



國立台灣科技大學  
電機工程系

---

# 微算機應用實習

(EE4801702)

HW1

班級： 四電機二乙

學號： B11030010

指導老師： 王乃堅

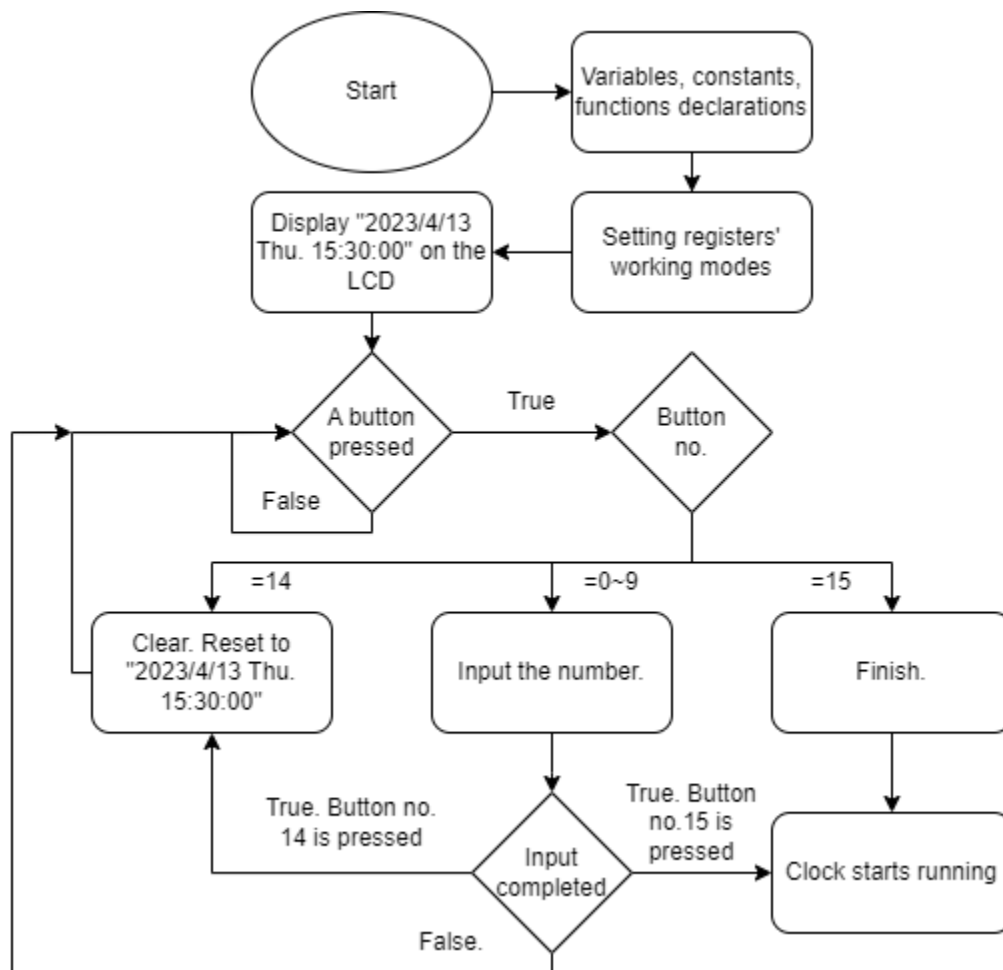
姓名： 陳奕侑

## 一. 目的&原理

- 顯示年月日與時間於 LCD 上

顯示 2023/4/13 Thu. 15:30:00 在 LCD 上，使用者能透過按鈕選擇起始秒數，並能清除重選，選定後即可開始當作時鐘使用。按鈕的輸入使用到 Port A，LCD 的輸出使用一些包好的函式（包含 OUTW、INW 巨集）。

## 二. 程式流程圖



## 三. 程式碼

- 變數宣告、初始化與函式宣告

```

int setsec = 2;
int nddigit = 0;
int final = 0;

const char year = 0x23; //BCD
const char month = 4;    //Decimal
const char week = 4;     //Decimal
const char day = 0x13;   //BCD
const char hour = 0x15;  //BCD
const char min = 0x30;   //BCD
const char sec = 0x00;   //BCD

void SendandSet(int);

```

變數 setsec 記錄目前能夠進行調整的秒數位數，nndigit 紀錄秒數十位的部分，final 則紀錄完整的秒數（若使用者輸入了兩位秒數）。const char 的部分則為初始設定的日期時間，SendandSet 為一函式，本節最後會介紹。

