

電工實驗(四)

數位實驗(二) 中斷處理程式與時鐘

班級：電機三乙

組別：第 12 組

學號：B103012002

姓名：林凡皓

一、 程式碼：

```
#include "stdio.h"
#include "NuMicro.h"
#include "tmr.h"
#include "system_init.h"
#include "GUI.h"
#include "display.h"

/* define */
#define MaxSpeed    50000
#define MinSpeed    1000
#define SW_UP       PC9           //UP           JoyStick
#define SW_DOWN     PG4           //DOWN        JoyStick
#define SW_CTR      PG3           //CENTER      JoyStick

/* 宣告變數 */
uint32_t SpeedCtl;
uint32_t timecount;
uint32_t sec = 0;
uint32_t hour = 0;
uint32_t min = 0;

/* 宣告函數 */
void Clock_Task(void);
void clock_init(void);
void clock_tick(void);
void LED_showing(uint32_t SpeedCtl);
void GPIO_init(void);
uint32_t JoyStick(unsigned char BTN_state);

/* 定義一個union為Joystick_union，可以透過JState來存取struct的值 */
typedef union{
    struct{
        unsigned UP      :1;
        unsigned DOWN    :1;
        unsigned CTR      :1;
    };
    unsigned char JState;
}Joystick_union;

/* 透過 JSUnion 來存取 Joystick_union */
Joystick_union JSUnion;

int main(void)
{
    /* initialize */
    char clock_buf[20];
    char speed_buf[20];
    uint32_t speed;

    SYS_Init();

    TMR0_Initial();

    clock_init();

    GPIO_init();

    Display_Init();

    while(1)
    {
        /* 定義JSUnion的動作 */
        JSUnion.UP = SW_UP;
        JSUnion.DOWN = SW_DOWN;
        JSUnion.CTR = SW_CTR;

        SpeedCtl = JoyStick(JSUnion.JState); //呼叫Joystick並傳入JSUnion來控制要加速、減速或停止
        speed = 51000 - SpeedCtl; // 定義速度

        LED_showing(SpeedCtl); //LED toggle function
        clock_tick(); //呼叫clock_tick

        /* define how to show the clock */
        sprintf(clock_buf, "%02d:%02d:%02d", hour, min, sec);
        sprintf(speed_buf, "speed = %.1f (s)", speed / 10000.0);

        Display_buf(clock_buf, 270, 1);
        Display_buf(speed_buf, 1, 1);
    }
}
```

```

/* GPIO initialize */
void GPIO_init(void)
{
    GPIO_SetMode(PA, BIT0, GPIO_MODE_INPUT) ;    // Sw1
    GPIO_SetMode(PH, BIT6, GPIO_MODE_OUTPUT) ;   // LEDR1
    GPIO_SetMode(PH, BIT7, GPIO_MODE_OUTPUT) ;   // LEDG1
    GPIO_SetMode(PC, BIT9, GPIO_MODE_INPUT) ;     // Joystick_UP
    GPIO_SetMode(PG, BIT4, GPIO_MODE_INPUT) ;     // Joystick_DOWN
    GPIO_SetMode(PG, BIT3, GPIO_MODE_INPUT) ;     // Joystick_CENTER
}

//time initialize
void clock_init(void)
{
    sec = 0;
    min = 0;
    hour = 0;
}

/* 定義計時 */
void clock_tick(void)
{
    static uint32_t old_timecount = 0;

    if((uint32_t)(timecount - old_timecount) < 10000)
        return;

    old_timecount = timecount;
    sec++;
    if (sec == 60)
    {
        sec = 0;
        min++;
        if (min == 60)
        {
            min = 0;
            hour++;
            if (hour == 24)
                hour = 0;
        }
    }
}

/* 定義Joystick函數來決定要加速、減速或式停止 */
uint32_t JoyStick(unsigned char BTN_state)
{
    static uint32_t old_timecount = 0;
    static uint32_t SpeedCtl = 41000;

    if ((uint32_t)(timecount - old_timecount)<1000)
        return SpeedCtl;

    old_timecount = timecount;

    switch(BTN_state)
    {
        case 0x06: // Joystick up
            if (SpeedCtl < MaxSpeed) //如果速度小於最大速度，就加速
                SpeedCtl -= 1000;
            else
                SpeedCtl = SpeedCtl; //否則，維持速度
            break;

        case 0x05: // Joystick down
            if (SpeedCtl > MinSpeed) //如果速度大於最小速度，就減速
                SpeedCtl += 1000;
            else
                SpeedCtl = SpeedCtl; //否則，維持速度
            break;

        case 0x03: // Joystick center
            SpeedCtl = 51000; //由於speed的定義為51000-SpeedCtl，因此讓SpeedCtl為51000即讓speed=0
            break;

        default:
            SpeedCtl = SpeedCtl;
            break;
    } //switch
    return SpeedCtl;
}

