微處理機系統實習　　Lab5

班級：資訊三丁 學號：D1053020 姓名：徐葆驊

一、【實驗目的】:

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

- 實驗目的: 經過上次的 Lab 課知道怎麼使用 LCD 後就在更深度的去運用 LCD，自己設計要印出的圖案，讓以後可以更靈活的使用 LCD 的顯示，更了解 draw_Pixel 和 lcdWriteData。
- 設計理念: 將想印出的圖案利用陣列存起來，在陣列裡面表示 16 進制，0 代表不亮，1 代表亮，在使用 draw_Bmp 將自己設計的圖案印出。

二、【遭遇的問題】:

What problems you faced during design and implementation?

1. 在 5-1 的實驗中，當長方形跑到邊界並且按下正在前行的方向的按鈕(2、4、6、8)，長方形會直接卡入邊界中。
2. 在 5-1 的實驗中，長方形直的跟橫的對於邊界的判定不相同。導致其中一個狀態可以剛好觸碰到邊界時，另一個狀態就碰不到邊界。
3. 在 5-2 實驗中，選格子的時候要讓該格閃爍，但是在閃爍的時候會把遊戲的格線給消除。
4. 在 5-2 實驗中，當同一格連續按兩次選擇鍵會把原本的 O 或 X 覆蓋掉。

三、【解決方法】:

How did you solve the problems?

1. 將按按鈕的時候新增判斷現在圖形所在的位子，如果在邊界了就固定住圖形的位子在邊界，不能繼續移動，除非沒有繼續按要反彈了。
2. 在判斷是否觸碰到邊界的時候將橫的直的分兩種狀態判斷，因為長方形的長和寬不一樣長，並且判斷點是取長方形的左上角的頂點，所以不能一起判斷。
3. 將 fill_Rectangle 的長寬縮小，這樣在畫的時候就不會把邊界畫到，就不會把格線刪除。但其實也可以將畫格線放入 while 迴圈裡，這樣雖然會多消耗一點 cup 的，但是還在可接受範圍。
4. 在按下確認選擇的時候，多判斷該格是不是已經被選過了，如果選過的話就忽略等到選到沒選過的格子。

四、【未能解決的問題】:

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

- 我的板子給別人用的時候，那個人先往下到了最下面那行並按了選擇，然後之後案的選擇都會在選到的格子的上面一行。
- 原因:我認為可能是選擇完後，有改動到選擇格的 x y 座標，所以才導致選錯格，或著是計算選到的格子的演算法寫錯。

五、【問題】:

1. 程式中的<<是甚麼意思? 假設今天 k 是 3 時 0X01<<k 之後回傳的答案是多少?
   A:<<為左位移運算子，x<<y 代表將 x 轉換成 2 進位然後向左位移 y 個位元，左移出的位元會被丟棄，右側會補上 0。
   若 0X01<<3 時，(00000001)左位移 3 個位元變成(00001000)，等於 16 進位的 0X08，所以會回傳 0X08。
2. fgColor 跟 bgColor 是甚麼意思?分別用在哪邊?
   A:fgColor 能控制想要畫的點會不會亮(黑色)，如果 fgColor 給 0 那在 draw_Pixel 判斷的時候就不會亮，如果給 1 就會亮，通常 fgColor 跟 bgColor 會設置為 0XFF 和 0X00，這樣在呼叫其他 draw 的 function 的時候會比傳給副程式 0 1 來的直覺好讀一點。但是看 draw_Pixel 的副程式，只會判斷到 fgColor，bgColor 並沒有用到，bgColor 可能是宣告給使用程式碼的人看的，讓別人知道不想亮的時候給 0 就不會亮。