高雄市高英高級工商職業學校

Kao Ying Industrial Commercial Vocational High School

專題製作報告

脈搏感測器

學生姓名：陳冠伯

李昕頤

林品傑

吳威承

指導老師：葉忠賢老師

中華民國 104 年 05 月

誌　謝

感謝葉忠賢老師，再高中三年級的時候，教導了我們在電腦上的技術，也教導了我們許許多多的做人處事道理，方向來邁進，更謝謝老師的指導，讓我們可以重新發現自己長久以來的不足與缺點，並且在學習的過程中，與老師師生之間的感情更加深厚。在我們不會的電路及沒有見過的零件老師都加以詳細告訴我們其中的功能，例如某些零件的腳位亦或者是程式碼上面出現的錯誤老師都會耐心的告訴我們其中的錯誤道理，沒有排解出這一些困難這一個專題想必很難完成所以非常的感謝老師對我們的加以重視希望在未來的專題中還可以繼續幫助我們找出問題排除困難，真的是很感謝老師及組員大家的合作。

陳冠伯、李昕頤、林品傑、吳威承　謹上 2015/05

脈搏感測器

摘　要

本專題藉脈搏感測輸入給 89S51 脈搏的資訊，89S51 並依此訊息做出顯示。脈搏感測是藉由紅外線感測元件來偵測脈搏，從微血管感應脈搏，達成偵測脈搏的效果，脈搏資訊將經由 89S51 程式進行處理計算，再由七段顯示器來呈現使用者的脈搏，這樣便結合了人體的建康資訊，踏出美好生活的第一步。

隨著工作步調加快以及工時的加長，工作壓力與日俱增，因工作過度勞累而發生猝死的意外時有所聞。根據統計，過勞患者的年齡多在 20 至 45 歲之間。

通常心跳感測器為了節省時間，一般不會作 1 分鐘的量測，通常是量測 10 秒鐘內心跳次數，再把結果乘以 6 即得到每分鐘的心跳數，即使這樣做還是比較費時，而且精度也不高。本專題改進方式為用單晶片 8051 製作的心跳量測器，只要人把手指放在感測器 1 個心跳間隔就可以推算出每分鐘的心跳數，測量結果用三位數七段數字顯示。本系統通過心跳感測器採集訊號輸出電壓訊號，經過訊號放大器對其進行放大。然後將放大後的訊號整形為單晶片易於處理的方波訊號。通過單晶片實現頻率的量測和計算並顯示在七段顯示器上。

關鍵詞：89S51 單晶片、脈搏感測、紅外線感測元件。

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 1
　四、製作預期成效 2
貳、理論探討 3
　一、電子相關零組件 3
　二、單晶片簡介 4
參、專題製作 15
　一、設備及材料 15
　二、製作方法和製作步驟 16
　三、專題製作 18
肆、製作成果 22
伍、結論與建議 23
　一、結論 24
　二、建議 24
參考文獻 25
附錄一 脈搏感測器程式碼 26

表目錄

表 2-2-1 MCS-51 的成員 4
表 2-2-2 中斷服務程式的進入點 8
表 2-2-3 8051 的前面 128 個位元位址 11
表 2-2-4 SFR 各個暫存器重置後的初始值 13
表 3-1-1 專題製作使用儀器（軟體）設備一覽表 15
表 3-1-2 專題製作計畫書 16
表 3-3-2 脈搏感測器之材料表 22

圖目錄

圖 1-3-1 專題製作流程圖 2
圖 2-1-1 CNY70 動作原理 CNY70 3
圖 2-1-2 紅外線 CNY70 電路 4
圖 2-2-1 單晶片的 40Pin DIP 包裝的接腳圖 5
圖 2-2-2 8051 記憶體映像圖 7
圖 2-2-3 將 PSEN 與 RD 合併成 MRD 9
圖 2-2-4 整形電路 14
圖 3-2-1 製作方法及步驟 18
圖 3-3-1 專題製作的簡報 19
圖 3-3-2 專題製作的 Word 19
圖 3-3-3 脈搏感測器電路圖 20
圖 3-3-4 紅外線原理作用圖 21
圖 4-1-1 脈搏感測器 製作過程(一) 22
圖 4-1-2 脈搏感測器 製作過程(二) 22
圖 4-1-3 脈搏感測器成品圖 22
圖 4-1-4 脈搏感測器測試功能 22
圖 4-1-5 脈搏感測器 功能介紹 23

壹、前言

一、製作動機

隨著工作步調加快以及工時的加長，使得資訊電腦業人員工作壓力與日俱增，工程師因工作過度勞累而發生猝死的意外時有所聞。根據統計，過勞患者的年齡多在 20 至 45 歲之間，不但是社會的菁英份子，也是一家經濟之所繫，一旦倒下，將增加社會成本的負擔。

二、製作目的

通常脈搏感測器為了節省時間，一般不會作 1 分鐘的量測，通常是量測 10 秒鐘內脈搏次數，再把結果乘以 6 即得到每分鐘的脈搏數，即使這樣做還是比較費時，而且精度也不高。

本專題改進方式為用單晶片 8051 製作的脈搏量測器，只要人把手指放在感測器 1 個脈搏間隔就可以推算出每分鐘的脈搏數，測量結果用三位數七段數字顯示。本系統通過脈搏感測器採集訊號輸出電壓訊號，經過訊號放大器對其進行放大。然後將放大後的訊號整形為單晶片易於處理的方波訊號。通過單晶片實現頻率的量測和計算並顯示在七段顯示器上。

三、製作架構

(一）專題製作流程

我們先研究討論主題，我們分成三組分配工作；一組材料購買，一組搜尋資料，一組電路設計，最後研究與整理，畫 Layout 圖，一起討論製作簡報，製作電路板，報告書的製作，最後製作完成。

(二)製作流程圖

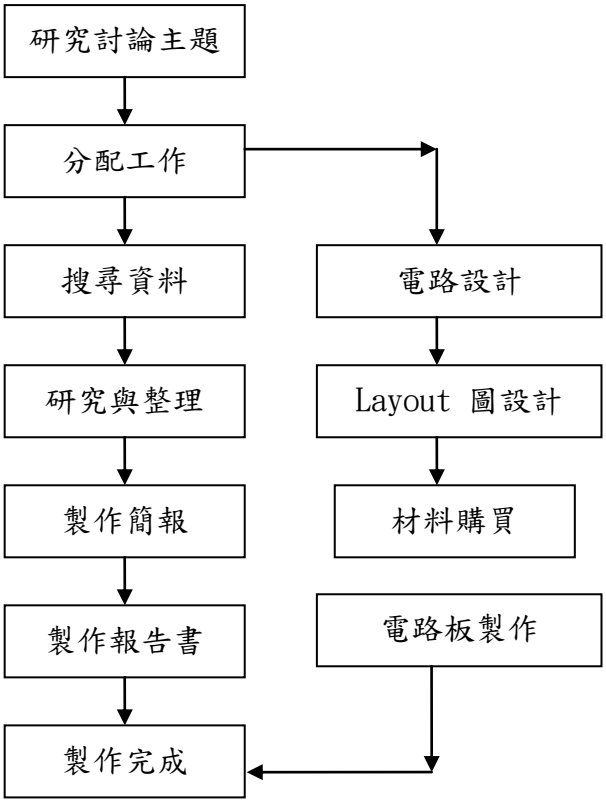


圖 1-3-1　專題製作流程圖

四、製作預期成效

有了脈搏感測器，可以隨時隨地讓人們測量自己的脈搏，也能可以給醫生做參考資料，能夠更方便醫生做治療。

(1)　利用手指放在感測器 1 個脈搏間隔就可以推算出每分鐘的脈搏數。

(2)　測量結果用三位數七段數字顯示。

(3)　量測速度快、精準度高、立即有效。

貳、理論探討

一、電子相關零組件

紅外線

紅外線感測器為利用光線以檢知受體的一種感測器。使用上分為二類：

A. 反射型：光源與感光元件並排放置，光線是否自壁面反射來判斷壁面的方式。

B. 遮光型：光源與感光元件設置相向的位置，當從光源射往感光元件的光線被遮斷時，即判斷其間有壁面存在的方向。光感測器使用的光大致上有分為可見光及紅外線兩種，分別稱為可見光感測器及紅外線光感器。可見光的壞處是易受環境的影響，例如可見光會被黑色吸收。使用紅外線的好處是，其不會受到室內照明的影響，但由於紅外線無法看見，很難確認其動作。

