五.排七段顯示器

下面的排七段顯示器實驗我們透過KeilC組譯我們完成編輯的程式碼，並且

透過燒錄程式(progisp)，將程式燒錄到單晶片。

A.完成硬體接線

B.建立新的專案並加入新的原始碼檔

後鍵入原始碼

C. 撰寫程式碼後組譯之

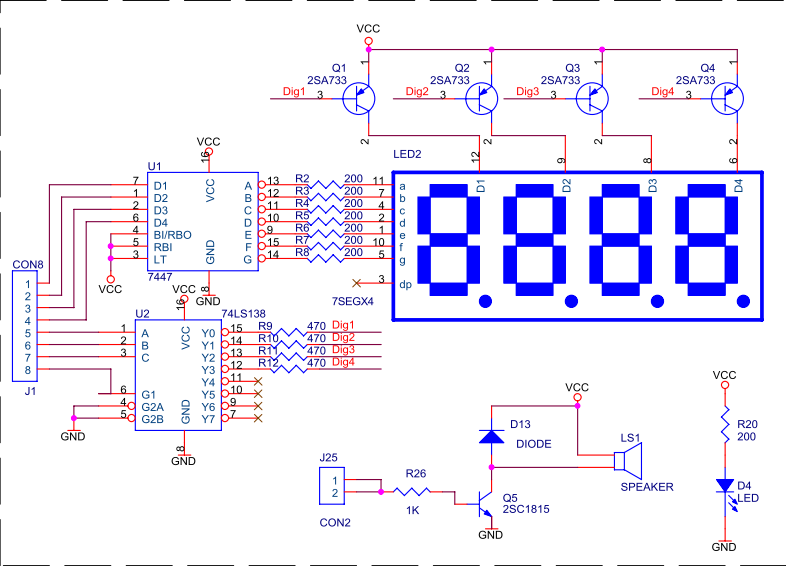
D. 將HEX檔燒錄至晶片

A.完成硬體接線

這次的實驗，將從P2埠輸出到排七段顯示器

第一步， 我們要找出排七段顯示器的跳線位置

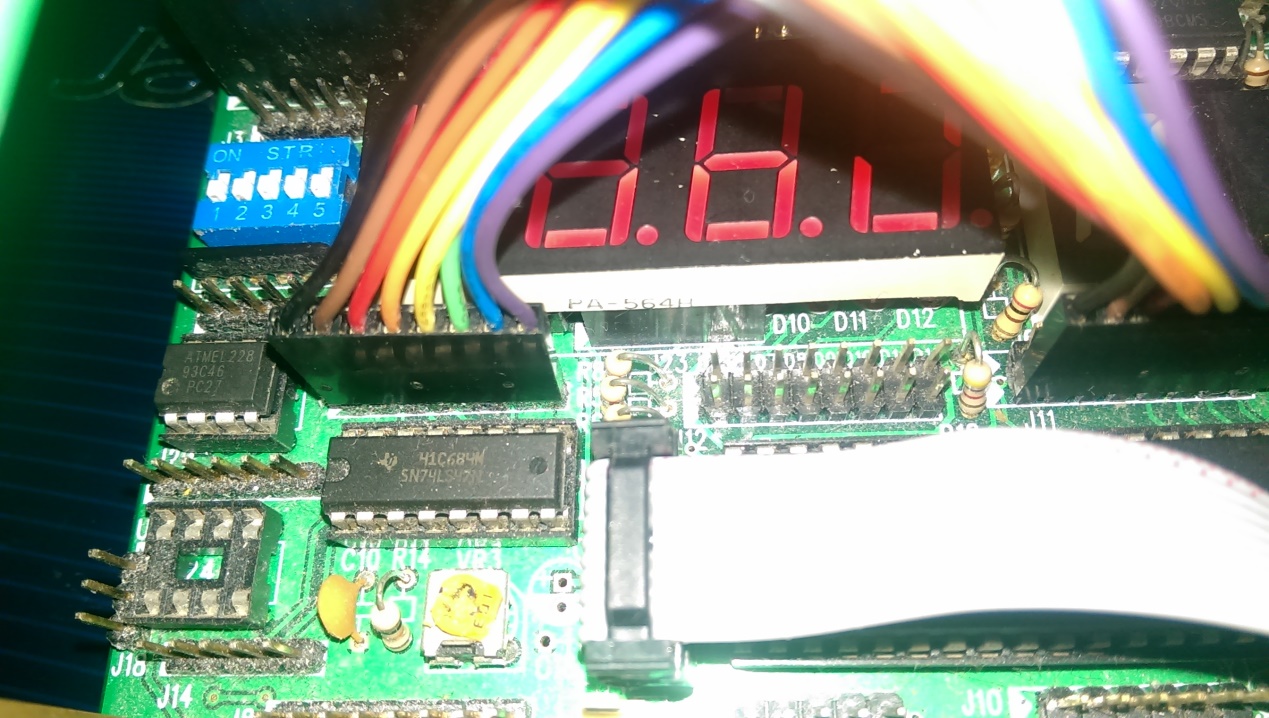
從下圖，我們可以的得知，排七段顯示器跳線位置位於J1



第二步，根據之前的實驗我們已經得知P2埠跳線位置位於J11

所以，我們將J1(排七段顯示器)接到J11(P2埠)

