組別：6，組員：王柏堯、張彧豪

微處理機 Lab8 Report

1. 實驗名稱：STM32 Interrupt and Exception
2. 實驗目的：

* 瞭解STM32 SysTick timer設定。
* 瞭解STM32 NVIC和External interrupt設定和原理。

1. 實驗步驟：
2. SysTick timer interrupt & DS18B20 & One wire protocol。
3. Keypad external interrupt。
4. 實驗結果與分析：
5. SysTick timer interrupt & DS18B20 & One wire protocol：

第一題算是兩題中較難的一題，我們覺得它難的地方就在於one-wire protocol的實作以及數位溫度計的操作，實作的原理其實不會太難，講義上都有講解，唯一讓我們卡住的點在於在reset的最後，除了要等60ns去收input值外，還要另外等大概420ns，讓整個reset的過程完整，一開始忘了做這部份，導致初始化沒有成功，困惱我們很久。而另外一個問題就是delay的問題，這邊要delay的時間都是微秒級，況且我們是使用預設的system clock rate(4Mhz)，只要4個cycle就1微秒了，加上我們一開始是用呼叫函數的方式在操作GPIO，導致常常指令做完，那個資料維持的時間或規定的時間長度也跑掉了，為了解決這個問題，我們只好放棄使用函數的方式，改為類似組語那樣的寫法直接去存取GPIO相關的暫存器，讓必要的存取不要浪費太多的clock cycle導致無法正確讀取到資料。

因為溫度變化不明顯的關係，我們也採取輸出小數點的方式，不過這邊有個問題卡住我們就是當我們要做浮點數運算時，不知為何的都會被trap到無窮迴圈內，最後才發現，要把FPU unit開啟，不然做浮點數運算一定會trap到無窮迴圈內，因為CPU認不得那些指令。

1. Keypad external interrupt：

這題算是兩題中比較簡單的一題，這題的做法是一開始先將SysTick給開起來，並設定好它所需的clock source以及reloader的值。這邊一開始遇到的困難的是找不到SysTick相關的register，最後找了資料才發現Systick以及NVIC都是處理器內建的register，難怪在STM32的reference找不到。接著就要設定EXTI模組，這邊我們一共使用四個EXTI模組(EXTI12, EXTI13, EXTI14, EXTI15)，要設定EXTI模組須先開啟SYSCFG的功能(經由APB2ENR來開啟)，接著選各個EXTI模組中斷的source port，接著設定EXTI的IMR register，讓我們想開啟中斷的EXTI模組不會被mask掉，再來設定什麼時候會trigger此中斷，我們這邊設定的是上緣觸發，當電壓被拉上來時會觸發中斷發生。最後是設定NVIC以及把中斷routine填在startup\_stm32.s中，設定NVIC讓我們設定的中斷可以被CPU看到以及將EXTI中斷routine填在正確的位置，這邊我們也卡了很久，最後發現是沒有將routine填在正確位置，填在正確位置及設好優先級後就可以執行了。

在執行的時候我們發現了另外一個問題，就是當我們按某個按鍵後，常常顯示出來的值都不是正確數值，而是跟它同一個row的其他按鍵數值，一直不懂該怎麼解決這個問題以及問題發生的原因，我們的改善方法是把Display這個函數放在EXTI\_handler裡面，只有當中斷發生時才執行，這樣顯示錯誤的機率就會降低不少，不過可能還需要仔細了解這個問題的成因才會有更好的解決辦法。

1. 心得討論與應用聯想：

這次的作業雖然時間給比較多，但因為最近作業太多，所以我們仍然也只有幾天能夠做，先審題後，發現第二題比較簡單，所以先做第二題，只要開好Interrupt表後，大致上即可使用，但我們卻發現數字顯示出來是同一排的其他數字，因此判斷是又被systick占掉，所以要讓它不能用，所以便把寫值這件事寫在Interrupt內；另一題因為溫度計的協議較複雜，且GPIO的函式好像也不能直接使用，所以實際上也DEBUG了許久，好不容易讀到值了，卻也發現值不太對，最後再修正許多後，終於完成。