紅外線 CNY70 內部結構

CNY70 的內部結構如圖2-1-1所示，其中包含紅外線發光二極體，光電晶體，以及光濾波器，其功能分別是：

A. 紅外線發光二極體：類似發光二極體(LED)的功能，當PN 兩端加上順向偏壓時可發出波長為800nm 的紅外線不可見光。

B. 光電晶體：為一個對紅外線波長具敏感反應的光偵測元件，當光電晶體受紅外線光照射時為低阻抗，而未受光時呈現高阻抗。

C. 濾波器：唯一僅讓波長為紅外線附近光譜通過的濾光透境，可用來加強光電晶體的雜抗訊能力(紅外線以外不可見與可見光的干擾)。

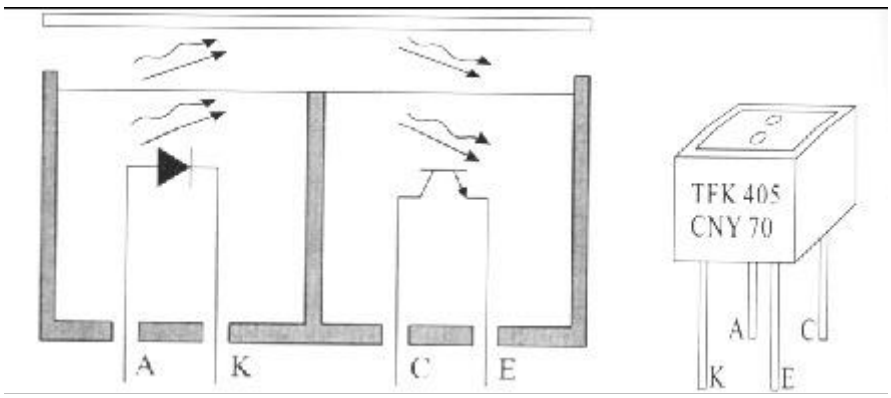


圖2-1-1 CNY70 動作原理 CNY70

紅外線 CNY70 動作原理：CNY70 光感測器，其光源和感測元件是做在一起，其動作是光遇到普通地面，就會反射到感測元件。

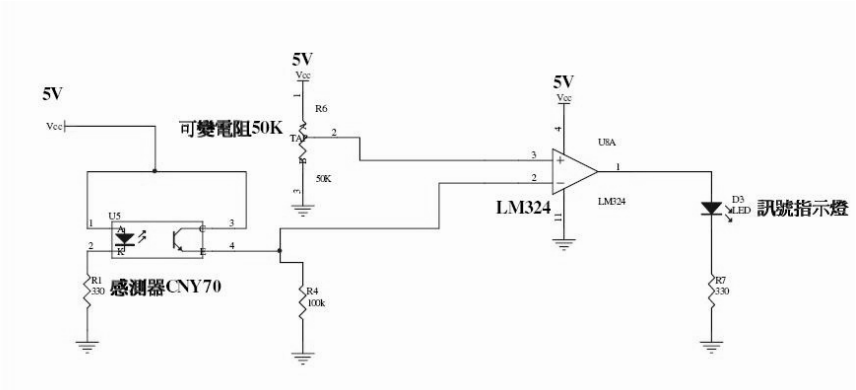


圖2-1-2 紅外線 CNY70電路

另一組紅外線發射器和接收器所組成的模組，特性如下：

(1)具低消耗、高靈敏度、防雜訊等優點。

(2)紅外線接收器接收距離為10公尺。

(3)搭配編解碼晶片(PT2248、PT2249A)，不易受外界雜訊干擾。

二、單晶片簡介

MCS-51是Intel公司所設計的8051系列單晶片的總稱，較具知名度的編號有8051、8751和8031，這些不同的單晶片都使用相同的核心CPU與指令集，只是能在靠製造IC時給予不同的周邊設計，分別賦予這些IC一個特別編號。

表2-2-1 MCS-51的成員

名稱	ROMLESS	EPROM	ROM(位元組)	RAM(位元組)	16 位元計時器	電路型式
8051	8031	(8751)	2K	128	2	HMOS
8051AH	8031AH	8751H	2K	128	2	HMOS
8052AH	8032AH	8752H	2K	256	3	HMOS
80C51BH	80C31BH	87C51	2K	128	2	CHMOS

以下將MCS-51系列單晶片的主要功能列舉如下：

1. 專為控制應用所設計的8位元CPU
2. 有完整的單位元邏輯運算指令
3. 有32條(4個Port)雙向且每條都可以被單獨定址的I/O

4. 內部有128byte可供讀/寫的RAM
5. 內部有兩個16位元Timer/Counter
6. 有一個通信用的全雙工UART(串列I/O)
7. 可接受5個中斷源，且有2層優先權的中斷結構
8. 內部有時脈振盪器(最高頻率可到12MHz)
9. 內部有4K的程式記憶體
10. 可在外部擴充到64K程式記憶體(EPROM)
11. 可在外部擴充64K資料記憶體(RAM)

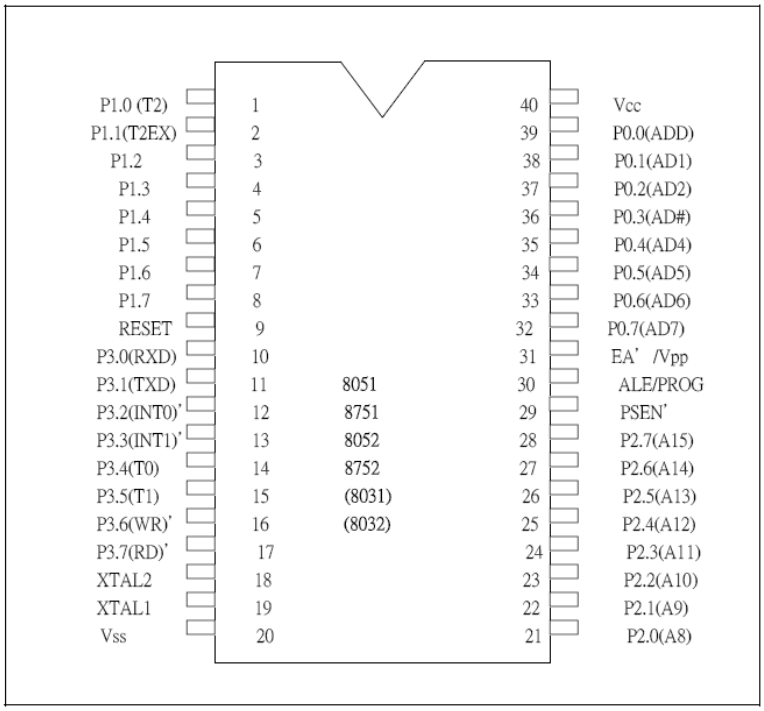


圖2-2-1 單晶片的40Pin DIP包裝的接腳圖

1~8 腳 (P1.0~P1.7)：

這8支角是8051的I/O port，稱為PI。第一腳(P1.0)是LSB，第8支腳(P1.7)是MSB。如果是8052(8032或8752)時，P1.0又可當作Timer2的外部脈波輸入腳，P1.1又當作T2EX，可當作另外一個外部中斷觸發輸入腳。P1上的每支腳都可推動4個LS TTL。