特別注意，黑色的線都在左邊

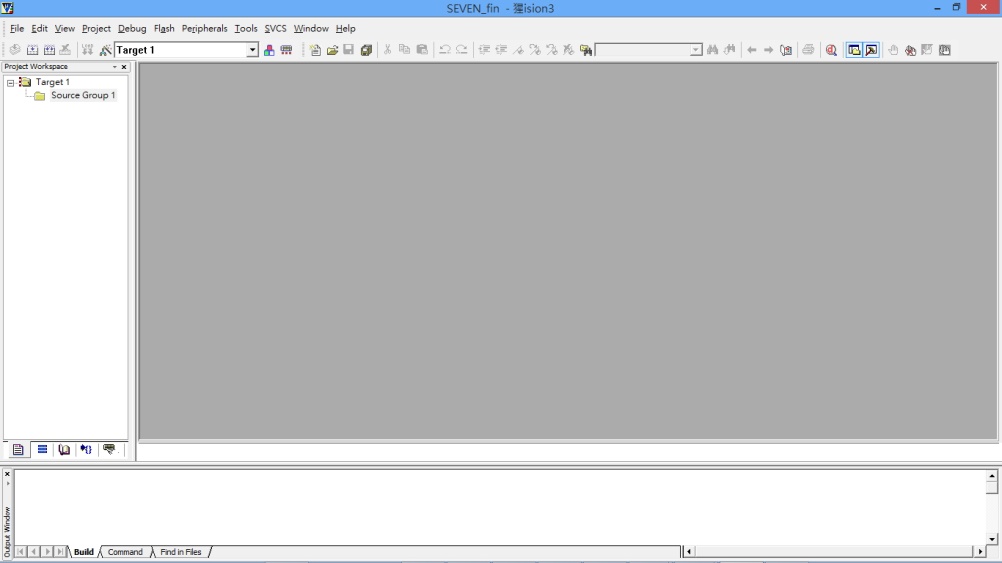


B. 建立新的專案並加入新的原始碼檔後鍵入原始碼

第一步，建立新的專案

這次我們將專案放置位置，指定放置於桌面

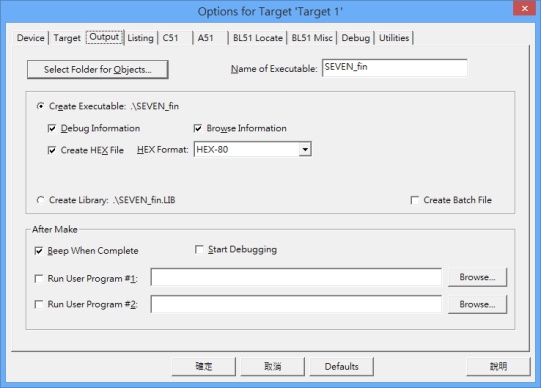
由於是指播開關與七段顯示器的實驗。所以，我們設專案名稱為More\_Seven



第二步，設定專案，使組譯後將產生HEX檔

進到Flash Configure Flash Tools Output

將Create HEX files打勾

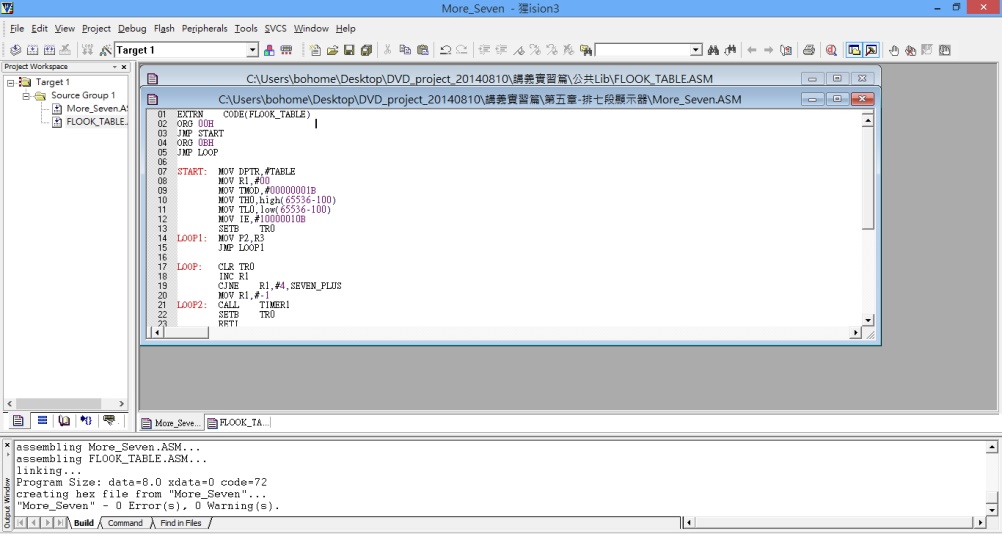


第三步，加入新的原始碼檔

這一步，我們要建立二個個原始檔

一個是主程式，設More\_Seven.ASM名稱

另一個是副程式，設FLOOK\_TABLE.ASM名稱



C. 撰寫程式碼後組譯之

首先，先將下列的程式碼鍵入到主程式(More\_Seven.ASM)

