

高雄縣高英高級工商職業學校  
Kao Ying Industrial Commercial Vocational High School

專題製作報告



搖 搖 棒

學生姓名：張 家 翔

張 啟 強

顏 宏 騰

吳 惟 倫

指導老師：葉 忠 賢 老師

中 華 民 國 102 年 05 月



## 誌 謝

本專題能順利完成，首先由衷的感謝專題指導老師葉忠賢老師，感謝她在專題寫作期間的細心教誨與指導，令學生終身受益匪淺。專題製作中遇到一些困難，指導老師認真地、有耐心地與組員多次討論，才能順利的將專題完成，對於葉忠賢老師，我們有無限的感激。對我們而言，不僅是教學上的良師，也是益友，生活中遇到的各種難題，都可找葉忠賢老師商量，謝謝葉忠賢老師這些日子的指導。

也感謝組員們的分工合作及資料收集，在學習期間的互相砥礪及陪伴與彼此加油打氣。最後，也要感謝在專題製作期間曾經幫助過我們的所有老師，正因為大家的同心協力，才能使我們的專題製作更加完整，也能更順利的完成，在此本組特別的感謝，僅致上最高的敬意及謝意。

張家翔、張啟強、顏宏騰、吳惟倫 謹上 2013/05

## 摘要

POV LED 旋轉顯示屏是利用人眼的視覺暫留特性，通過手擺動控制一排 LED 燈，單片機控制它們的閃爍時間和頻率，就可呈現出一幅完整的畫面。本系統是以 89C51 單晶片為核心，16 顆 LED 排列，利用人眼視覺暫留現象而顯示特定圖像、字符或文字信息，並利用按鈕方式控制畫面的切換及相關顯示控制 16 隻 LED 發光管作為畫面每一列的顯示，左右搖晃起到了掃描的作用，人眼的視覺暫留現象使得看到的是一幅完整的畫面。與 LED 點陣的顯示有幾分相仿。我們在以水銀開關去控制 LED 的閃爍來達成文字與圖像的顯現。

**關鍵詞：**單晶片、水銀開關、LED

# 目 錄

誌謝 .....	I
摘要 .....	II
目錄 .....	III
表目錄 .....	IV
圖目錄 .....	V
壹、前言 .....	1
一、製作動機 .....	1
二、製作目的 .....	1
三、製作架構 .....	2
四、製作預期成效 .....	3
貳、理論探討 .....	4
一、電子相關零組件 .....	4
二、微電腦單晶片 .....	11
參、專題製作 .....	16
一、設備及器材 .....	16
二、製作方法與步驟 .....	17
三、專題製作 .....	18
肆、製作成果 .....	21
伍、結論與建議 .....	22
參考文獻 .....	23
附錄一 搖搖棒之程式碼 .....	24
附錄二 程式工具 Keil C 介紹 .....	34
附錄三 中文字型編譯程式 .....	37

## 表目錄

表 2-1-1 中斷進入點位址 .....	4
表 2-1-2 8051 內部控制 .....	6
表 2-1-3 TMOD 計時控制 .....	7
表 2-1-4 TCON 計時控制 .....	8
表 2-1-5 串列埠控制 .....	9
表 2-1-1 記憶體分佈 .....	10
表 2-1-1 專題製作使用儀器（軟體）設備一覽表 .....	16
表 2-1-1 專題製作計畫書 .....	18
表 2-1-1 搖搖棒材料表 .....	20
表附錄 變數的資料型態 .....	39

## 圖目錄

圖 1-1-1 系統架構圖 .....	3
圖 2-1-1 M1/M0 工作模式圖 .....	8
圖 2-1-2 水銀開關的構造 .....	11
圖 2-2-1 8051 內部結構圖 .....	12
圖 2-2-2 8051 接腳圖 .....	13
圖 2-2-3 MOS 元件連接圖 .....	14
圖 2-2-4 8051 基本電路圖 .....	15
圖 2-2-5 8051 記憶體結構示意 .....	15
圖 3-2-1 製作方法與步驟 .....	17
圖 3-3-1 搖搖棒電路圖 .....	19
圖 4-1-1 麵包版電路模擬 .....	21
圖 4-1-2 搖搖棒程式編譯與燒錄 .....	21
圖 4-1-3 PC 板電路確認 .....	21
圖 4-1-4 搖搖棒成品圖 .....	21
圖附錄 1 Keil C 流程圖 .....	35
圖附錄 2 中文字型產生程式 (GetChWdPat) .....	37





# 壹、前言

## 一、製作動機

有感於電子產品是未來市場中最主要的消費之一，如何發展更多、更方便及更好用的新興電子產品，將是一項重要的課題。因此本專題以 POV (POV 即 persistence of vision) LED 即 LED 圓柱型顯示器為主，以簡易之 POVLED 模擬市面上實際之產品，研究並探討新興電子產品之特色與實用的目的。

本專題研究之 LED 圓柱型顯示器乃專業領域學程內容之延伸，包括基本電學、電子電路、微處理機等，也包含其餘非本科系之專業技術，如機械結構之設計。在專題進行的當中，我們透過各部分的完成，學習整合知識、團隊合作、問題解決的方法、自我表達與實務運用的能力，由此獲益良多。

POV LED 視覺暫留顛覆以往 LED 的顯示方式，接觸過的人都深深著迷，這也是我選擇這個題目的動機。而近來在 LED 的蓬勃發展下，我們希望能利用 8051 的便利性與 LED 的特性，能夠將 LED 發揮到最大功用，在此學以致用是我們最大的目的。

## 二、製作目的

從走馬燈到留影盤，再從留影盤到動畫與電影，視覺暫留為人類的文明開啟了另一個新紀元。現今電影蓬勃發展，卡通動畫風靡全球，都是由靜止圖像所造成的動態幻覺，其深深吸引人之處，正是源自於視覺暫留的獨特與魅力。近年來，由於微處理機的普及、LED 蓬勃發展，市面上常常可以看到許多由 LED 快速轉動所產生的圖案，不只吸引了我們的注意，也給人們留下了深刻印象。這些千奇百怪，充滿創意的電子作品，即是 POV LED。有鑑於市面上 POV LED 的蓬勃發展，本專題想研究並探討 POV LED 的製作方法與概念，利用現有的自身技術—LED 應用、程式語言、電路繪製，並配合專題製作課程加以研究。

這個專題的目的是利用視覺暫留，給人們帶來視覺上的美感，並研究如何使一排 LED 旋轉顯示出文字畫面或其他圖案，以製作簡易型的視覺暫留器為主。專題分成實際電路、程式部份及創意部份，需要用到我們之前所學的很多課程的結合，像是電子學、數位邏輯、微處理機...等的知識，也有許多其他課外的新知識，

更讓我們體會遇到困難然後找出解決的精神意義，並把它應用在專題上，藉此機會去實際規劃、製作整個專題，讓我們更了解軟體與硬體的應用與結合，對我們未來的升學考試有很大的幫助，也增加我們許多的實務經驗。

### 三、製作架構

本專題主要分為2大部分，程式系統、電路板系統。我們使用C語言控制89S51單晶片，再利用掃完一行的延遲時間，搭配擺動速度，最後清晰顯示出我們想要的文字。其中，電路板部分我們完全發揮89S51基本電路的功能，利用Port1輸出，使顯示器完整顯示。