9 腳(RESET)：

8051的重置(RESET)輸入腳，當這支腳由外部輸入High(+5V)的信號時，8051就被重置，8051被重置後就從位址0000H開始執行程式。且特殊功能暫存器(SFR)裡的所有暫存器都會被設成已知狀態。

10~17 腳 (P3.0~P3.7)：

這8支腳是8051的I/O port，稱為P3。第10支腳(P3.0)為LSB，第17支腳(P3.7)為MSB。P3裡的每支I/O腳除了可以當作單純的輸入/輸出使用外，也當作8051內部的某些週邊與外界溝通個I/O腳。例如P3.0和P3.1接腳的另外一個名稱為RxD和TxD，當8051內部的UART被軟體啟動後，UART會將串列資料從TxD腳輸出，而UART也接收由外部送進來的串列信號。INT0和INT1是8051的兩個外部中斷輸入部。T0是Timer0的外部脈波輸入腳。T1是Timer1的外部脈波輸入腳。WR，RD，當您再8051的外部擴充資料記憶體(RAM)時，這兩條線是控制寫與讀的信號。P3上的每一隻I/O腳都可以做兩種用途。那8051怎麼知道P3上的某支腳是當I/O或當另一種用途，例如您要使用UART時您將第10腳看成RxD，第11腳看成TxD加以使用就可以了。但是有一點必須特別注意，那就是當作其他功能(不當I/O使用)使用的那支腳的內部栓鎖器的內容必須設為1，其他的功能(如TxD，RxD，RD，ER，…等)才會有作用。P3上的每支I/O腳都可推動4個LS TTL。

18~19 腳(XTAL2，XTAL1)：

這兩支腳是8051內部時脈振盪器的輸入端，您可以在這兩支腳上跨街一個12MHz的工作頻率，供內部使用。8051會根據這個速度工作。若未特別註明，這個振盪器的工作頻率是在1MHz~12MHz之間的任何一個。如果線路板上已有振盪器，那這個振盪器所產生的脈波(Clock)也可以直接輸入給8051使用。這個外部送給8051使用的脈波是從第18腳(XTAL2)輸入，而19腳(XTAL1)必須接地，以上的接法是CMOS的8051(如8051AH)。

如果您是使用CMOS的8051(80C51，80C31等)，外部的脈波必須從19腳(XTAL1)輸入而18腳空接，這個差別必須特別注意。

40，20 腳(Vcc，Vss)：

這是8051的電源輸入端，40腳接電源的正端的20腳接地。電源規格是5V +/- 10% 。

21~28 腳(P2.0~P2.7)：

這8支腳是8052的I/O port，稱為P2，P2.0為LSB，P2.7為MSB。P2除了當作I/O使用之外。如果您在8051的外面擴充程式記憶體或資料記憶體時，P2就變成8051的位址匯流排的高位元組(即A8~A15)，此時P2就不能當作I/O使用。P2上的每支I/O腳可推動4個LS TTL。

39~32 腳(P0.0~P0.7)：

這8支腳也是8051的I/O port，稱為P0其中P0.0為LSB，P0.7為MSB。如果將P0當作I/O使用時必須特別注意P0的輸出型態是Open Drain，其他三個I/O port(P1，P2，P3)內部有pull high電路。P0除了當作I/O使用外，如果您在8051的外面擴充程式記憶體或資料記憶體時，P0就當作位址匯流排(A0~A7)和資料匯流排(D0~D7)多工使用。您必須再外部加一個8位元栓鎖器將位址匯流排從PC上分離出來，這個A0~A7與P2所提供的A8~A15合成一個16位元的位址匯流排，因此8051可以在外部定址到64K的記憶體。

29 腳(PSEN)：

這支腳是8051用來讀取放在外部程式記憶體的指令時所用的讀去信號，通常這支腳是接到EPROM的OE腳。8051分別致能放在外部的EPROM(程式記憶體)與RAM資料記憶體是兩塊獨立的記憶體，且這兩塊記憶體都可以接到64K，因此我們說8051的定址能力可達128K。

30 腳(ALE)：

這支腳的名稱為”位址拴住致能” (Address Latch Enable，簡稱ALE)，8051可以使用這支腳觸發外部的8位元栓鎖器，將P0上的位址匯流排信號(A0~A7)鎖入栓鎖器中。

31 腳(EA)：

這是一支輸入腳，當EA=0時，8051一律執行外部程式記憶體裡的程式，因此8051內部的4K程式記憶體就沒有用了。因此如果您要使用內部的程式記憶體時，一定要將EA接+5V。因為8031(或8032)內部沒有程式記憶體，它的EA必須接地。

8051的記憶體結構

8051的記憶體可以分成3塊獨立的記憶體，如圖所示：

1. 內部加上外部的程式記憶體(ROM)共64K byte
2. 可在外部擴充64K byte資料記憶體(RAM)
3. 內部資料記憶體空間256 byte

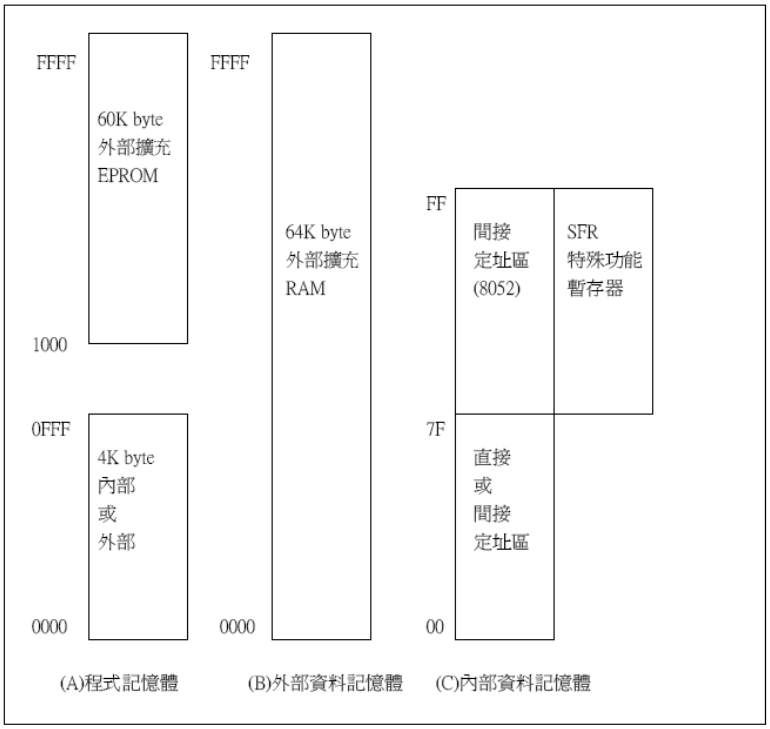


圖2-2-2 8051記憶體映像圖

程式記憶體

程式記憶體是存放8051所執行的程式碼的地方，CPU會主動到這塊記憶體讀取要執行的指令碼，因此這塊記憶體的資料只能被CPU讀取，而無法寫入資料。

程式記憶體的空間最多可達64K byte，在8051，8751裡已有最低的4K byte(0000H ~ 0FFFH)，因此在外部可再擴充60K byte EPROM；而8031H，8032H內部沒有ROM，因此外部可擴充64K byte EPROM；8052AH，8752AH內部已有8K byte的程式記憶體，因此可以在外部擴充56K byte EPROM。8051讀取程式記憶體的激發信號是PSEN。

