

5-2 功能鍵程式設計

A. 題目

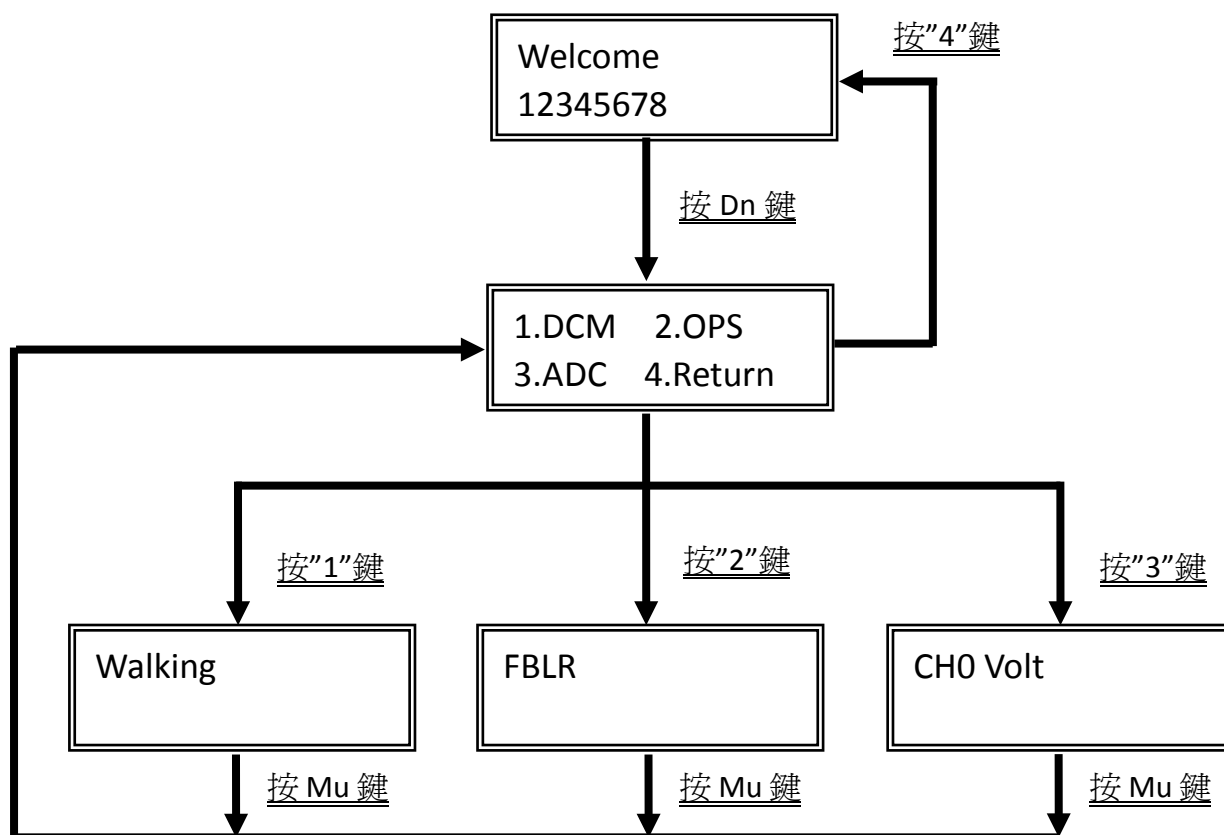
在上一小節已完成初始畫面和按下” Dn” 鍵後LCM出現的主功能表選項，在此節將進一步完成按鍵選單之設計。

依考題要求，當操作者在LCM出現功能選單的狀態下，假若按下”4”數字鍵，系統就返回初始狀態，LCM出現歡迎畫面，假如選擇了編號1至3項當中的一個之後，系統應該依照下列的說明和要求進行動作，各項動作如下：

1. 按下” 1” 鍵即進入直流馬達控制(DCM)的設定功能，
2. 按下” 2” 鍵即進入光學感應器(OPS)的設定功能，
3. 按下” 3” 鍵即進入類比數位轉換(ADC)的功能，
4. 按下” 4” 鍵即系統就返回初始狀態；