**EXTRN CODE(FLOOK\_TABLE)**

**ORG 00H**

**JMP START**

**ORG 0BH**

**JMP TIMER\_ISR**

**//主程式**

**START: MOV DPTR,#TABLE**

**MOV R1,#00**

**MOV TMOD,#00000001B**

**MOV TH0,high(65536-100)**

**MOV TL0,low(65536-100)**

**MOV IE,#10000010B**

**SETB TR0**

**//將R3的值循環輸出至排七段顯示器**

**LOOP1: MOV P2,R3**

**JMP LOOP1**

**//中斷服務常式(ISR)**

**TIMER\_ISR: CLR TR0 //停止計時**

**INC R1**

**CJNE R1,#4,SEVEN\_CODE //若查表未查到最後一個資料，則跳到SEVEN\_CODE副程式**

**MOV R1,#-1 ////若查表已查到最後一個，則將查表計數器設為-1**

**JMP Initial\_Timer1**

**// 查表並將值放入R3，以供主程式顯示使用**

**SEVEN\_CODE: CALL FLOOK\_TABLE**

**MOV A,R2**

**MOV R3,A**

**Initial\_Timer1: MOV TH0,high(65536-100)**

**MOV TL0,low(65536-100)**

**SETB TR0**

**RETI**

**TABLE: DB 081H,092H,0A3H,0B4H**

**END**

**接下來，將以下的副程式(**FLOOK\_TABLE.ASM**)，鍵入下列的程式碼**

**;----------------------------------------------- --------------------------------**

**;順向查表 : index -> code**

**;輸入 : index(R1),table\_name(DPTR)**

**;輸出 : code(R2)**

**;使用 : R1,R2**

**;查表在R1,結果放R2**

**;----------------------------------------------- --------------------------------**

**PUBLIC FLOOK\_TABLE ;模組名稱**

**ROM SEGMENT CODE ;共享ROM空間**

**RSEG ROM ;選取可重置的 ROM空間**

**FLOOK\_TABLE: ;模組名稱**

**MOV A,R1 ;將R1位置的值放在A**

**MOVC A,@A+DPTR ;將表格內第A(R1)個的數字放於A**

**MOV R2,A ;將A的值放在R2**

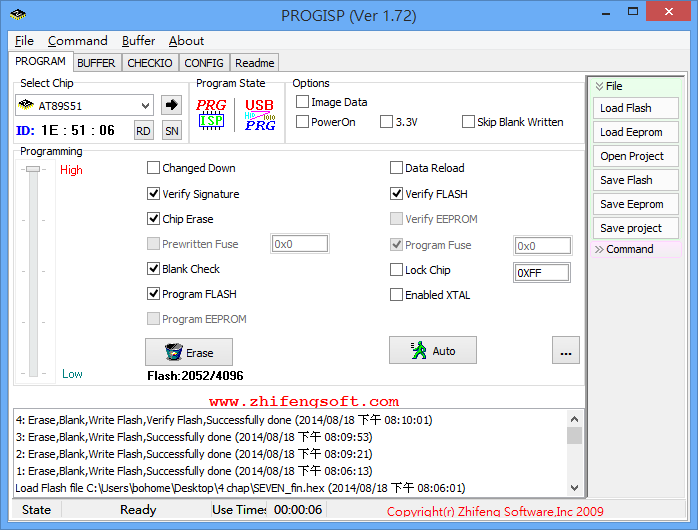