8051是如何決定程式記憶體的前面4K byte(8052是8K)要到內部或到外接程式記憶體去讀取指令呢？這就是8051的EA腳(第31腳)的功能，如果我們將EA腳接地(邏輯0)，則8051會將前面4K移到外部，也就是說原來在8051內部的4K byte的程式記憶體無效，就算將程式燒到內部的4K byte程式記憶體裡，8051也看不到。

如果將EA接到+5V(邏輯1)，則8051就會到內部去讀去前面4K的程式記憶體，超過4K的部分(1000H~FFFFH)，8051會自動切換到外部來讀取。因此EA接腳是決定內部程式記憶體是否有效的控制腳，當EA=0，內部程式記憶體無效；當EA=1內部程式記憶體有效。例如8031AH，8032AH內部沒有ROM，因此使用8031AH或8032AH時，必須將它的EA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，表2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開始執行程式。

表2-2-2 中斷服務程式的進入點

中 斷 源	向 量 位 址
RESET	0000h
TNT0	0003h
Timer0	000Bh
INT1	000Bh
Timer1	001Bh
UART	0023h
Timer2	002Bh

外部資料記憶體

8051允許您在外部擴充64K byte資料記憶體(RAM)。這64K位址空間裡，除了可以放RAM以外，也可以採用Memory Map I/O的方式將一些標準I/O(例如8255，8253等)的位址解在這一塊記憶體裡。

定址64K資料記憶體空間需要16條位址線和8條資料線，這16條位址匯流排和8條資料匯流排與程式記憶體使用相同的匯流排，然後8051是以控制匯流排來區分這兩塊不同的記憶體。8051讀取外部程式記憶體時使用PSEN，而讀/寫外部資料記憶體使用RD和WR信號。如此一來程式記憶體和資料記憶體就是兩個完全獨立的64K空間。

8051是執行到MOVX A,@DPTR和MOVX A,@Ri 指令時，就會到外部資料記憶體讀入一個byte資料，當執行MOVX @DPTR,A 或 MOVX @Ri,A 就會將資料寫到外部資料記憶體。

有時候在外部擴充程式記憶體和資料記憶體，其總和不超過64K時，我們可以採用兩塊記憶體合併成一個64K的設計方式，合併的好處是可以讓程式設計更具彈性。

合併的方法很容易，因為8051將程式記憶體和資料記憶體分開的方法是將這兩塊記憶體的讀取激發信號分別使用不同的信號，即PSEN讀取程式記憶體，RD讀取資料記憶體，因此要將這兩塊記憶體合併，只要將PSEN及RD信號合併成一個信號即可，方法是將PSEN與RD使用AND閘做邏輯AND即可，如圖所示，可將AND閘的輸出看成一個記憶體讀取激發信號(MRD)激發(MRD=0)，然後我們就將MRD接到程式記憶體(EPROM)的輸出致能，或資料記憶體(RAM)的輸出致能就可以讀到這兩塊記憶體的內容。

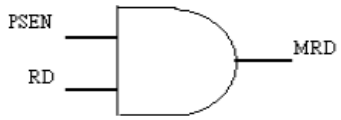


圖2-2-3 將PSEN與RD合併成MRD

程式記憶體與資料記憶體合併之後，8051的整個記憶體空間就縮減成64K，也可以使用合併外接32K EPROM (27256)和32K SRAM (62256)的方法。在這種合併的記憶體結構裡就沒有所謂程式記憶體或資料記憶體之分別，不管是MOVC或MOVX指令都可以定址到這64K的內容。換句話說，也可以將程式放入RAM(62256)裡執行。

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

不論8051或8052都有這塊記憶體，並且可以使用直接定址或間接定址法讀/寫其內部資料。8051將這塊記憶體分成數種不同的用途。

(A) 暫存器庫

由上圖我們看到，位址00~1FH這32個byte被分成4組工作暫存器(Register Bank)，分別稱為RB0，RB1，RB2和RB3，每一組暫存器庫有8個byte。程式指令將每組裡的8個byte稱為R0~R7。但是8051共有四組R0~R7，到底目前所指的R0~R7是屬於哪一組的R0~R7呢？它是由PSW暫存器裡的RS1和RS0 這兩個bit加以選擇，如上圖所示，當RS1=0和RS0=0時，就指到RB0，當RS1=0和RS0=1，就指到RB1。

(B) 可位元定址區

位址20H~2FH這16個byte是8051內部256個位元位址中的128個位元的所在位址。每個位元組佔了8個位元位址，下圖是這16個位元組裡每一個bit的位元位址，例如20H這個byte的第0位元，其位元位址為00H，然後依序編到2FH這個位元組的第7位元為7FH，8051有一組單位元運算指令可以直接對這些位元作運算。

位元位址區的另外128個bit(80H~FFH)，是在SFR暫存器裡，如下圖所示。

(C) 一般用途

內部RAM的30H~7FH這些位元組，8051並未定義這些位元作任何用途，8051稱這塊區域為使用者的RAM(User RAM)。因此可以規劃這塊區域當作其他用途，例如計時器的緩衝區或印表機資料的緩衝區等。但是有一點必須注意的是，8051的堆疊區也是使用內部RAM，因此必須保留一塊足夠大的RAM給堆疊區使用，堆疊區的大小是依所寫的程式所需而定。

表2-2-3 8051的前面128個位元位址

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								

1.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。

2.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。

在8051被重置後(RESET=1)，在SFR裡面的各個暫存器都會被設定一個固定值，這些僅在每次RESET後都是一樣。

SFR裡面的各個暫存器：

1. PSW(程式狀態字語)暫存器
2. SP暫存器(堆疊指標)
3. DPTR(資料指標)
4. P0，P1，P2，P3暫存器
5. SBUF(串列阜緩衝區)
6. 計時器暫存器
7. 捕捉式暫存器(Capture Register)
8. 控制暫存器(Control Register)

表2-2-4 SFR各個暫存器重置後的初始值

暫存器名稱	以二進制表示之值
*ACC	00000000
*B	00000000
*PSW	00000000
SP	00000111
DPTR：	
DPH	00000000
DPL	00000000
*P0	11111111
*P1	11111111
*P2	11111111
*P3	11111111
*IP	8051 XXX00000
	8052 XXX00000
*IE	8051 0XX00000
	8052 0X000000
TMOD	00000000
*TCON	00000000
*+T2CON	00000000
TH0	00000000
TL0	00000000
TH1	00000000
TL1	00000000
+TH2	00000000
+TL2	00000000
+RCAP2H	00000000
+RCAP2L	00000000
*SCON	00000000
SBUF	未定
PCON	HMOS 0XXXXXXX
	CHMOS 0XXX0000
X = 未定	
* = 可位元定址	
+ = 只 8052 有	