各按鍵的動作要求與顯示畫面如程式流程圖說明。

程式流程圖：



B. 功能表設計程式說明：

步驟一：複製以下幾行到主程式結尾(JMP \$)之前

```
call    scan_key
mov     a, keynum
cjne   a, #0ah, loop      ;判斷是否為 Menu 鍵
```

步驟二：加上按鍵判斷程式（紅色字體），

此程式的功能為：如果按"1"鍵即進入 DCM 直流馬達設定功能，按"MU"鍵返回主功能

```
Dn_scan:    call    scan_key
             mov     a, keynum
             cjne    a, #1, Dn1      ;判斷是否為"1"鍵
             call    DCM_menu        ;執行直流馬達控制(DCM)的設定功能
             jmp     Dn_menu         ;跳回顯示功能表畫面

Dn1:
             jmp     Dn_scan         ;重新掃描按鍵
```

步驟三：設計 DCM_menu 選單顯示副程式

方法：複製原程式顯示主功能表畫面程式到步驟二程式的下方

```
sho_main:
    mov     dptr, #main_scr1
    call    sho_lcm1
    mov     dptr, #main_scr2
    call    sho_lcm2

loop:
    call    scan_key
    mov     a, keynum
    cjne   a, #0ah, loop      ;判斷是否為 Menu 鍵
```

以上程式功能為顯示第一個字串，按 Menu 鍵離開，修改 sho_main 名稱為 DCM_menu，其餘內容照以下紅字部分修改。

```
DCM_menu:
    mov     dptr, #DCM_scr1
    call    sho_lcm1
    mov     dptr, #DCM_scr2
    call    sho_lcm2
```

DCM_scan:

```
call    scan_key
mov     a, keynum
cjne    a, #0ah, DCM_scan    ;判斷是否為 Menu 鍵
ret
```

按"MU"鍵離開副程式

步驟四：修改顯示字串宣告，使其能顯示直流馬達控制(DCM)功能的畫面。

```
main_scr1:  db    "Welcome", 0
main_scr2:  db    "12345678", 0
Dn_scr1:    db    "1.DCM  2.OPS", 0
Dn_scr2:    db    "3.ADC  4.Return", 0
DCM_scr1:   db    "Walking", 0
DCM_scr2:   db    "", 0
```

;字體大小寫請與題目要求一致

可複製前 2 行再修改內容即可

到此按下"1"鍵即進入直流馬達控制(DCM)設定功能畫面的設計已經完成，各位同學可以先將程式組譯，測試按"1"鍵是否可顯示"Walking"畫面，按"Mu"回到主畫面，如果可以就可以擴充到其它3個鍵。

步驟五：複製按"1"鍵判斷程式3次

```
Dn_scan:    call    scan_key
            mov     a, keynum
            cjne    a, #1, s1    ;判斷是否為"1"鍵
            call    DCM_menu    ;執行直流馬達控制(DCM)的設定功能
            jmp     Dn_menu    ;跳回顯示功能表畫面

Dn1:
    cjne    a, #1, Dn1    ;判斷是否為"1"鍵
    call    DCM_menu    ;執行直流馬達控制(DCM)的設定功能
    jmp     Dn_menu    ;跳回顯示功能表畫面

Dn1:
    cjne    a, #1, Dn1    ;判斷是否為"1"鍵
    call    DCM_menu    ;執行直流馬達控制(DCM)的設定功能
    jmp     Dn_menu    ;跳回顯示功能表畫面

Dn1:
    cjne    a, #1, Dn1    ;判斷是否為"1"鍵
    call    DCM_menu    ;執行直流馬達控制(DCM)的設定功能
```

```

        jmp      Dn_menu      ;跳回顯示功能表畫面
Dn1:
        jmp      Dn_scan      ;重新掃描按鍵