系統架構如圖 1-1-1 所示。

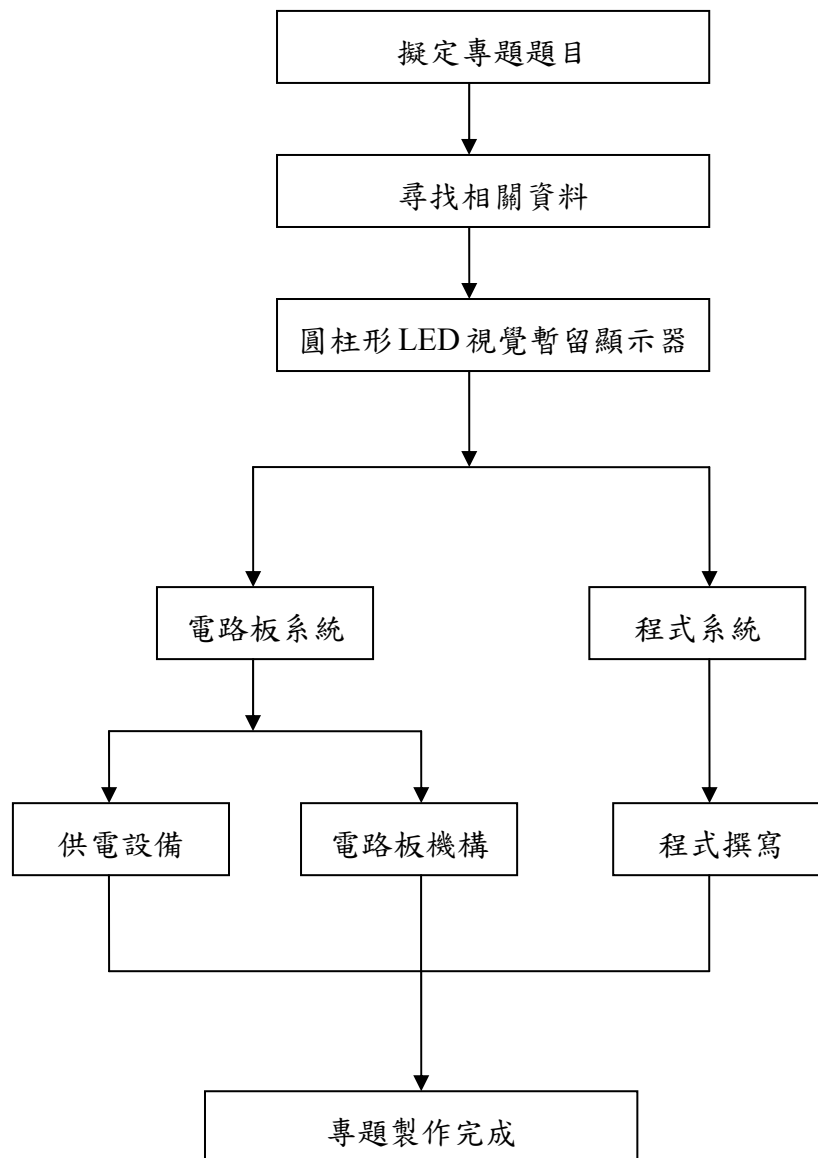


圖 1-1-1 系統架構圖

#### 四、製作預期成效

眼觀看物體時，成像於視網膜上，並由視神經輸入人腦，感覺到物體的成像。但當物體移去時，視神經對物體的印象並不會立即消失，而要延續 0.1 -0.4 秒的時間，人眼的這種性質被稱為「眼睛的視覺暫留」。我們以手搖方式擺動 IC 電路板，而架設在電路板上之 LED 陣列將產生高速擺動，再利用人類之視覺暫留產生圖形。

- (一) 利用 8051 單晶片燒入程式來使 LED 變換不同文字效果。
- (二) 以 16 顆 LED 來呈現移動後產生的影像。
- (三) 安裝一個 RESET 防止作品無法正常顯示的解決辦法。

## 貳、理論探討

### 一、電子相關零組件

#### (一) 程式記憶體(ROM)

程式記憶體主要是存放程式碼，CPU 會自動到程式記憶體提取程式碼，並依照程式內容執行工作，所以 CPU 無法存入或更新程式記憶體的資料。

8051 的程式記憶體可以分成內部(4K)及外部(64K)，利用 EA 接腳來做設定，若 EA 接高電位則表示使用內部程式記憶體，反之，EA 接低電位則表示使用外部記憶體。

程式記憶體用來存放 8051 程式，可使用內部的 4K 位元組 (EA 接腳為高電位)，若使用者在做內部記憶體的讀取時，程式區超 4Kbytes 後，CPU 自動會送出 Low 的訊號至 PSEN 接腳，進而繼續至外部 ROM 提取程式指令。使用外部的 64K 位元組則將 EA 接腳接地。在寫 8051 的程式時，必須知道幾個程式記憶體的特殊位址，這些位址是各種中斷服務程式的進入點，下表列出了各種中斷的進入點位址，其中位址 0000H 是重置(RESET)的進入點，這意思是說，8051 被重置時，從位址 0000H 開始執程式。

表 2-1-1 中斷進入點位址

中斷源	向量位址
RESET	0000h
TNT0	0003h
Timer0	000Bh
INT1	000Bh
Timer1	001Bh
UART	0023h
Timer2	002Bh

## (二)資料記憶體(RAM)

### ◎內部資料記憶體

8051 內部有一塊 256 個 byte 的位址空間，這塊空間是存放資料記憶體(RAM)和特殊功能暫存器(SFR)的地方。這塊記憶體空間雖然只有 256byte，但是 8051 將其中位指教高的 128byte(80H~FFH)採用不同的定址方式而容納了兩組 128byte 的記憶體空間，因此總共的空間為  $128+128+128=384$  byte。

以下三個部分開加以解說：

1. 位址 00H~7FH 的 RAM
2. 位址 80H~FFH 的 RAM
3. 位址 80H~FFH 的 SFR
4. 位址 00H~7FH 的 RAM：

此 128 位元組可以使用直接或間接定址法的方式來存取其內部的資料，以下圖來說明，此位址可以分為以下三部分

- A. 暫存器庫 00H~1FH
- B. 可用位元定址區 20H~2FH
- C. 一般用途空間 30H~7FH

表 2-1-2 8051 內部位址

7FH	一般資料存放區或堆疊區							
2FH	7F	7E	7D	7C	7B	7A	79	78
2EH	77	76	75	74	73	72	71	70
2DH	6F	6E	6D	6C	6B	6A	69	68
2CH	67	66	65	64	63	62	61	60
2BH	5F	5E	5D	5C	5B	5A	59	58
2AH	57	56	55	54	53	52	51	50
29H	4F	4E	4D	4C	4B	4A	49	48
28H	47	46	45	44	43	42	41	40
27H	3F	3E	3D	3C	3B	3A	39	38
26H	37	36	35	34	33	32	31	30
25H	2F	2E	2D	2C	2B	2A	29	28
24H	27	26	25	24	23	22	21	20
23H	1F	1E	1D	1C	1B	1A	19	18
22H	17	16	15	14	13	12	11	10
21H	0F	0E	0D	0C	0B	0A	09	08
20H	07	06	05	04	03	02	01	00
1FH	RB3 ( 8 bytes )							
	RB2 ( 8 bytes )							
	RB1 ( 8 bytes )							
	RB0 ( 8 bytes )							
00H								

2.位址 80H~FFH 的 RAM：只有 8052，8752 和 8032 的內部 RAM 有這 128byte，8031，8051 和 8751 則沒有。這塊 RAM 的內容必須使用間接定址法。

3.位址 80H~FFH 的 SFR：特殊功能暫存器是一塊 128byte 的記憶空間，它是存放 8051 內部的週邊所使用的暫存器的地方，例如 I/O port 的輸出栓鎖器(P0，P1，P2，P3)，計時器的 counter，致能中斷系統的 IE 暫存器等。因為 8051 的週邊設備並不多，因此 SFR 裡 128 個位址空間並未用完，這些目前沒有用到的位址，裡面是空的。SFR 所使用個位址是 80H~FFH，這塊區域與 8051 的較高 128 位元組的 RAM 使用了同一塊記憶空間，8051 採用了不同的指令的定址法來區分這兩塊記憶體，如前面所述，RAM 是使用間接定址法，SFR

是使用直接定址法。在 SFR 裡的各種位元組都有其個別的名稱，在寫程式時，要用到這些位元組，可直接呼叫其名稱，而不需要使用位址。

特殊功能暫存器( Special Function Register , SFR )

□ SFR 中暫存器的功能及用途 ( 僅說明有用到的部分 ) :

(1) 累加器( Accumulator , ACC ) :

累加器又可稱之為 ACC 或 A 暫存器，這是一個使用頻率頗高的一個通用暫存器，而有許多指令是以其為操作對象。

(2) P0、P1、P2、P3 埠暫存器 :

這四個埠暫存器可存放 8051 單晶片的 4 個 I/O 埠的輸出閘鎖(Latch)，主要是存放並保持 I/O 的輸出資料。

(3) TH0~TH2、TL0~TL2 計時器/計數時暫存器 :

這 3 組 16 位元的暫存器是分別用來儲存計時器/計數器的計時/計數值。TH0、TH1、TH2 為高位元組，TL0、TL1、TL2 為低位元組。TH0 及 TL0 對應於計時器/計數器 0，TH1 及 TL1 對應於計時器/計數器 1，TH2 及 TL2 對應於計時器/計數器 2(8052 系列)。

(4) 計時器模式控制(Timer/Counter Mode Control , TMOD)暫存器 :

表 2-1-3 TMOD 計時控制

位元	7	6	5	4	3	2	1	0
TMOD	GATE	C/T	M1	M0	GATE	C/T	M1	M0
	計時器1				計時器0			

GATE:計時器動作閘控位元，當 GATE=1 時，INT0 或 INT1 接腳 為高電位，同時 TCON 中的 TR0 或 TR1 控制位元為 1 時，計時/計數器 0 或 1 才會動作。若 GATE=0，則只要將 TR0 或 TR1 控制位元設為 1，計時/計數器 0 或 1 即可動作。C / T : 做計時器或計數器功能之選擇位元。C/T=1 為計數器，由外部接腳 T0 或 T1 輸入計數脈波。C/T=0 為計時器，由內部系統時脈

提供計時工作脈波。

M1：模式選擇位元 1。

M0：模式選擇位元 0。

M1	M0	工作模式
0	0	0 13-bit 計時器
0	1	1 16-bit 計時器/計數器
1	0	2 8-bit 自動載入
1	1	3 (Timer 0)TL0 為 8 位元計時器/計數器，由標準之計時器 0 之控制位元控制，TH0 為 8 位元計時器，且由計時器 1 控制位元控制。
1	1	3 (Timer 1)計時器/計數器 1 停止。