低通濾波電路

由C9、C12、R10、R11構成低通濾波電路，首先由C9濾去直流成分，然後由C12、R10、R11構成RC濾波電路，提取低頻訊號成分。本設計的設號頻率為W=1/RC=1/105X2.2X10-6=4.5HZ。也就是說頻率為4.5HZ左右的低頻訊號可以通過，而人的心跳最高為4HZ，滿足了低通的需求。

其心跳訊號經過4069反相放大後輸出到8051的P1.7 腳位上。

放大電路

COMS反相器在高低電位轉換之間,即載止與飽和之間，有一過渡區,這一段過渡區就是放大區(線性區)，利用這一區域,可將COMS反相器作放大器之用。使用方法：將大電阻接至反相器的輸入與輸出之間作為直流偏置電阻，這時，反相器工作於放大區.就可作放大器用了。一般情況下，一級放大倍數可達20多倍。

心跳訊號經過3組COMS反相器放大後得到約8000倍的放大率。

此3組為CD4069第1到6腳、C3、C5、C7、C10、C13、R4、R8、R14

整形電路

由三個反相器組成的施密特觸發器，輸入電壓經電阻R12和R9分壓後來控制反相器的工作狀態。當輸入電壓上升到略大於正向電壓或下降到略小於負向電壓時，施密特觸發器的狀態才會迅速翻轉，而輸出方波波型如下圖所示：

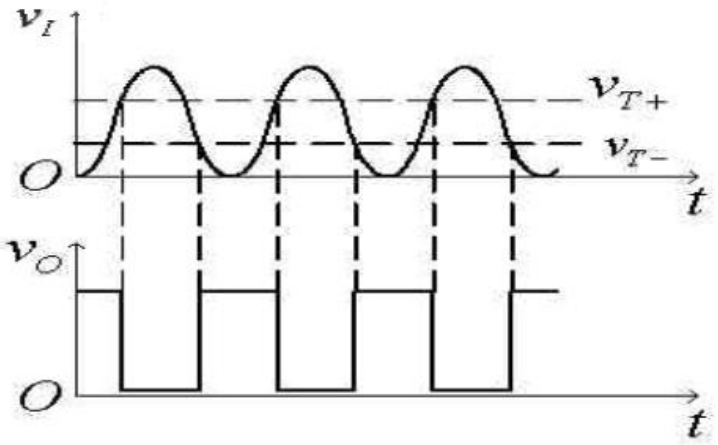


圖2-2-4 整形電路

心跳程式計算：

假設1次心跳間格為0.5秒->1秒為2次心跳->60秒為120次心跳

得出公式：1次心跳t秒->1秒為1/t次心跳->60秒為60/t次心跳

使用8051計時中斷取1ms中斷一次，若心跳間格為n次中斷，可得出60秒心跳為60000/(n)

參、專題製作

本章分成三個階段由說明此專題所應用到的設備和器材及製作方法與步驟專題製作等等。

一、設備及材料

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
9V 電池	供應所需電源
麵包版	模擬電路
三用電錶	檢測
PC 電腦	編譯程式
燒錄器	程式燒錄 89S51
Word	書面報告
Powerpoint	上台報告之簡報
示波器	觀察波行，找出錯誤
程式編譯器	編好之軟體除錯即轉燒錄檔
數位相機	拍照以用圖片說明

表3-1-2 脈搏感測器之材料表

項目	數量	編號	規格	
1	1	C1	220uF	電解電容
2	2	C11,C2	104	陶瓷電容
3	3	C3,C7,C13	473	陶瓷電容
4	2	C4,C6	30PF	陶瓷電容
5	3	C5,C8,C10	10uF/25V	電解電容
6	1	C9	47uF	電解電容
7	1	C12	2.2uF	電解電容
8	1	D1	LED(紅)	
9	1	J1	9V 變壓器插座	
10	2	LED1,LED2	七段顯示器	
11	3	R1,R2,R3	300	電阻
12	3	R4,R8,R14	1M	電阻
13	1	R5	1K	電阻
14	1	R6	100	電阻
15	1	R7	560	電阻
16	1	R9	3.9M	電阻
17	1	R10	100K	電阻
18	2	R13,R11	10K	電阻
19	1	R12	220K	電阻
20	1	SW1	POWER	電源開關
21	2	U2,U1	74LS47	七段顯示 IC
22	1	U3	7805	5V 穩壓 IC
23	1	U4	89S51	單晶片
24	1	U5	4069	反相 IC
25	1	U6	CNY70	光感測器
26	1	Y1	12MHZ	振盪器