```

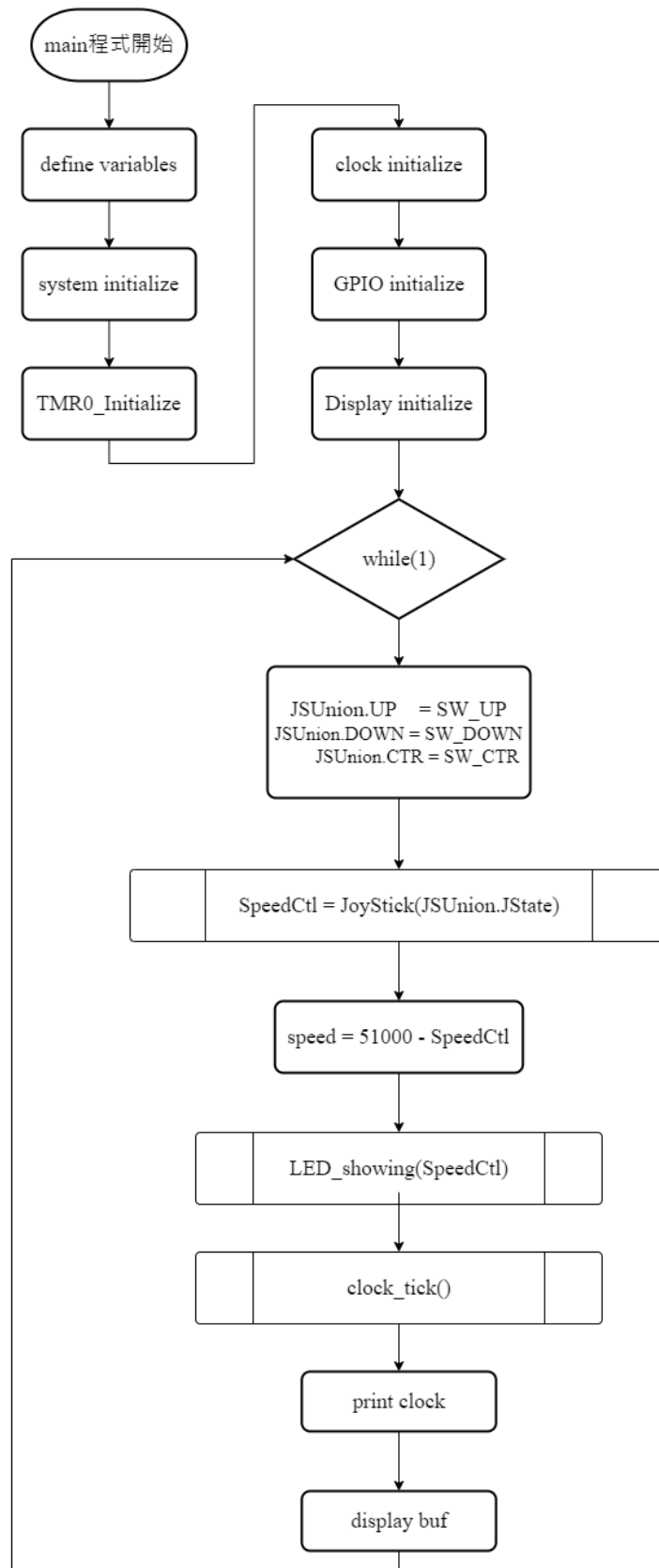
```

/* 定義讓LED閃爍的函數 */
void LED_showing(uint32_t SpeedCtl)
{
    static uint32_t old_timecount = 0;
    static unsigned char f=0x01;
    if ((timecount - old_timecount) >= SpeedCtl)
    {