圖 2-1-1 M1/M0 工作模式圖

(5)計時器控制(Timer Control, TCON)暫存器：

表 2-1-4 TCON 計時控制

位元	7	6	5	4	3	2	1	0
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IR1

TF1(TCON.7)：計時器 1 溢位旗號，當計時溢位時，由硬體設定為 1 在執行過相對的中斷服務常式後則自動清除為 0。

TR1(TCON.6)：計時器 1 啟動控制位元，可以由軟體來設定或清除。

TF0(TCON.5)：計時器 0 溢位旗號，當計時溢位時，由硬體設定為 1，在執行過相對的中斷服務常式後則自動清除為 0。

TR0(TCON.4)：計時器 0 啟動控制位元，可以由軟體來設定或清除。

IE1(TCON.3)：外部中斷 1 動作旗號，當外部中斷被偵測出來時，



硬體自動設定此位元，在執行過中斷服務常式後，則消除為 0。

IT1(TCON.2)：外部中斷 1 動作型態選擇，當 IT1=1 時，中斷型態為負緣觸發，當 IT1=0 時，中斷型態則為低準位觸發。

IE0(TCON.1)：外部中斷 0 動作旗號，當外部中斷被偵測出來時，硬體自動設定此位元，在執行過中斷服務常式後，則消除為 0。

IT0(TCON.0)：外部中斷 0 動作型態選擇，當 IT1=1 時，中斷型態負緣觸發，當 IT1=0 時，中斷型態則為低準位觸發。

(6) 串列埠控制( Serial Port Control )暫存器：

表 2-1-5 串列埠控制

位元	7	6	5	4	3	2	1	0
SCON	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0(SCON.7)：串列傳輸模式選擇，共有 4 種模式。

SM1(SCON.6)：串列傳輸模式選擇，共有 4 種模式。

SM2(SCON.5)：在串列傳輸動作模式 2 或模式 3 時，作多處處機控制功能用。

REN(SCON.4)：串列介面接收位元，當 REN=1 時表示接收致能。

TB8(SCON.3)：在模式 2 或 3 時，所送出的第 9 個資料位元，可以由軟體指令來做控制設定或清除。

RB8(SCON.2)：在模式 2 或 3 時，所接收到的第 9 個資料位元，存放在此位元中。

TI(SCON.1)：串列資料傳送中斷旗號，在工作模式 0 時，送出 8 個資料位元後，TI 設為 1，而在其他模式時，在送出停止位元時，TI 也會被設為 1；此位元必須由軟體來清除。

RI(SCON.0)：串列資料接收中斷旗號，在工作模式 0 時，收到第 8 個串列輸入資料位元後，RI 會設為 1，在其他模式時，收到停止位元的一半時，硬體會自動將此位元設為 1。此位元必須由軟體來清除。

(7) 串列資料緩衝( Serial Data Buffer, SBUF )暫存器：8051 單晶片的串列埠是全雙工的，故實際上 SBUF 暫存器分開為兩個不同的暫存器，一個是當作 UART 傳送資料的緩衝區，另一個是當作

UART 接收資料的緩衝區。若將資料寫到 SBUF 時，就會將資料放入傳送緩衝區，UART 就會將這個資料轉成串列資料透過 TXD 傳出去。若去讀 SBUF，就會讀到接收緩衝區的資料。

下表為 SFR 的記憶體分佈圖：

表 2-1-6 記憶體分佈

F8								FF
F0	B							F7
E8								EF
E0	ACC							E7
D8								DF
D0	PSW							D7
C8								CF
C0								C7
B8	IP							BF
B0	P3							B7
A8	IE							AF
A0	P2							A7
98	SCON	SBUF						9F
90	P1							97
88	TCON	TMOD	TLO	TL1	TH0	TH1		8F
80	P0	SP	DPL	DPH			PCON	87

此記憶體位元組可做位元定址。

### (三) 水銀開關

水銀開關，又稱傾側開關，是電路開關的一種，以一接著電極的小巧容器儲存著一小滴水銀，容器中多數注入惰性氣體或直接真空。

#### (1) 構造

在一密閉的真空玻璃容器內，封入一對電極及水銀，利用水銀的良好導電性，當開關傾斜時，水銀介於兩電極間使接點導通，裡面真空並加上隋氣是為了不使電極產生火花，延長其壽命。

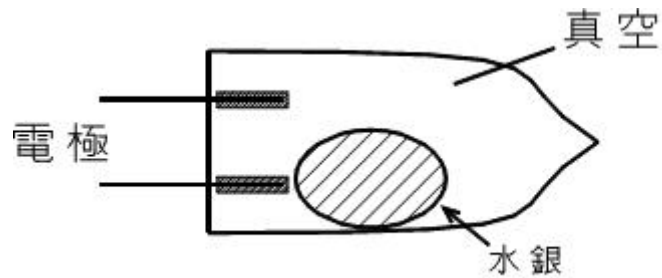


圖 2-1-2 水銀開關的構造

### (2)特性

由於水銀開關是利用液體的導電性，而且在真空內無摩擦現象可免除一般機械開關的躍動火花現象，使用壽命相當長。而且，水銀開關接觸電阻小，安定性高，在小信號的處理上，有更高的信賴性。

### (3)原理：

因為重力的關係，水銀水珠會隨容器中較低的地方流去，若果同時接觸到兩個電極的話，開關便會將電路閉合，開啟開關。

容器的形狀亦會影響水銀水珠接觸電極的條件，例如郵包炸彈使用的會是倒V字型的，令收件人在不知情的情況下傾側郵包，閉合電路、制動爆炸，但送件人（或是郵包攜帶者或製造者）則不會。

### (4)注意：

水銀對人體及環境均有毒害，故此使用水銀開關時，請務必小心謹慎，以免破損導致水銀漏出；在不再使用時，也應該妥善處理。

## 二、微電腦單晶片

微電腦普遍應用在日常生活的一些自動化設備中，可以說帶動了整個人類科技的進步。本章主要介紹微電腦組成的基本架構，單晶片微電腦的特點及其應用領域使初學者可以很快地了解一套控制系統是如何構成的。

8051 內部結構：

8051 為 Intel 公司推出的 MCS-51 系列產品之一，其結構如下圖：

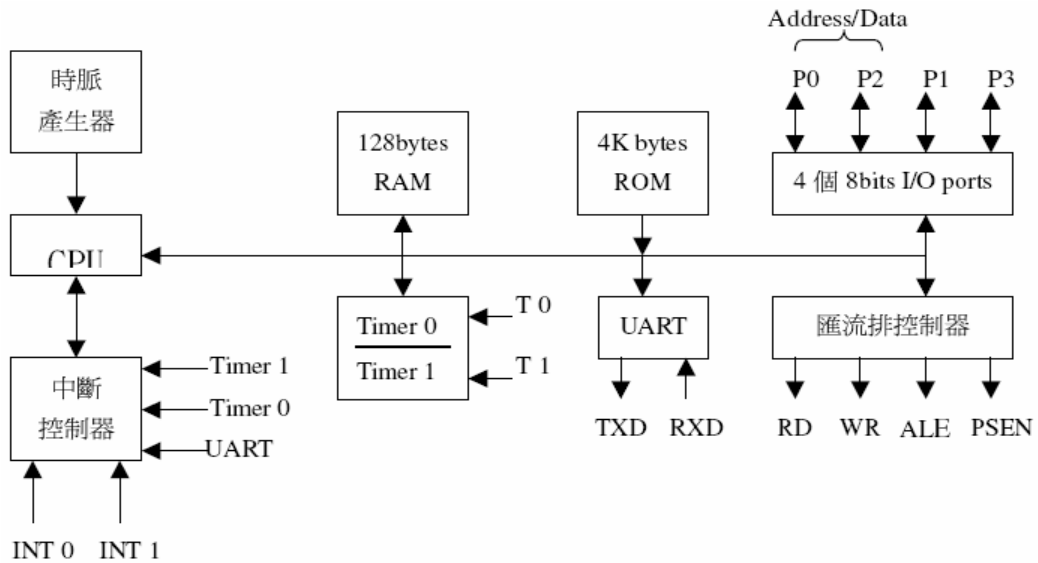


圖 2-2-1 8051 內部結構圖

8051 主要功能列舉如下：

- 為一般控制應用的 8 位元單晶片
- 晶片內部具時脈振盪器（傳統最高工作頻率可至 12MHz）
- 內部程式記憶體（ROM）為 4K 位元組
- 內部資料記憶體（RAM）為 128 位元組
- 外部程式記憶體可擴充至 64K 位元組
- 外部資料記憶體可擴充至 64K 位元組
- 32 條雙向輸入輸出線，且每條均可以單獨做 I/O 的控制
- 5 個中斷向量源
- 2 組獨立的 16 位元定時器
- 1 個全多工串列通信埠
- 8751 及 8752 單晶片具有資料保密的功能
- 單晶片提供位元邏輯運算指令