```

步驟六：修改判斷程式內容，注意按鍵"4"與其他鍵不同，不需要再呼叫任何副程式，直接跳到最開始顯示歡迎畫面位置

```

Dn_scan:    call      scan_key
            mov       a, keynum
            cjne      a, #1, Dn1      ;判斷是否為"1"鍵
            call      DCM_menu        ;執行直流馬達控制(DCM)的設定功能
            jmp       Dn_menu        ;跳回顯示功能表畫面
Dn1:
            cjne      a, #2, Dn2      ;判斷是否為"2"鍵
            call      OPS_menu        ;執行光學感應器(OPS)的設定功能
            jmp       Dn_menu        ;跳回顯示功能表畫面
Dn2:
            cjne      a, #3, Dn3      ;判斷是否為"3"鍵
            call      ADC_menu        ;執行類比數位轉換(ADC)的設定功能
            jmp       Dn_menu        ;跳回顯示功能表畫面
Dn3:
            cjne      a, #4, Dn4      ;判斷是否為"4"鍵
            jmp       sho_main        ;直接跳回顯示歡迎畫面(注意跳躍的地方與前3個不同)，
                                     ;而且不需要再另外設計顯示畫面副程式
Dn4:
            jmp       Dn_scan        ;重新掃描按鍵

```

程式修改到這裡，可以先嘗試組譯一下，可發現程式中會出現 2 個錯誤，都是 UNDEFINED SYMBOL(未定義符號)，因為要呼叫的副程式在程式中並不存在。

在錯誤位置點兩下，可發現位置是以下 2 行

```

        call      OPS_menu
        call      ADC_menu

```

因為這 2 個副程式還沒寫，組譯程式找不到 OPS_menu 和 ADC_menu。

步驟五：修改顯示字串宣告，在此先將 Walking、FBLR 和 CH0 Volt 3 個功能要顯示的畫面準備好。

```
main_scr1:    db    "Welcome",0
main_scr2:    db    "12345678",0
Dn_scr1:      db    "1.DCM  2.OPPS",0
Dn_scr2:      db    "3.ADC  4.Return",0
DCM_scr1:     db    "Walking",0      ;字體大小寫請與題目要求一致
DCM_scr2:     db    "",0
DCM_scr1:     db    "FBLR",0
DCM_scr2:     db    "",0
DCM_scr1:     db    "CH0 Volt",0
DCM_scr2:     db    "",0
```

可先將前 2 行複製 2 次即可

步驟七：複製 DCM_menu 選單顯示副程式 2 次（注意：因為按 4 時直接跳回去顯示歡迎畫面處，不需要再為它另外設計顯示畫面）

```
DCM_menu:
    mov     dptr,#DCM_scr1
    call    sho_lcm1
    mov     dptr,#DCM_scr2
    call    sho_lcm2
DCM_scan:
    call    scan_key
    mov     a, keynum
    cjne    a, #0ah, DCM_scan      ;判斷是否為 Menu 鍵
    ret
```

```
DCM_menu:
    mov     dptr,#DCM_scr1
    call    sho_lcm1
    mov     dptr,#DCM_scr2
    call    sho_lcm2
DCM_scan:
    call    scan_key
    mov     a, keynum
    cjne    a, #0ah, DCM_scan      ;判斷是否為 Menu 鍵
    ret
```

```

DCM_menu:
    mov     dptr,#DCM_scr1
    call    sho_lcm1
    mov     dptr,#DCM_scr2
    call    sho_lcm2
DCM_scan:
    call    scan_key
    mov     a, keynum
    cjne    a, #0ah, DCM_scan    ;判斷是否為 Menu 鍵
    ret

```

步驟八：修改成 OPS_menu 和 ADC_menu 選單顯示副程式

```

DCM_menu:
    mov     dptr,#DCM_scr1
    call    sho_lcm1
    mov     dptr,#DCM_scr2
    call    sho_lcm2
DCM_scan:
    call    scan_key
    mov     a, keynum
    cjne    a, #0ah, DCM_scan    ;判斷是否為 Menu 鍵
    ret

```

```

OPS_menu:
    mov     dptr,#OPS_scr1
    call    sho_lcm1
    mov     dptr,#OPS_scr2
    call    sho_lcm2
OPS_scan:
    call    scan_key
    mov     a, keynum
    cjne    a, #0ah, OPS_scan    ;判斷是否為 Menu 鍵
    ret

```

```

ADC_menu:

```

```
        mov     dptr,#ADC_scr1
        call    sho_lcm1
        mov     dptr,#ADC_scr2
        call    sho_lcm2
ADC_scan:
        call    scan_key
        mov     a, keynum
        cjne    a, #0ah, ADC_scan      ;判斷是否為 Menu 鍵
        ret
```

到此基本選單畫面已經完成