        f ^= 0x01; // 讓f在0跟1之間交替
        PH6 = (f)?1:0; // 利用f來控制PH6是否發光
        PH7 = (f)?0:1; // 利用f來控制PH6是否發光，要跟PH6交替
        old_timecount = timecount; // 更新old_timecount
    }
}

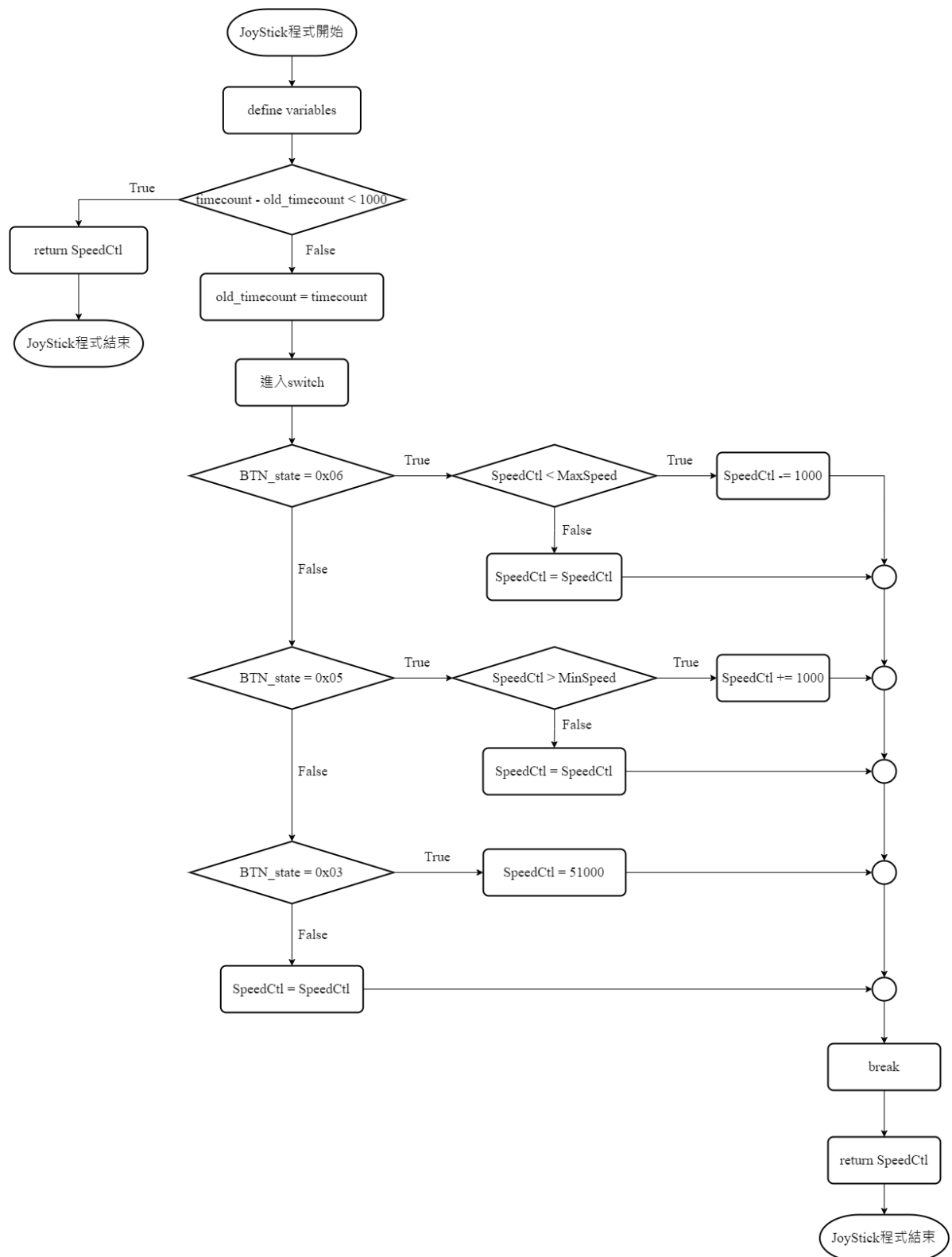
```