8051 單晶片的接腳

8051 為 40 隻接腳的單晶片，其位置圖如下圖所示：

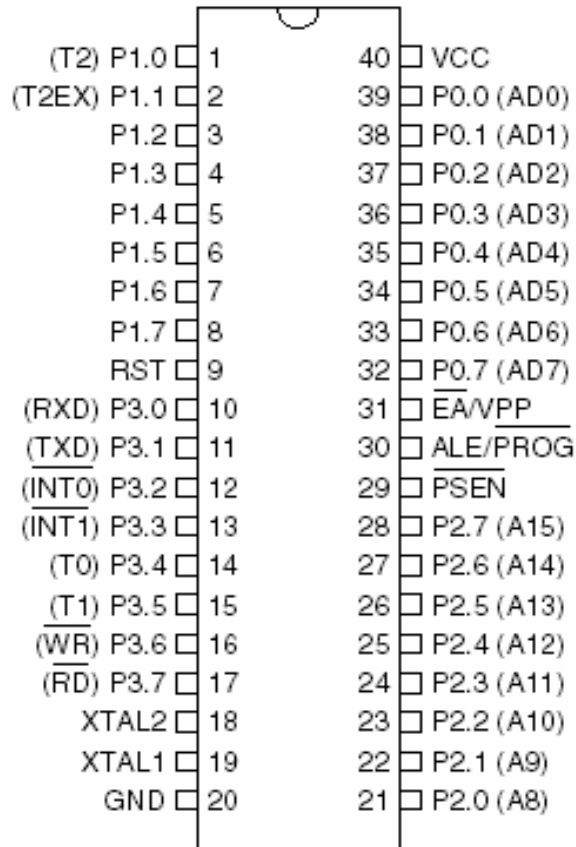


圖 2-2-2 8051 接腳圖

接腳功能說明如下(僅說明有用到的部分)：

- Pin40: VCC：8051 電源正端輸入，接+5V。
- Pin20: VSS：電源地端。
- Pin19: XTAL1：單晶片系統時脈的反相放大器輸入端。
- Pin18: XTAL2：系統時脈的反相放大器輸出端，一般在設計上只要在 XTAL1 和 XTAL2 上接上一只石英振盪晶體系統就可以動作了，此外可以在兩接腳與地之間加入一 20PF 的小電容，可以使系統更穩定，避免雜訊干擾而當機。

□ Pin9: RESET：8051 的重置接腳，高電位動作，當要對晶片重置時，只要對此接腳電位提昇至高電位並保持兩個機器週期以上的時間，8051 便能完成系統重置的各項動作，使得內部特殊功能暫存器之內容均被設定成已知狀態，並且至地址 0000H 處開始讀入程式碼而執行程式。

- Pin31: EA/Vpp：外部存取致能(External Access，EA)。

- ◇ EA=0 時，系統會存取外部記憶體中的程式碼。
- ◇ EA=1 時，系統會存取內部記憶體中的程式碼
- Pin39~32: P0.0~P0.7：Port 0 為 8 位元開路汲極(Open Drain)的雙向 I/O Port。
  - ◇ 可以推動 8 個 TTL 負載。
  - ◇ 與 MOS 元件連接時必須加上提升電阻。如下圖：

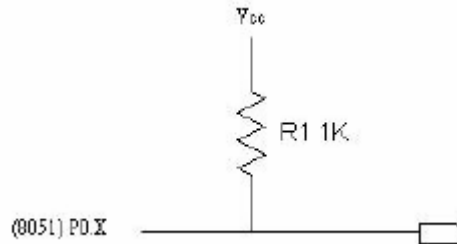


圖 2-2-3 MOS 元件連接圖

- ◇ 存取外部程式及資料記憶體時，利用多工的方式送出低位元位址(A0~A7)或資料(D0~D7)。
- Pin21~28: P2.0~P2.7：Port2 為可位元定址的雙向 I/O Port。
  - ◇ 內部具提升電阻。
- ◇ 可以推動 4 個 TTL 負載。
  - Port2 另外一個功能：存取外部記憶體時，為高位元組的位址(A8~A15)。
- Pin1~8: P1.0~P1.7
  - Port1 為可位元定址的雙向 I/O Port。
    - ◇ 內部具提升電阻。
    - ◇ 可以推動 4 個 TTL 負載。
    - ◇ Port1 大部分只做單純的 I/O 用。
- Pin10~17: P3.0~P3.7
  - Port3 為可位元定址的雙向 I/O Port。
    - ◇ 內部具提升電路的雙向 I/O Port。
      - ◇ 可以推動 4 個 TTL 負載。 Port3 另一功能如下： P3.0：RXD，串列通訊輸入。 P3.1：TXD，串列通訊輸出。 P3.2：INT0，外部中斷 0 輸入。 P3.3：INT1，外部中斷 1 輸入。 P3.4：T0，計

時計數器 0 輸入。 P3.5：T1，計時計數器 1 輸入。

P3.6：WR：外部資料記憶體的寫入信號。

P3.7：RD，外部資料記憶體的讀取信號。

基本電路連接如下圖：

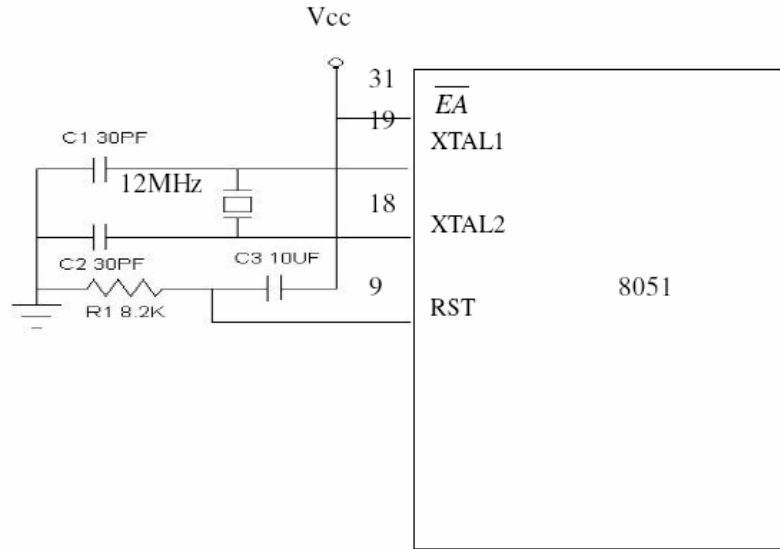


圖 2-2-4 8051 基本電路圖

8051 的記憶體分為程式記憶體及資料記憶體，所以程式與資料是 分開存放的，又可在分為內部及外部，下圖為 8051 記憶體結構示意圖：

8051 記憶體結構示意圖：

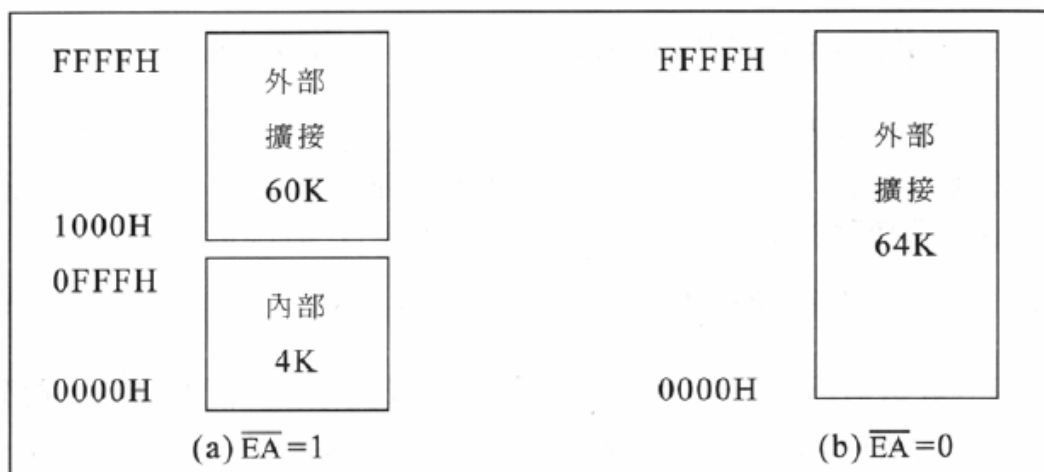


圖 2-2-5 8051 記憶體結構示意

## 參、專題製作

此章共分為三節依序說明本專題所應用到之設備及器材、製作方法與步驟及專題製作等。

### 一、設備及器材

表 3-1-1 專題製作使用儀器（軟體）設備一覽表

儀器（軟體） 設備名稱	應用說明
個人電腦	專題報告、電路圖製作及進行專題成品電路測試
數位相機	拍攝小組合作過程、專題功能使用及紀錄整個專題製作流程
雷射印表機	列印專題資料、圖片及專題報告成果
三用電表	測量零件有無損壞及專題電路板各信號之量測
IC 萬用燒錄器	利用燒錄器將程式燒錄至 89C51 單晶片中
電源供應器	提供專題成品所需之電源
Microsoft Office Word	專題報告、製作過程的撰寫
Microsoft Office Power Point	進行口頭報告、製作及專題成品報告呈現
Keil-C	單晶片組合語言程式之編輯、燒錄軟體
Protel 99SE	繪畫專題電路之線路圖