二、製作方法和製作步驟

這次專題利用的是行動並且研究，由循環的反覆研究及製作過程所構成，包含準備零件、實驗模擬、電路資料查詢及報告篇寫等階段。此研究之製作方法與製作步驟。

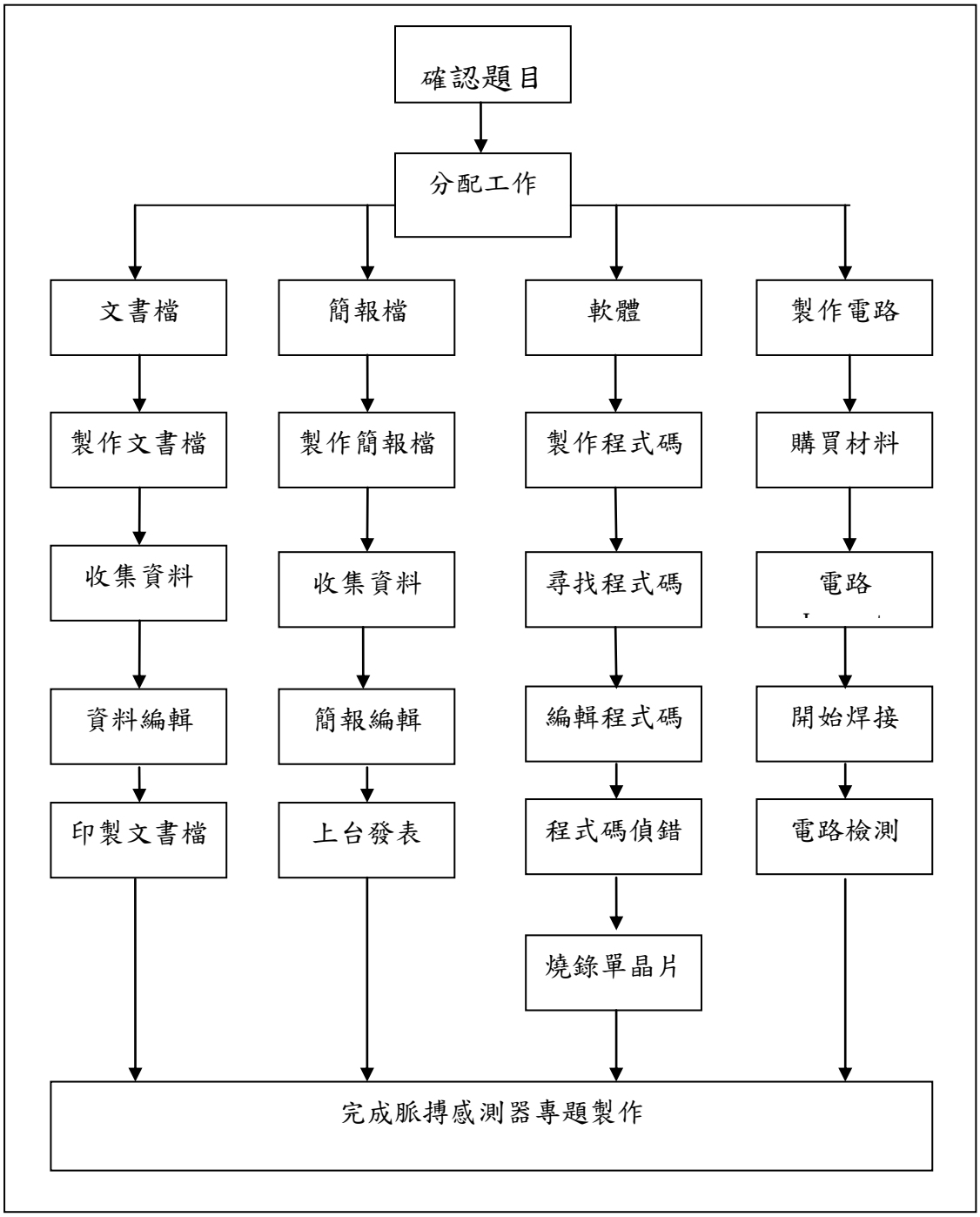


圖 3-2-1 製作方法及步驟

三、專題製作

表 3-3-1 專題製作計畫書

<table>
<tr><td colspan="2">專題型別</td><td colspan="2">□ 個人型專題 ▌ 團隊型專題</td></tr>
<tr><td colspan="2">專題性質</td><td colspan="2">單晶片控制研究</td></tr>
<tr><td colspan="2">科別／年級</td><td colspan="2">電子科三年級</td></tr>
<tr><td rowspan="2">專題名稱</td><td>中文名稱</td><td colspan="2">脈搏感測器</td></tr>
<tr><td>英文名稱</td><td colspan="2">Pulse Sensor</td></tr>
<tr><td colspan="2">專題內容簡述</td><td colspan="2">本專題藉脈搏感測輸入給 89S51 脈搏的資訊，89S51 並依此訊息做出顯示。脈搏感測是藉由紅外線感測元件來偵測脈搏，從微血管感應脈搏，達成偵測脈搏的效果，脈搏資訊將經由 89S51 程式進行處理計算，再由七段顯示器來呈現使用者的脈搏，這樣便結合了人體的建康資訊，踏出美好生活的第一步。
隨著工作步調加快以及工時的加長，工作壓力與日俱增，因工作過度勞累而發生猝死的意外時有所聞。</td></tr>
<tr><td colspan="2">指導老師姓名</td><td colspan="2">葉忠賢 老師</td></tr>
<tr><td colspan="2" rowspan="2">參與同學姓名</td><td>陳冠伯組長</td><td>林品傑組員</td></tr>
<tr><td>李昕頤組員</td><td>吳威承組員</td></tr>
<tr><td colspan="2">專題執行日期</td><td colspan="2">103 年 9 月 1 日至 104 年 5 月 31 日</td></tr>
</table>

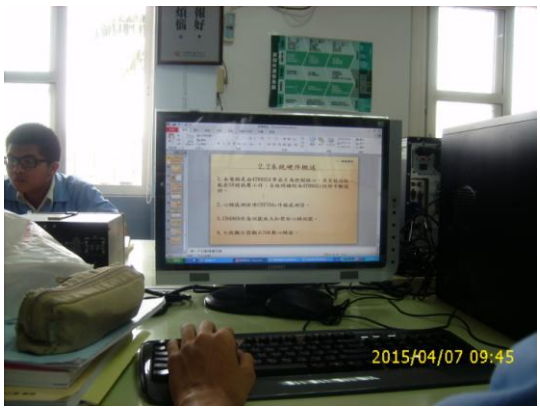
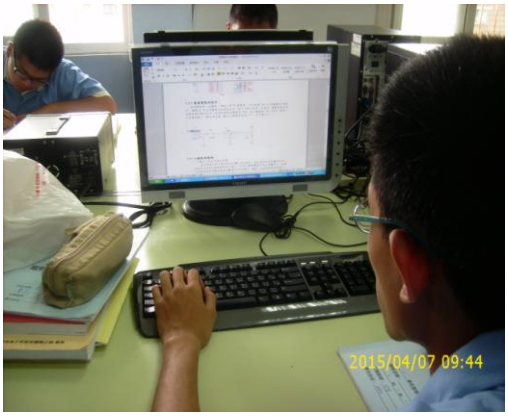
	

圖 3-3-1 專題製作的簡報	圖 3-3-2 專題製作的 WORD

(一) 脈搏感測器功能及設定

透過這個專題製作，去了解如何運用單晶片在紅外線自動控制上，當在使用脈搏感測器時，能進行遠距離感測，且經由透過程式碼，來更改紅外線距離，使脈搏感測器的表示方法：剛開始脈搏感測器送電時，七段顯示器會先作用然而進而感測。