二、 流程圖：

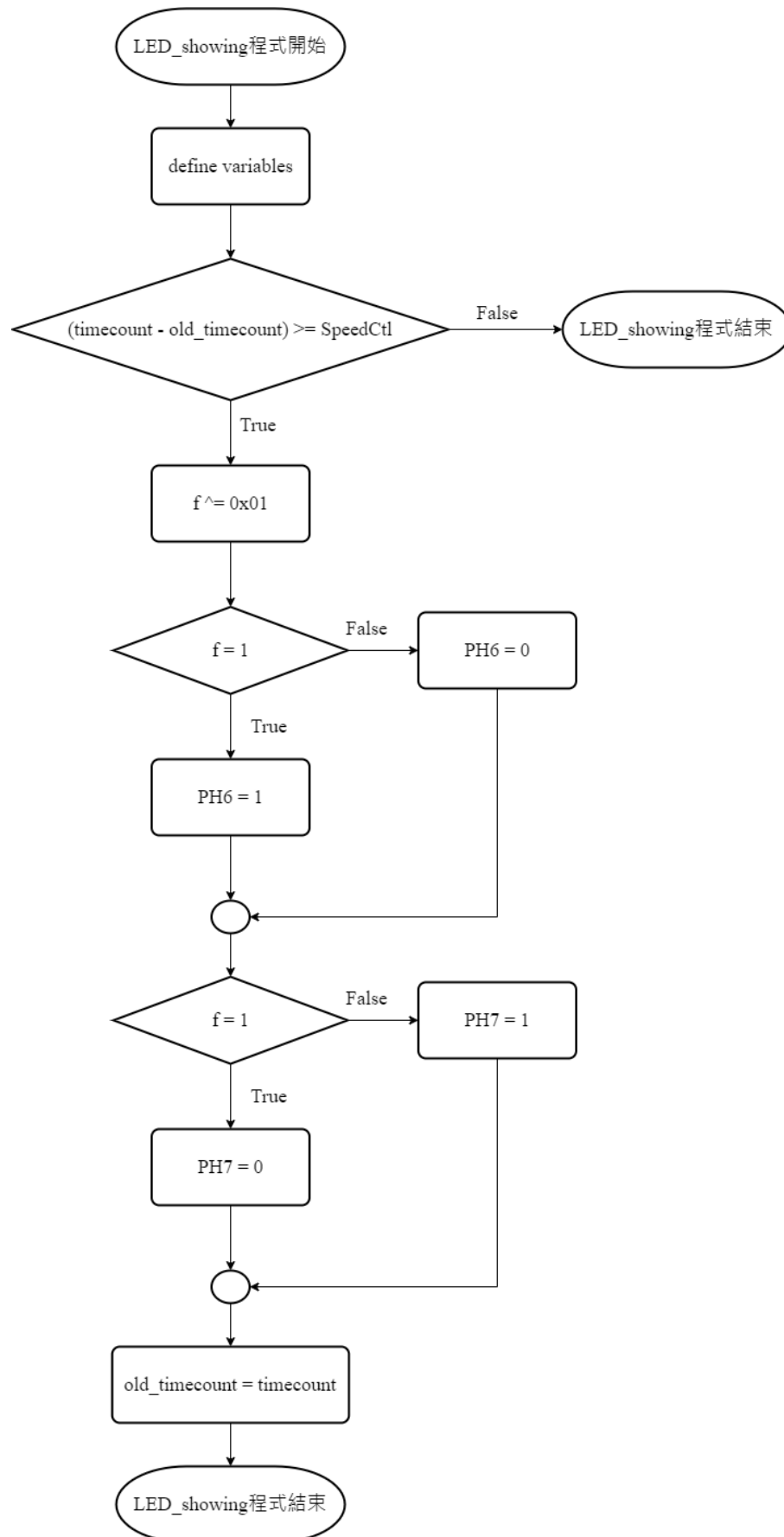
- main



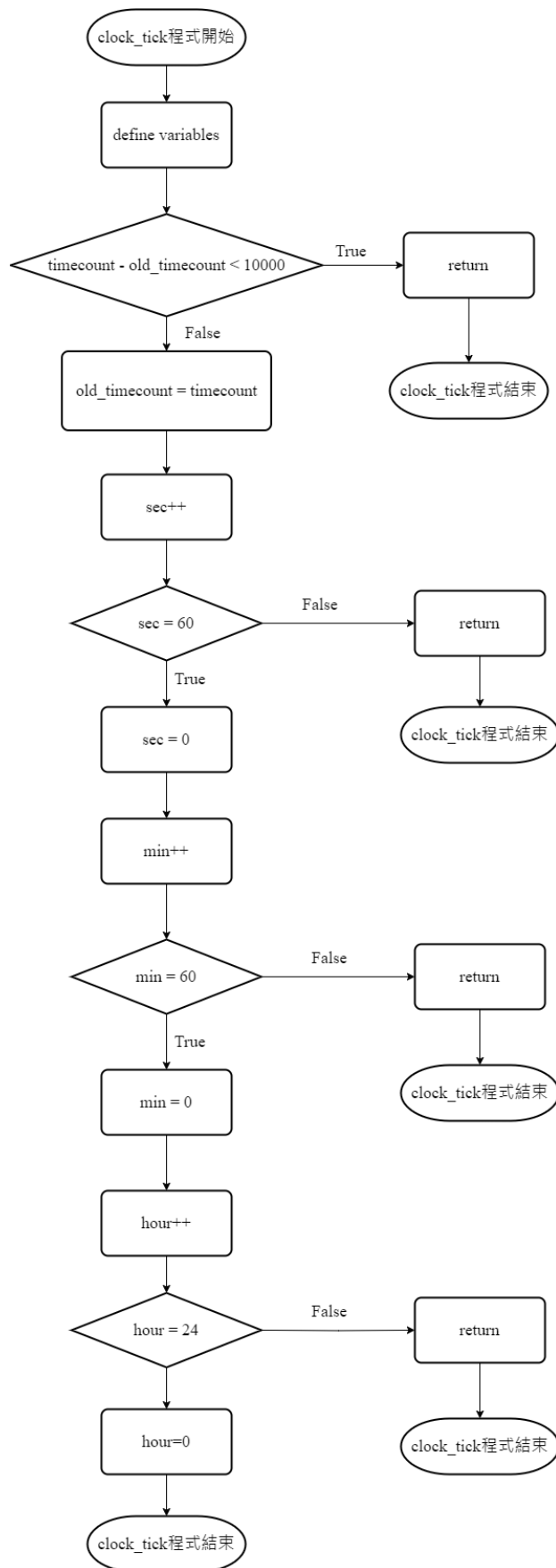
● JoyStick



- LED_showing



- Clock_tick



三、心得：

這次實驗遇到最大的問題就是對於 C 語言很不熟悉。因為不太懂 C 語言的語法的關係，讀 code 讀得很辛苦，debug 也比寫其他程式語言還要辛苦。還好有助教的講解，才讓我對於 C 語言的語法以及這次實驗的內容有了一定程度的了解。在這次實驗中，除了熟悉建專案、M47JIDAE 的使用之外，我還學習到 C 語言的流程控制，特別是在畫流程圖的時候，對於這次實驗的 code 又多了一些了解。雖然說實驗算是半知半解的狀態下完成的，但是看到成果完成後還是很開心的，也透過回家寫結報的過程將實驗內容再次複習，最後也算是學會了這次實驗的 code 與內容。