### 二、製作方法與步驟

本專題研究採用的是行動研究法，主要是由循環的研究歷程所構成，包括準備、實驗教學、電路資料分析及報告撰寫等階段。本研究之製作方法與步驟，如圖3-2-1所示。



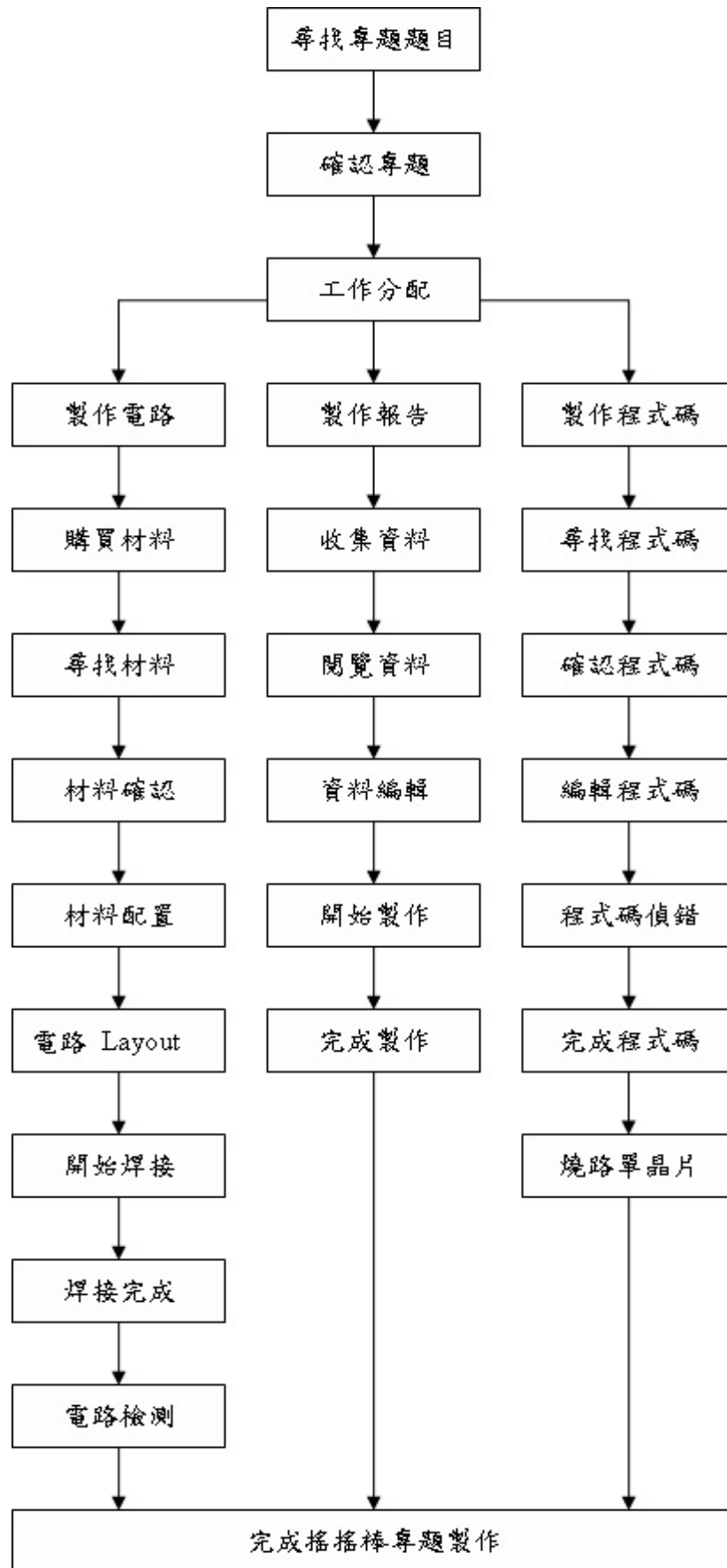


圖3-2-1 製作方法與步驟

### 三、 專題製作

表3-3-1 專題製作計畫書

專題型別		<input type="checkbox"/> 個人型專題	<input checked="" type="checkbox"/> 團隊型專題
專題性質		利用單晶片89C51製作自動化控制電路	
科別／年級		資訊 科 三 年級	
專題名稱	中文名稱	搖搖棒	
	英文名稱	Swings the stick	
專題內容簡述		本專題是利用人眼的視覺暫留惰性，通過手擺動控制一	
		排LED燈，單晶片控制他們閃爍時間和頻率，就可呈現	
		出一幅完整的畫面，本系統是以8051單晶片為核心，	
		利用16顆LED排列，利用人眼視覺暫留現象，並利用	
		按鈕方式控制畫面的切換及相關顯示控制。	
指導老師姓名		葉忠賢 老師	
參與同學姓名		張家翔(資訊3-2)	張啟強(資訊3-2)
		顏宏騰(資訊3-2)	吳惟倫(資訊3-2)
專題執行日期		101 年 9 月 1 日至 102 年 5 月 30 日	

## 硬體電路圖：搖搖棒

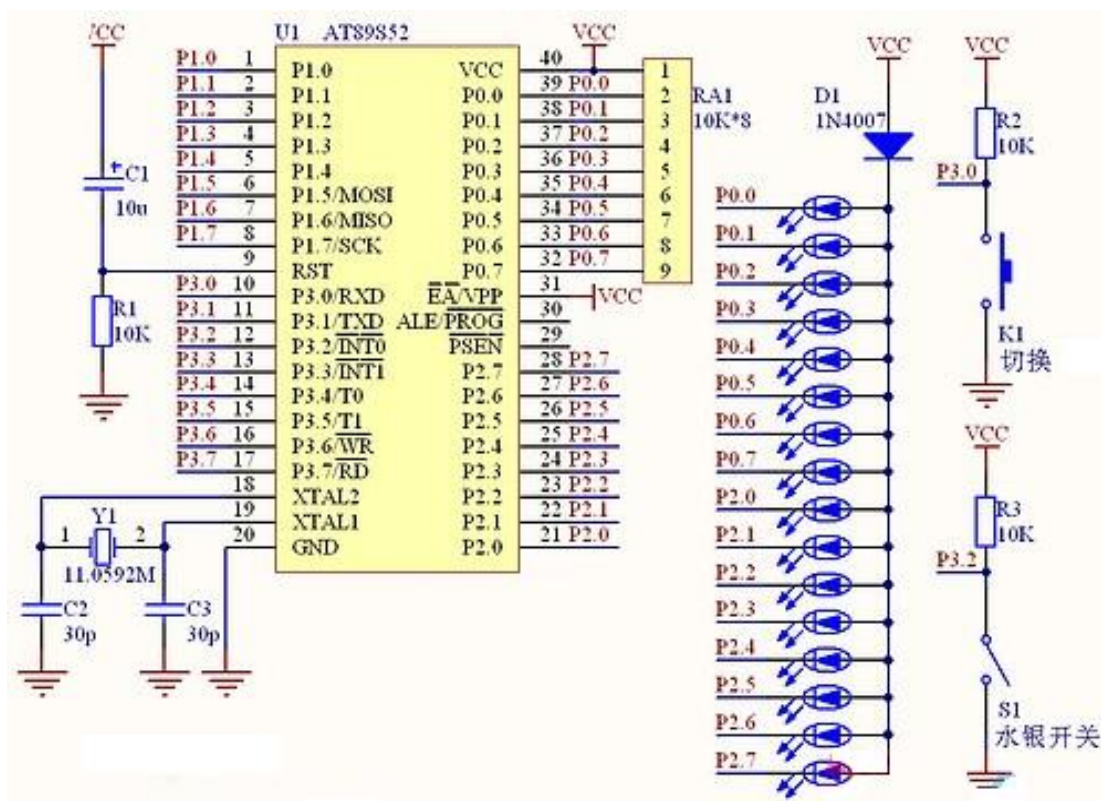


圖3-3-2 搖搖棒電路圖

基本原理與硬體電路設計：

16 隻 LED 發光管作為畫面每一列的顯示，左右搖晃起到了掃描的作用，人眼的視覺暫留現象使得看到的是一幅完整的畫面。與 LED 點陣的顯示有幾分相仿。系統電源 VCC 為 5V，下載程序和調試時一定要保證 5V 電壓，實際使用時用 3 節乾電池串聯 4.5V 即可。AT89S52 單片機作為控制器，在它的 P0、P2 口接有 16 隻以共陽的方式連接的高亮度 LED，由單片機輸出低電平點亮。P0 口的上拉電阻 RA1 不能少。串在 LED 公共端的二極管 D1 會產生一定的壓降，用來保護 LED，經實測 LED 點亮時兩端電壓為 3V 左右，在 LED 的安全承受範圍內。K1 是畫面切換開關，用於切換顯示不同內容。