本專題的內容由89S51為基礎的做電路架構，其中功能、設定：

1. 脈搏感測器設定：

(1) 送電時，LED燈會先啟動。

(2) LED燈會先熄滅，開始讀取程式及光感測器

(3) 當手指放在紅外線感測器，七段顯示器會直接亮數字。

(4) 程式會讀取電路當中所設定紅外線感測。

2. 脈搏感測器操作：

(1) 透過紅外線發送接收器，了解紅外線作用距離。

(2) 透過89S51之接腳該如何運作；程式如何進行更改的方法。

(二) 硬體電路圖：脈搏感測器

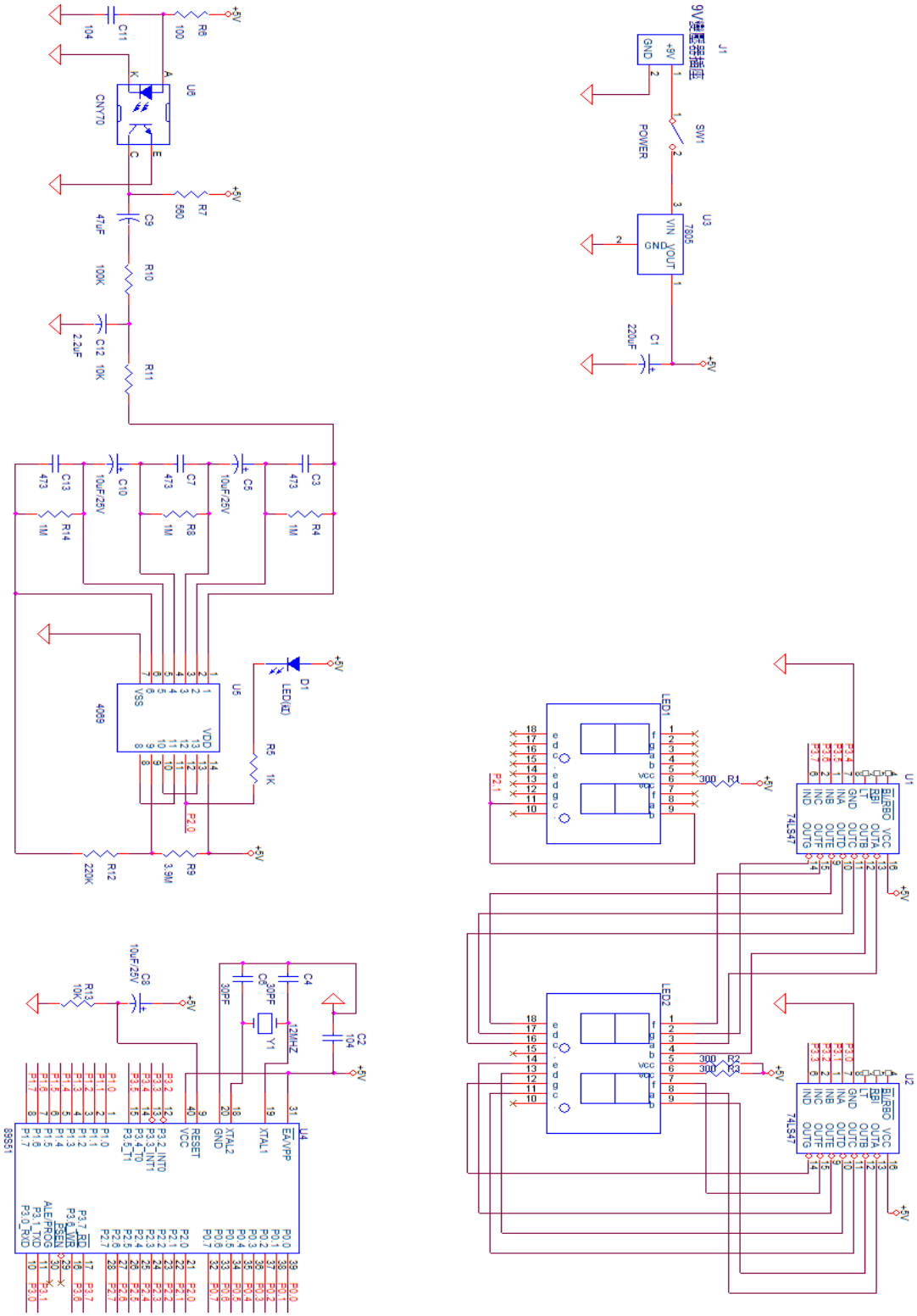


圖 3-3-3 脈搏感測器電路圖

由85C51中撰寫程式碼使，電路中的紅外線發射器送出紅外線，經由阻擋紅外線之物體，使紅外線接收器接收讓七段顯示器產生作用，當阻擋物體離開七段顯示器立即停止功能。

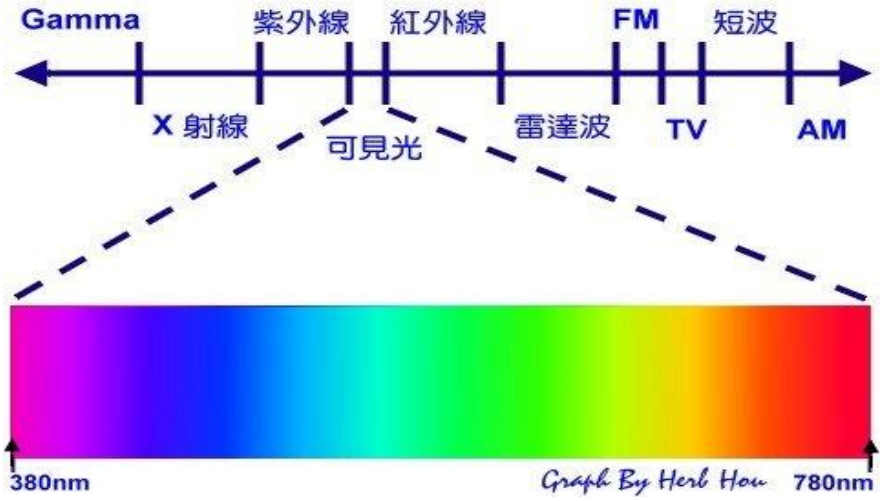


圖3-3-4紅外線原理作用圖

在高雄的各電子材料行都難以買的到這些東西，經由我們小組討論後去長民街買材料結果都要等一個禮拜，一個禮拜買完之後就開始製做我們這組的專題製作，脈搏感測器是發出肉眼所看不到的東西、發射出去後由紅外線接收器接收就能達成你所要的目的了。

小組分工的配置：

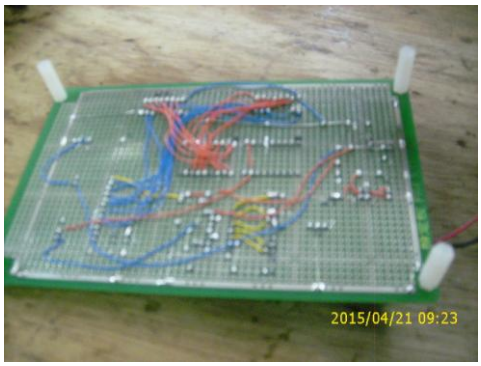
昕頤跟冠伯負責找資料、製作簡報內容，品傑負責製作文書檔，過程中亦有購買相關書籍當成參考資料，選擇要如何去製作專題，讓昕頤和冠伯知道要作何專題，然後再經過小組討論、商量，有問題時，會再去徵詢老師的意見。

冠伯是要上網查有關專題相關的資料，負責把相關的資料和圖片下載下來，讓品傑製作文書軟體得作業，如有發生困難時可以找小組討論或者找老師諮詢，解決不懂的是事情。

昕頤和威承負責去買電路中所需之零件，來製作我們這組的專題報告書家維負責把Layout圖畫出來，在讓品傑看著Layout圖去和接我們的專題製作的成品，昕頤負責製作簡報讓威承和上台發表此專題內容。

肆、製作成果

我們小組決定題目脈搏感測器，開始製作並且模擬麵包板、設計Layout圖，進而去完成焊接且製作整個完整電路；這整個流程中，我們小組都用相機及相關電腦設備將之拍照下來，經將這些資料完整處理過後，我們呈現在這個專題報告之中，如下所示：

	
圖4-1-1 脈搏感測器 製作過程(一)	圖4-1-2 脈搏感測器 製作過程(二)
圖4-1-3脈搏感測器成品圖	圖4-1-4 脈搏感測器 測試功能

目前為止，我們所製作的電路實作部分，面對我們所遭遇的困難以及問題，都已克服了，緊接著，我們的下一步驟即是將我們開始製作外面包裝的成品製作，將之接到我們的，由於有牽涉到木工配置的問題，所以待我們將之完成後，會再以照片及實體的方式慢慢呈現之。

