define MCU_INTERFACE_SPI3
```

```
#define SPI3_CLOCK_SOURCE_HCLK // HCLK, PLL
```

```
#define PIN_SPI3_SS0_PD8: SS (Slave Select): 片選訊號，由主機發出，一般是
```

低電位有效

#define PIN_SPI3_SCLK_PD9: SCLK (Serial Clock): 串列時脈, 由主機發出

#define PIN_SPI3_MISO0_PD10: MISO (Master Input, Slave Output): 主機輸入從機輸出訊號 (資料由從機發出)

#define PIN_SPI3_MOSI0_PD11: MOSI (Master Output, Slave Input): 主機輸出從機輸入訊號 (資料由主機發出)

2. 為甚麼以下 code Text 這個陣列使用 Print_Line 印出時, 在第 10 格的 Hello 沒有印出來?

A: 因為在執行 `printf(Text, "%d&&%d = %d", 1, 0, 0)` 的時候會把字元陣列的第 8 格後面都設置為 `\0`, 如果我們使用 `printf(Text+8, "HELLO")`, 會發現可以印出文字, 但是 `printf(Text+10, "HELLO")` 卻不行了, 因為陣列的第八第九格都是 `\0`, 所以當他要印出時讀到第八格編譯器就判斷字串結束了, 所以才不會印出。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Text[16]															
2	1	&	&	0		=		0	\0	\0	\0	\0	\0	\0	\0	\0
3	1	&	&	0		=		0	H	E	L	L	O	\0	\0	\0
4	1	&	&	0		=		0	\0	\0	H	E	L	L	O	\0

第二行為執行 `printf(Text, "%d&&%d = %d", 1, 0, 0)` 的時候 Text[16] 的陣列狀態

第三行為執行 `printf(Text+8, "HELLO")` 的時候 Text[16] 的陣列狀態

第四行為執行 `printf(Text+10, "HELLO")` 的時候 Text[16] 的陣列狀態, 可以看出第八第九格是 `\0`, 導致輸出的時候判斷不如預期。