水銀開關的作用：棒在搖動時，只能在朝某一方向搖動時顯示，否則會出現鏡像字或鏡像畫面，所以通過接一隻水銀開關來控制，使搖搖棒從左向右搖動時將內容顯示出來。

材料表

表3-3-2 搖搖棒材料表

材料名稱	規格	單位	數量	備註
電阻	10K ohm, 1/4W	個	3	
電阻	1K ohm, 1/4W	個	1	
電阻	5K ohm, 1/4W	個	1	
二極體	1N4007	個	2	
電解電容	10u/25V	個	1	
陶瓷電容	30p, pitch=2.5mm	個	2	
單晶片	89C51(含程式)	個	1	
XTAL	11.0592M Hz	個	1	
DIP IC 座	40Pin, 600mil	個	1	
LED	5mm(紫光)	個	16	
水銀開關	M619	個	1	
PCB	洞洞板	個	1	
電池座	3A * 4	個	1	
單排針座	2Pin	個	1	
底座	圓形(黑色)	個	1	
接觸開關		個	1	

## 肆、製作成果



圖4-1-1 麵包版電路模擬



圖4-1-2 搖搖棒程式編譯與燒錄



圖4-1-3 PC板電路確認

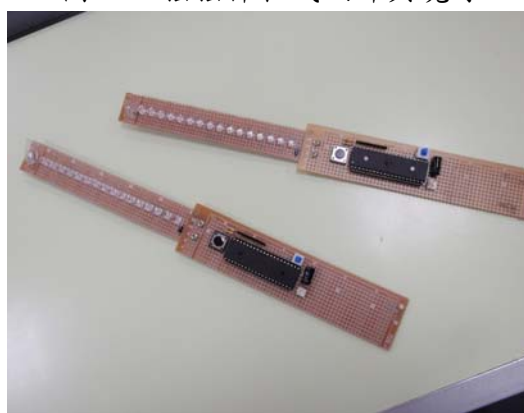


圖4-1-4 搖搖棒成品圖

至目前為止，我們電路實作部分，面對我們所遭遇的困難及問題，都已一一克服了，緊接著，我們的下一步驟即是我們的程式碼：C 語言，之前我們所學的只到 VB，而我們挑戰了更高階的語言，使得編譯程式碼困難重重，所已帶我們將之完成後，會在以影片及實體的方式呈現之。

## 伍、結論與建議

在未來的世界，電子產品將繼續朝向精緻化、生活化邁進，POVLED 肯定是未來消費市場中更加活躍的產品。配合單晶片微處理機、LED 與機械元件等技術，我們終於深入研究並了解其原理。其實，本專題真正的精神不只在於 POVLED，更重要的是，我們從這樣的一個電子產品中，發現並探討出它的價值，然後延伸於更廣泛的地方，這種概念才是我們這次專題製作的過程當中最重要收穫。

### 建議：

- (一) 可以使用電子擺動裝置使操作起來較為輕鬆
- (二) 顯現的文字與圖畫可以在多樣化
- (三) 程式碼編譯是較為困難的部分，難懂的部分多詢問專業老師
- (四) 學習前清楚的說明：請老師在進行專題製作學習前，能對學生清楚的說明整個專題進行的方式，包括專題報告的格式、課程進度的安排、需要的準備工具以及評量方式等，如都能在事前做好詳細的說明、規範，如此則能避免學生因疑惑而做錯方向。
- (五) 在學習過程中給予回饋：同學建議，在專題製作學習研究過程中，老師能否可以在學習的進行過程，給予立即性的回饋，讓學生可以及早發現其缺失，盡早進行改善。
- (六) 增長專題製作學習的時間：進行專題活動的學習，每個階段皆需完成一個學習報告，而單元學習的時間太少，連帶影響了期末完整報告的製作，所以希望老師能增長同學學習時間，讓成果報告的製作能更加完整，避免同學因時間緊迫而草率完成其作品。

### 結論：

- (一) 透過此次專題製作學習的方式能幫助我們提升對課程的學習興趣。
- (二) 透過此次專題製作學習的方式更能幫助我們獲得知識的建構及整合，且亦可以幫助我們提升其創造思考的能力。
- (三) 小組同學認為專題製作學習為一主動探究的學習，學習，學習，中強調學習者必強調學習者必須負起主動探究學習的責任。
- (四) 小組同學認為專題製作學習的階段中，會遇到，會遇到不同的困難及問題，但看不同的困難及問題，但看不同的困難及問題，但看到自己的成品時，會很有成就感。

## 參考文獻

- 1.朱永昌，2007，8051 單晶片微電腦原理與專題製作(上)，台北縣：台科大圖書公司。
- 2.吳金成、郭庭吉，2008，單晶片 8051 專題製作-使用 Keil AX51，台北市：文魁資訊。
- 3.林明德，WonDerSun，2008，專題製作-電子電路篇，台北縣：台科大圖書公司。
- 4.長高企業，1998，U3-TARGET 單晶微電腦實驗裝置，台中市：長高企業公司。
- 5.柯南，2001，全能電路設計之 Protel Schematic 99 SE，台北縣：台科大圖書公司。
- 6.郭庭吉，2008，8051 單晶片微電腦專題製作，台北縣：台科大圖書公司。
- 7.創意導航核心，2007，PhotoImpact 12 影像哈燒秀，台北市：全華圖書公司。
- 8.蔡朝洋，2007，單晶片電腦 8051/8951 原理與應用，台北縣：全華圖書公司。
- 9.鄧明發，陳茂璋，2000，微電腦專題製作應用電路，台北市：知行文化公司。
- 10.鍾明政，1999，單晶片 8051 原理與實作，台中市：長高企業公司。
- 11.張義和、王敏男、許宏昌、余春長，2012，例說89S51-C語言，新北市，新文京開發出版股份有限公司

## 附錄一 搖搖棒之程式碼

/\*\*\*\*\*

程序名稱：LED 搖搖棒顯示 64\*16 像素

簡要說明：外部中斷方式 INT0 顯示；取模方式：縱向取模、字節倒序

編 寫：JiangX.net

最後更新：08/11/15 晚

\*\*\*\*\*/

```
#include <AT89X52.h>
```

```
#define uchar unsigned char
```

```
#define uint unsigned int //宏定義
```

```
#define KEY P3_0 //定義畫面切換按鍵
```

```
uchar KY; //KY 作用在後面說明
```

```
uchar disp; //顯示漢字指針
```

```
uchar pic=0,num=0; //pic 為按鍵次數；num 為中斷次數
```

```
uchar code love[] = {
```

```
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
```

```
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x3F,0x00,0x20,0x00,0x20,0x00,0x20,
```

```
0x00,0x20,0x00,0x20,0x00,0x20,0x00,0x20,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
```

```
0x04,0x10,0x02,0x20,0x02,0x20,0x02,0x20,0x02,0x20,0x04,0x10,0xF8,0x0F,0x00,0x00,
```

```
0x00,0x00,0x00,0x00,0xFE,0x07,0x00,0x08,0x00,0x10,0x00,0x20,0x00,0x20,0x00,0x10,
```

```
0x00,0x08,0xFE,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x3F,0x82,0x20,0x82,0x20,
```

```
0x82,0x20,0x82,0x20,0x82,0x20,0x82,0x20,0x82,0x20,0x00,0x00,0x00,0x00,0x00,0x00,
```



```

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,/*LOVE*/
};

```

```

uchar code loveyou[] = {
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,
0x78,0x00,0xFC,0x00,0xFE,0x01,0xFE,0x03,0xFE,0x07,0xFE,0x0F,0xFE,0x1F,0xFC,
0x3F,
0xF8,0x7F,0xFC,0x3F,0xFE,0x1F,0xFE,0x0F,0xFE,0x07,0xFE,0x03,0xFE,0x01,0xFC,
0x00,
0x78,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,/*心形圖案*/
};

```

```

uchar code hehe[] = {
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,
0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0x01,0x40,0x01,0xC0,0x01,0x00,0x00,0x00,0x00,0x0
0,
0xF0,0x0F,0x08,0x10,0x04,0x20,0x00,0x00,0x00,0x00,0xF0,0x3F,0x08,0x00,0x04,0x0
0,
};

```

```

0x04,0x00,0x04,0x00,0x08,0x00,0xF0,0x3F,0x00,0x00,0x00,0x00,0x20,0x00,0x2
0,
0x00,0x20,0x00,0x20,0x00,0x20,0x00,0x00,0x00,0x00,0xF0,0x3F,0x08,0x00,0x04,0x0
0,
0x04,0x00,0x04,0x00,0x08,0x00,0xF0,0x3F,0x00,0x00,0x00,0x00,0x04,0x20,0x08,0x1
0,
0xF0,0x0F,0x00,0x00,0x00,0x00,0xC0,0x01,0x40,0x01,0xC0,0x01,0x00,0x00,0x00,0x
00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x0
0,/*呵呵 o(∩_∩)o 圖案*/
};