- 主程式 main 開始與設置

```

int main()
{
    OS_PowerOnDriverInitial();

    GPIO_PTA_FS = 0x0000;

    GPIO_PTA_DIR = 0x0FF0;
    GPIO_PTA_CFG = 0x0000;
    GPIO_PTA_PADINSEL = 0x0000;

    // Initial RTC to 2023/04/14 Thu. 15:30:00
    InitialLCD(); //Initial LCD Module
    InitialRTC(year, month, week, day, hour, min, sec); //Initial RTC Module
    DisplayTime(); //Display time on LCD
}

```

剛開始是按鍵部分（PortA）暫存器的設置。InitialLCD 啟用 LCD 模組後，將欲顯示之日期與時間傳入 InitialRTC 函式進行設置，再使用 DisplayTime 顯示內容於 LCD 上，達到「開機後顯示 2023/4/13 Thu. 15:30:00」的目的。

- 檢查是否有按鈕被按下

```

unsigned int tmp = 0;
unsigned int col;
unsigned int key;
unsigned int keep = 0xFF;

while(1){
    key = 0xFF;
    for (col = 0; col < 4; col++){
        GPIO_PTA_BS = 0x000F;
        GPIO_PTA_BR = 0x0000 | (1 << col);
        tmp = ((~GPIO_PTA_PADIN) & 0xFF0) >> 4;
        if (tmp > 0){
            if (tmp & 0x1)
                key = 0*4 + col;
            else if (tmp & 0x2)
                key = 1*4 + col;
            else if (tmp & 0x4)
                key = 2*4 + col;
            else if (tmp & 0x8)
                key = 3*4 + col;
        }
    }

    if(keep != key){
        keep = key;
    }
    else{
        key = 0xFF;
    }
}

```

按鈕若沒被按下，key 值會為 0xFF，任一鍵按下後，key 值儲存該按鈕的編號 (SW2~SW17 我們當作 0~15)。即使按下未放開，處理器也會當作只有按下一次。

- 若有按鈕被按下 (key 不再為 0xFF) 則執行

```

if (key != 0xFF){
    if(key == 14){          //SW16 - Clear to 00
        setsec = 2;
        nddigit = 0;
        final = 0;
        InitialRTC(year, month, week, day, hour, min, sec); //Initial RTC Module
        DisplayTime(); //Display time on LCD
    }
    else if(key == 15){      //SW17 - Confirm and start counting
        if(setsec == 2){
            InitialRTC(year, month, week, day, hour, min, sec); //Initial RTC Module
        }
        else if(setsec == 1){
            InitialRTC(year, month, week, day, hour, min, nddigit); //Initial RTC Module
        }
        else if(setsec == 0){
            InitialRTC(year, month, week, day, hour, min, final); //Initial RTC Module
        }
        break;              //Leave this while loop
    }
    else if((key >= 0) && (key <= 9)){ //SW 2~11 Add
        SendandSet(key);
    }
}

```

按下任一鍵後，程式會依照該按鍵編號進行分類。key 值為 14、15、0~9 時，則重新設置時間（回歸 2023/4/13 Thu. 15:30:00）、啟用時鐘功能、調整秒數（第一次選擇秒數十位數，第二次選擇秒數個位數）。重設時間須考慮到回歸 setsec、nndigit 與 final，以免錯誤產生。啟用時鐘功能則須依照輸入情況給予不同的更新。調整秒數的部分都在 SendandSet 中完成。

- 執行時鐘功能

```

while(1)
    DisplayTime(); //Display time on LCD

return 0;

```

若使用者按下確認（key 值為 15），程式進入此無窮迴圈，時鐘功能啟用。

- 函式 SendandSet

```

void SendandSet(int secnum){
    if(setsec == 2){          //Setting second digit
        if(secnum >= 6){
            return;
        }
        nddigit = secnum << 4;
        InitialRTC(year, month, week, day, hour, min, nddigit); //Initial RTC Module
        DisplayTime();
        setsec--;
    }
    else if(setsec == 1){      //Setting first digit
        final = nddigit + secnum;
        InitialRTC(year, month, week, day, hour, min, final); //Initial RTC Module
        DisplayTime();
        setsec--;
    }
    else{                      //No action
        return;
    }
}

```

在使用者按下 0~9 後，程式會將按下之按鍵編號傳送至此函式，此函式會先透過 setsec 了解使用者欲輸入之秒數位數（十位、個位或已輸入完成）。欲輸入至秒數十位的數字必須小於 6，否則函式執行終止，在正常情況下，應將輸入數字左移四位（乘以 16，因秒數的部分是使用 BCD 格式編碼）並以 InitialRTC 更新欲顯示內容，輸入成功後 setsec 會減一。下一次輸入時，會進入到 else if 區塊，進行秒數個位的更新，final 會記錄秒數十位及個位，紀錄完後便更新欲顯示內容，再將 setsec 減一，如此一來，使用者若再按下一次 0~9，則不會有任何效果。

#### 四. 程式執行結果

