實驗八 STM32 Interrupt and Exception

1. 實驗目的

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

2. 實驗原理

請參考上課講義。

3. 實驗步驟

Lab8.1: SysTick timer interrupt & DS18B20 & One wire protocol

- 實作一個SysTick interrupt handler，當中斷發生時顯示當前溫度於七段顯示器上。
- 瞭解 OneWire Protocol 並且實作 onewire.c, onewire.h
- 瞭解 DS18B20 Protocol 並且實作 ds18b20.c, ds18b20.h
- 溫度顯示整數部分即可
- 當使用者按下user button開啟或關閉SysTick timer。

- Implement a SysTick interrupt handler. Show current temperature on 7-segment display when interrupt has been triggered.
- Understand OneWire Protocol and implement onewire.c, onewire.h
- Understand DS18B20 Protocol and implement ds18b20.c, ds18b20.h
- It is fine to show integral part of temperature value only.
- Enable or disable SysTick timer when user push user button.

Notes: 設定SysTick clock source為10MHz，SysTick timer每 2秒 interrupt一次。請利用操作NVIC的interrupt mask register的方式開啟或關閉Systick timer (not disable all interrupts)，NVIC相關register請參閱Programming manual section 4.3。

Notes: Set SysTick clock source to 10MHz. System timer triggers interrupt every 2 seconds. Please use NVIC interrupt mask register to enable or disable Systick timer. Please reference Programming manual section 4.3 for NVIC registers.

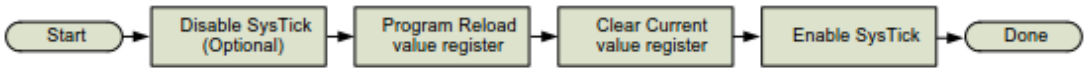

Figure 10.4:
Setup sequence for SysTick timer.

```
void SystemClock_Config(){
    //TODO: Setup system clock and SysTick timer interrupt
}
void SysTick_Handler(void) {
    //TODO: Show temperature on 7-seg display
}
int main(){
    SystemClock_Config();
    GPIO_init();
    while(1){
        if(user_press_button())
        {
            //TODO: Enable or disable Systick timer
        }
    }
}
```

3.1. Lab8.2: Keypad external interrupt (30%)

這部分的實驗主要請同學將Lab6中所實作的鍵盤掃描程式改成利用SysTick與外部中斷EXTI完成(無須掃描迴圈)。主要原理由以下3個部分完成。

- 將Column output掃描由SysTick interrupt handler完成，中斷時間間隔0.1s，當SysTick中斷發生時更改scan column。
- 當EXTI中斷發生時讀取4個input的值，並根據目前column掃描狀態判斷是哪個鍵按下。

在主程式中依使用者所按下的按鍵值利用lab6的display()顯示至7段顯示器上。

	X0	X1	X2	X3
Y0	1	2	3	10
Y1	4	5	6	11
Y2	7	8	9	12
Y3	15	0	14	13

請依以下TODO說明完成程式碼

```
char key_value = 0;
void EXIT_Setup(){
    //TODO: Setup EXTI interrupt
}
void SystemClock_Config(){
    //TODO: Setup system clock and SysTick timer interrupt
}
void SysTick_Handler(void) {
    //TODO: Scan the keypad column
}
void EXTIx_IRQHandler(void){
    //TODO: Read the keypad row value
}
int main(){
    SystemClock_Config();
    GPIO_init();
    EXTI_Setup();
    while(1){
        display(key_value,2);
    }
}
```