```

```

unsigned char code hanzi[] = {

```

```

//-- 歡 --

```

```

    0x04,0x08,0x74,0x06,0xDE,0xFF,0x74,0x55,
    0x84,0x55,0x74,0x7F,0x5E,0x55,0x74,0x55,
    0x84,0x80,0x60,0x60,0x1E,0x18,0xD2,0x07,
    0x10,0x18,0x50,0x60,0x38,0x80,0x00,0x00,

```

```

//-- 迎 --

```

```

    0x10,0x82,0x22,0x62,0x24,0x1F,0x04,0x22,
    0x00,0x44,0xFE,0x4F,0x04,0x84,0x02,0x82,
    0x02,0x81,0x00,0x80,0xFE,0xBF,0x04,0x84,
    0x04,0x8C,0xFE,0x47,0x04,0x40,0x00,0x00,

```

```

//-- 光 --

```

```

    0x00,0x00,0x00,0x81,0x08,0x81,0x10,0x41,
    0x30,0x31,0x60,0x0F,0x00,0x01,0x00,0x01,
    0xFE,0x01,0x02,0x7F,0x40,0x81,0x30,0x81,
    0x18,0x81,0x08,0x81,0x80,0xF1,0x00,0x41,

```

//-- 臨 --

0xFE,0x7F,0x44,0x22,0x7C,0x3E,0x46,0x22,  
0xE4,0x27,0x44,0x20,0x20,0xFC,0x18,0x44,  
0xEE,0x45,0x2A,0xFD,0x28,0x01,0x28,0xFD,  
0xE8,0x45,0x0C,0x44,0x08,0xFC,0x00,0x00,

//-- 德 --

0x10,0x01,0x88,0x00,0xC6,0xFF,0x32,0x00,  
0x10,0x40,0x08,0x32,0xE8,0x02,0xA8,0x72,  
0xE8,0x82,0xBE,0x8A,0xAA,0xB2,0xE8,0x82,  
0xA8,0x62,0xEC,0x12,0x08,0x62,0x00,0x00,

//-- 霖 --

0x20,0x44,0x1C,0x34,0xA8,0x0C,0xAA,0xFE,  
0xAA,0x14,0xAA,0x24,0x0A,0x00,0xFE,0x44,  
0x0A,0x34,0xAA,0x0C,0xAA,0xFE,0xAA,0x14,  
0xA8,0x26,0x1C,0x44,0x08,0x40,0x00,0x00,

//-- 技 --

0x10,0x46,0x10,0xC2,0xFE,0x7F,0x12,0x01,  
0x90,0x00,0x00,0x00,0x90,0x80,0x90,0x81,  
0x90,0x46,0xFE,0x28,0x92,0x18,0x90,0x24,  
0xD0,0x43,0x98,0x80,0x10,0x80,0x00,0x00,

//-- 術 --

0x10,0x01,0x88,0x00,0xC6,0xFF,0x32,0x00,  
0x10,0x10,0xA0,0x0F,0x20,0x00,0xFE,0xFF,  
0x20,0x00,0xA4,0x0F,0x08,0x10,0x24,0x5C,  
0x24,0xC0,0xE4,0x7F,0x24,0x00,0x00,0x00,

//-- 學 --

0x00,0x04,0x00,0x13,0xFE,0x11,0x54,0x11,  
0x52,0x15,0x00,0x15,0xAA,0x55,0x44,0xD5,  
0xAA,0x75,0x00,0x1D,0x54,0x15,0x54,0x11,  
0xFE,0x15,0x04,0x13,0x00,0x01,0x00,0x00,

//-- 院 --

0xFE,0xFF,0x04,0x08,0x64,0x18,0x9E,0x08,  
0x04,0x07,0x20,0x80,0x1C,0x82,0x48,0x62,  
0x48,0x1E,0x4E,0x02,0x4A,0x02,0x48,0x7E,  
0x68,0x82,0x1C,0x83,0x08,0xF2,0x00,0x00,

/\*心形圖案\*/

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0,  
0x78,0x00,0xFC,0x00,0xFE,0x01,0xFE,0x03,0xFE,0x07,0xFE,0x0F,0xFE,0x1F,0xFC,  
0x3F,  
0xF8,0x7F,0xFC,0x3F,0xFE,0x1F,0xFE,0x0F,0xFE,0x07,0xFE,0x03,0xFE,0x01,0xFC,  
0x00,  
0x78,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0,

//-- 電 -- ,

0x20,0x00,0x1C,0x00,0xA8,0x00,0xAA,0x3E,  
0xAA,0x2A,0xAA,0x2A,0x0A,0x2A,0xFE,0x7E,

0x0A,0xAA,0xAA,0xAA,0xAA,0xAA,0xAA,0xBE,  
0xA8,0x80,0x1C,0xF0,0x08,0x40,0x00,0x00,

//-- 空白 --

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

//-- 子 --,

0x00,0x01,0x00,0x01,0x04,0x01,0x04,0x01,  
0x04,0x21,0x04,0x41,0x04,0xC1,0xE4,0x7F,  
0x24,0x01,0x14,0x01,0x14,0x01,0x0E,0x01,  
0x04,0x01,0x80,0x01,0x00,0x01,0x00,0x00,

//-- 空白 --

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,  
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

//-- 工 --

0x00,0x40,0x08,0x40,0x08,0x40,0x08,0x40,

```

0x08,0x40,0x08,0x40,0x08,0x40,0xF8,0x7F,
0x08,0x40,0x08,0x40,0x08,0x40,0x08,0x40,
0x0C,0x40,0x08,0x60,0x00,0x40,0x00,0x00,

//-- 程 --
0x24,0x08,0x24,0x06,0xA4,0x01,0xFE,0xFF,
0x22,0x01,0x22,0x06,0x00,0x40,0x7E,0x49,
0x24,0x49,0x24,0x49,0xA4,0x7F,0xA4,0x48,
0xA4,0x4C,0xFE,0x68,0x04,0x40,0x00,0x00,

//-- 系 --
0x00,0x00,0x04,0x80,0x04,0x42,0x24,0x36,
0x24,0x12,0x34,0x03,0xCC,0x02,0x46,0xFE,
0x22,0x02,0x92,0x02,0x12,0x11,0x02,0x23,
0x02,0x66,0x00,0xC0,0x00,0x00,0x00,0x00,

//-- 空白 --
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

};

/*****函數聲明*****/
void display1(void);
void display2(void);
void display3(void);
void display4(void);

```

```

/*****n (us) 延時子程序*****/
void DelayUs(uint N)
{
    uint x;
    for(x=0; x<=N;x++);
}

/*****中斷服務程序*****/
void intersvr0(void) interrupt 0 using 1
{
    KY=~KY;      //每個搖動來回水銀開關會在擺幅兩端分別產生下降沿中斷，只
    提取其中一次（從左向右搖才顯示）
    if(KY==0)
    {
        num++;    //計算中斷次數
        switch(pic)    //選擇畫面
        {
            case 0:{display1();}break;
            case 1:{display2();}break;
            case 2:{display3();}break;
            case 3:{display4();}break;
            default: {display1();}
        }
    }
}

/*****顯示子程序 1（漢字）*****/
void display1(void)
{
    uchar i;

```

```

if(num>10){disp++;num=0;} //15 個漢字分為 5 次顯示完（每次顯示 4 個），
每中斷 10 次切換
if(disp>4)disp=0;
DelayUs(5200); //此處延時時間依各硬件差別而各不相同，試著調整使得顯
示內容居中即可
for(i=0;i<64;i++)
{
    P0=~hanzi[disp*128+i*2];
    P2=~hanzi[disp*128+i*2+1];
    DelayUs(100);
}
}

```

/\*\*\*\*\*\*顯示子程序 2 (LOVE) \*\*\*\*\*/

```
void display2(void)
```

```

{
    uchar i;
    DelayUs(4000);
    for(i=0;i<64;i++)
    {
        P0=~love[i*2];
        P2=~love[i*2+1];
        DelayUs(120);
    }
}

```

/\*\*\*\*\*\*顯示子程序 3 (心形圖案) \*\*\*\*\*/

```
void display3(void)
```

```

{
    uchar i;