**RET ;從模組返回**

**END ;程式結束**

D. 將HEX檔燒錄至晶片

第二步，燒錄器根據下圖設定後，按下下圖中的AUTO鍵

在這裡，我們將速度調到High(高速)。如果，跳出錯誤訊息，可以將速度調到中間，再試一次。



做到這裡，恭喜你了，排七段顯示器實驗已經完成了

**觀念提醒:**

**A.程式架構:**

透過中斷，將排七段顯示器上的四個七段顯示器進行掃描

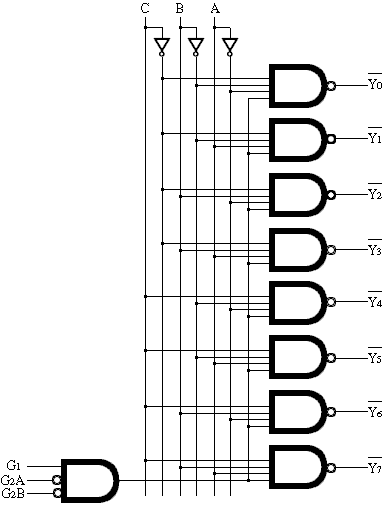
**B. 排七段顯示器是由四顆七段顯示器並排組成，為了減少接線**

**將34條PIN調整為12條，其中顯示(8條)，選擇(4條)**

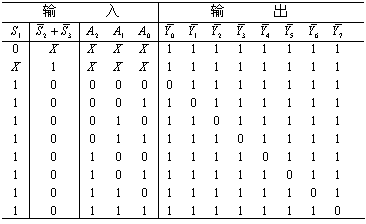
**1. 排七段顯示器 選擇部分**

透過74138 (3對8解碼器):

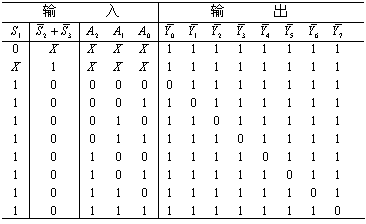
將3個輸入解碼成8個輸出，這裡我們將其特性用來選擇欲點亮之七段顯示器，下圖為74138的邏輯電路圖



下圖為74138的真值表



由於這次排七段顯示器實驗只用到了Y0~Y3，所以我們將真值表簡化成下圖



**C B A**

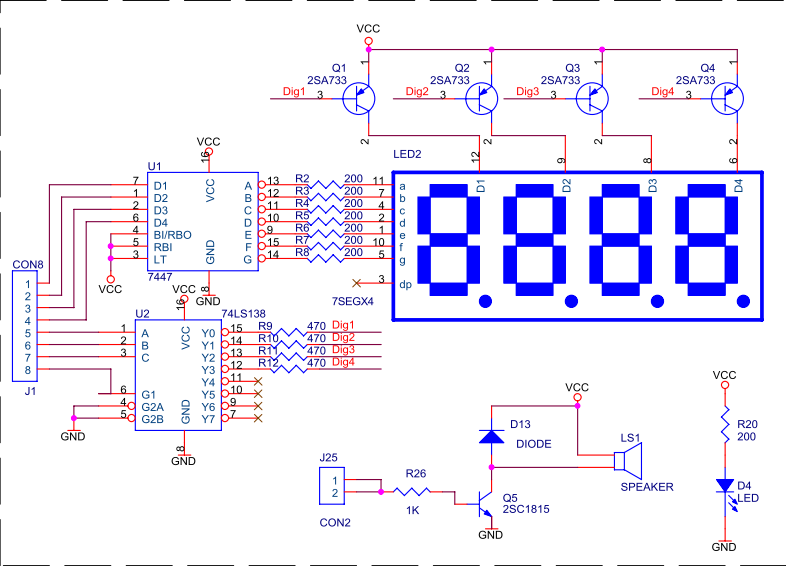
例如，點亮左邊第一顆七段顯示器，C,B,A依序輸入0,0,0即可。

**2. 排七段顯示器 顯示部分**

透過7447解碼器:

輸入BCD值，將其值經由編碼，直接顯示於共陽七段顯示器上

例如: 輸入0001至7447解碼器，將輸出F9H到七段顯示器



**C. 排七段顯示器編碼**

經由上述提到的7447解碼器與74138解碼器，可以將排七段顯示器進行選擇與顯示。

下圖為這次實驗的腳位對應圖

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 74138解碼器 | | | | 7447解碼器 | | | |
| P2.7 | P2.6 | P2.5 | P2.4 | P2.3 | P2.2 | P2.1 | P2.0 |
| G1(致能) | C | B | A | D4(最高位) | D3 | D2 | D1(最低位) |

PS: G1(致能)決定74138解碼器是否工作

G1=1代表工作，G1=0代表停止工作

例如，我們要選擇左邊第一顆七段顯示器，顯示”1”，這個數字。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| G1(致能) | C | B | A | D4(最高位) | D3 | D2 | D1(最低位) |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 |

**Q&A**

Q1. 如何一次將四個七段顯示器都點亮

ANS1.使用超過30HZ的頻率掃描四個七段顯示器，人眼會

因為視覺暫留，造成四個七段顯示器同時點亮的錯覺