```



```

    DelayUs(4000);
for(i=0;i<64;i++)
{
    P0=~loveyou[i*2];
    P2=~loveyou[i*2+1];
    DelayUs(120);
}
}

/*****顯示子程序 4 (呵呵 o(∩_∩)o 圖案) *****/
void display4(void)
{
    uchar i;
    DelayUs(4000);
for(i=0;i<64;i++)
{
    P0=~hehe[i*2];
    P2=~hehe[i*2+1];
    DelayUs(120);
}
}

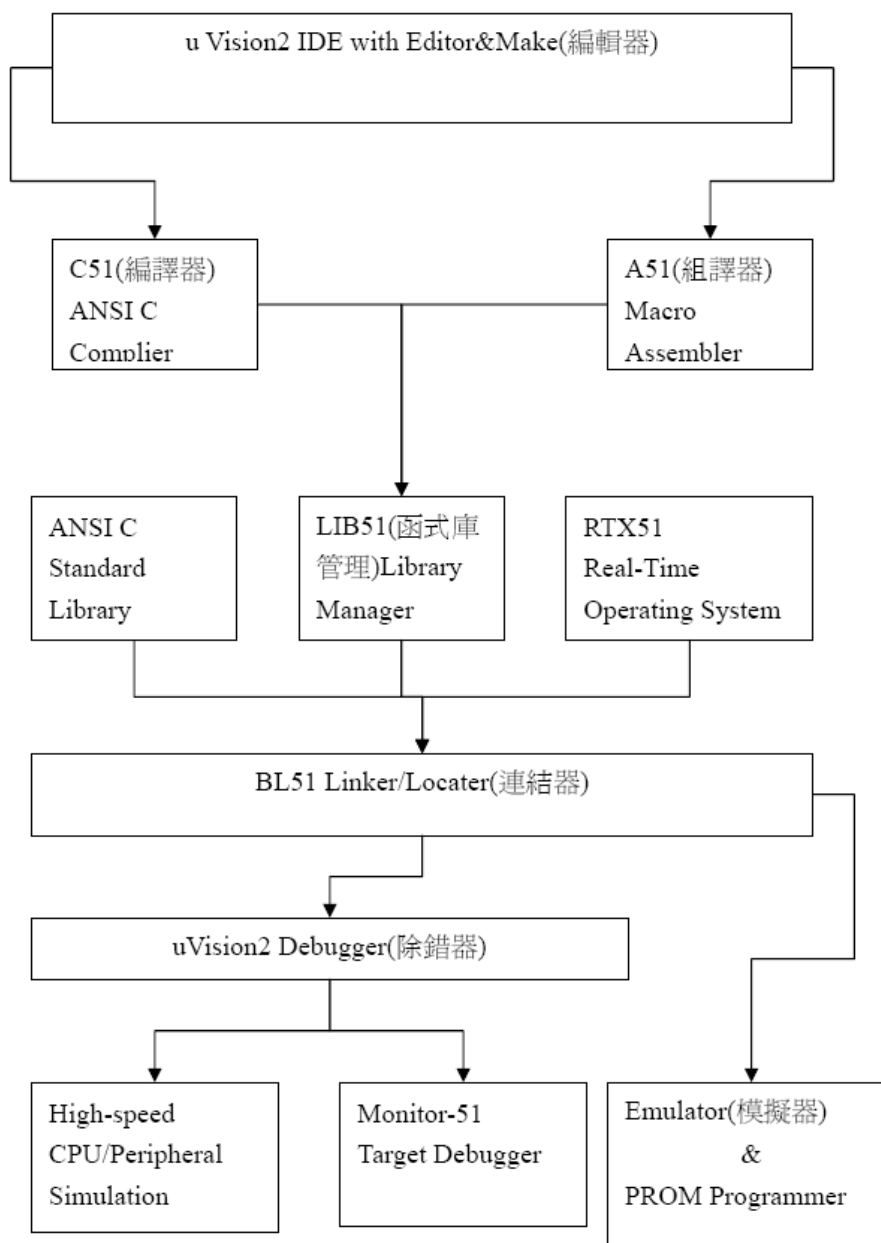
/*****主函數*****/
void main(void)
{
    IT0=1;
    EX0=1;
    EA=1;    //開中斷，下降沿中斷
    KY=0;
while(1)    //主程序中只檢測按鍵

```

```
{
if(KEY==0)    //畫面切換鍵按下
{
DelayUs(10000);    //按鍵去抖
if(KEY==0);
pic++;}
if(pic>3)pic=0;
}
}
/*****END*****/
```

## 附錄二 程式工具Keil C 介紹

u Vision2 視窗版是一個獨立而且功能強大的整合性開發環境，結合了計畫 (project) 經營管理、原始程式編輯器(editor)、組譯器(assembler)、編譯器(compiler)、連結器(linker/locater)、程式除錯器(debugger)等功能，其方塊流程圖如下圖所示。



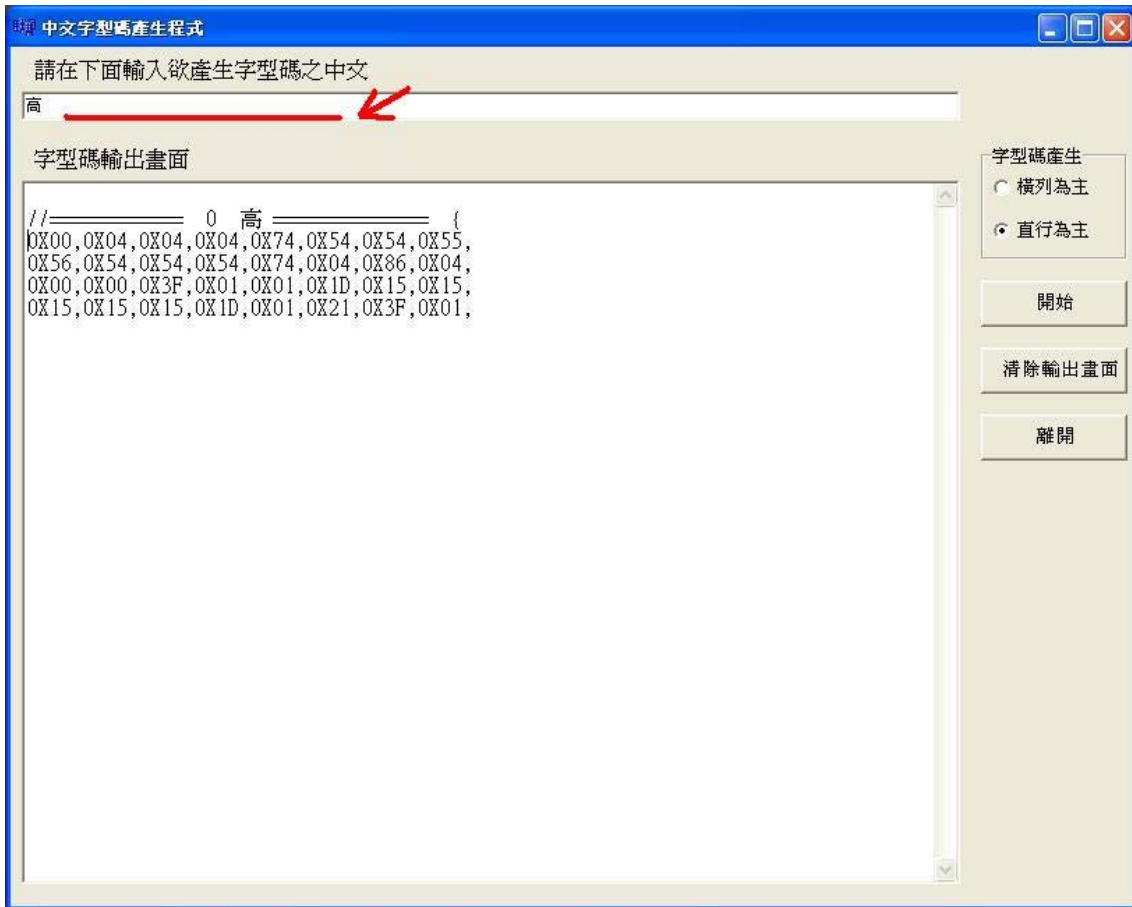
圖附錄1 Keil C 流程圖

組合語言與C 語言的優缺點：

- (1) 所有有牽涉到精確計時的副程式中，用組合語言會有更快的反應度。
- (2) 所有8051 C 語言編譯程式都必小心撰寫，以免造成當機

- (3) C 語言編譯時可指定對執行速度或是程式碼CODE SIZE 進行最佳化。
- (4)大部份C 語言的最佳化僅僅是針對該語言而已，並未對8051 的程式碼進行最佳化。
- (5)當C 語言處理中斷時的速度並不慢，只是在使用時須瞭解其進入及返回原程式的步驟，以免造成當機。
- (6) C 語言所寫的控制系統比組合語言而言，有更多不確定性存在，這些不確定性包含了我們程式寫法的執行錯誤(RUNTIMEERROR)，以及編譯程式的隱藏性BUG，所以必需嚴謹的驗證程式的步驟。

### 附錄三 中文字型編譯程式



圖附錄 2 中文字型產生程式 (GetChWdPat)

首先搖搖棒由搖晃後所產生的文字是國字，但是國字還是需要用編碼器來編出程式碼加以燒錄至 IC，這個編碼程式操作如上圖，開啟 GetChWdPat 這個編碼程式後（需要有 stdfont.15 此檔案才能成功開啟）。

在”紅色箭頭”所標示之處，輸入想要的文字之後，在字型碼產生之處點選”直行為主”之後，點選”開始”就會開始進行編碼，編輯後的程式碼會成現在”字型碼輸出畫面”之處，就可以複製所編碼完的程式碼到燒錄 IC 的程式 (Wellon) 裡進行燒錄。