89S51單晶片，進行程式運算

脈搏顯示在 LED 七段顯示器

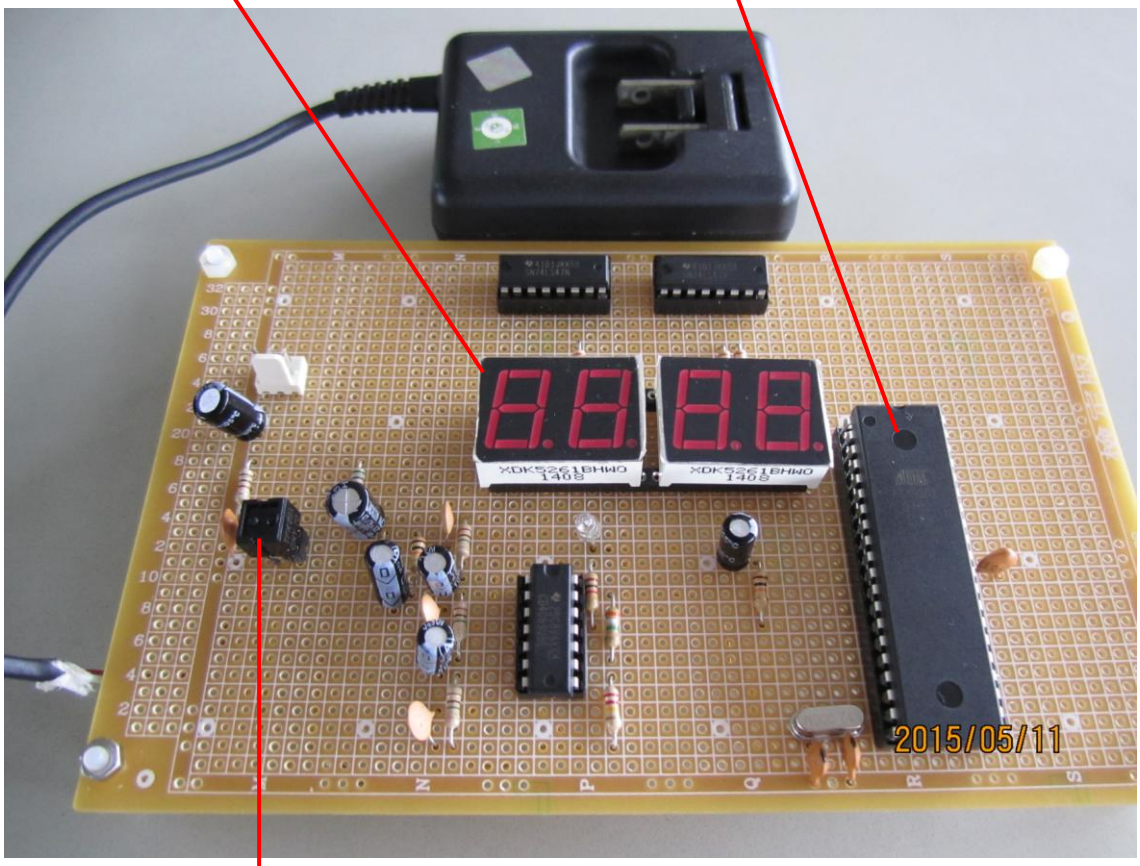


圖4-1-5 脈搏感測器 功能介紹

把手指放在紅外線 CNY70 感測器上。(一秒碰一次)

伍、結論與建議

本次的單元使我們將對整體我們小組來說是一個專題製作的學習過程，做一整個過程到最後完整的記錄以及報告，作為在未來學弟學妹們日後學習及參考。

一、結論

1. 學校裡所學的皆以理論為主，將理論結合電路應用在日常生活中將有效提昇學習興趣。
2. 自己動手做專題，親手製作所得到的東西最多，印象也很深刻。
3. 專題製作可培養團隊合作的精神，因彼此都會有自己的意見。
4. 專題的製作可以帶來很大的成就感，當完成專題時的那份喜悅只有認真負責的人能感受的到。
5. 專業知識的不足是一般專題製作過程中常遇到的事，此時老師所扮演的角色就相對重要。
6. 資料收集不易，因往往不知從何著手，面對的是一個大電路，電路元件皆複雜，往往會因此而失去方向。

二、建議

在學習過程中給予回饋：同學建議或老師的想法也可以聽組任的評論，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：

1. 進行感測的時候只能一秒碰一次，希望我們以後可以研究出把手指放在感測器上就可以有動作，就不用一直一秒碰一次了。
2. 脈搏感測器的精準度比較不正確，數據值誤差很大，希望我們以後可以研究如何改成很正確的數據，就不會因為別個影響而導是數據值誤差很大。
3. 如果手沒有放上去，LED七段顯示器會自己跑出數字，希望我們以後可以研究出能夠改變成用手指頭放在感測器上會正常跳動，就不會因為亂掉的機會。

4. 這個專題是我們手動做出來，會有很多缺點，但要給我們很多時間與研究或者建議，我們一定會克服一切的困難，這就是我們的學習方式。

參考文獻

1. 楊明峰，2008，8051 入門輕鬆學，台北市：碁峰資訊。
2. 鍾富昭，2003，8051 專題製作，台北市：全華文化。
3. 陳俊榮，2007，組合語言，台北市：全華文化。
4. 林豐隆，2007，專題製作，台北市：全華文化。
5. 傅榮鈞・林偉政，專題製作8051單晶片篇，台北縣：台科大圖書。
6. 徐椿樑・陳輔賢，2004，8051/8951理論與實物應用，台北市：全華文化。
7. 蔡朝洋單晶片微電腦8051 / 8951原理與應用。台北市：全華科技圖書股份有限公司。
8. 郭庭吉、吳金戊（2008），單晶片8051 專題製作-使用keil AX51。台北市：松崗電腦圖書有限公司。
9. 許桂樹、范逸之Visual Basic-網路監控實務。台北縣：新文京開發出版股份有限公司。
10. 林文恭（2007）。電腦硬體裝修乙級檢定學術科。台北市：碁峰資訊股份有限公司。
11. 陳永達、詹可文（2004）。VB 串並列埠控制。台北市：全華科技圖書股份有限公司。
12. 黃宏彥、余文俊、楊國輝編著，感測器原理與應用電路實習，高立圖書有限公司，1999。
13. 蔡朝陽，電工實習（四），全華科技圖書股份有限公司，中華民國八十三年五月。
14. 鄧錦城編著，8051單晶片實作寶典，益眾資訊有限公司，2000。

附錄一 脈搏感測器程式碼

```
        ORG     00H         ;程式從 00H 開始
    AJMP    START           ;跳到 START
        ORG     0BH             ;設定 Timer 0 中斷程式起始位置
        AJMP    TIMER0              ;中斷發生後，到 Timer 0 副程式執行
;----- -
; MAIN
;----- -
START:
        MOV     TMOD,#01H           ;設定 TIMER O 工作模式 1
        MOV     IE,#10000010B       ;計時中斷設定
        MOV     TH0,#>(65536-10000)     ;計時中斷設定
        MOV     TL0,#<(65536-10000)     ;計時中斷設定
    SETB    TR0     ;計時中斷設定
    CLRET0      ;計時中斷設定
    MOV     30H,#0          ;30H 設為 0
    MOV     31H,#0          ;31H 設為 0
    MOV     40H,#0          ;40H 設為 0
;==========================
;========主程式迴圈========
;==========================
LOOP:
    JB  P2.0,LOOP   ;等待心跳，有則程式往下，無則跳 LOOP
    MOV     31H,#0          ;31H 設為 0 初值
    SETB    ET0     ;開始計時中斷
WAIT:
    MOV     A,30H
    CJNE    A,#1,WAIT   ;30H 初值為 0，當為 1 時表示下一心跳已到，程式往下，否則跳 WAIT
```

```
    CLRET0      ;停止計時中斷
    MOV     30H,#0          ;31H 設為 0 初值
    CALL    CONVERT         ;將 40H 所累加的值 轉成心跳值給七段可以顯示的值
    MOV     40H,#0          ;清除 40H
    JMP LOOP            ;跳到 LOOP

;=========6000/(心跳間隔)================
;====舉例：6000/55=109(心跳數/秒)========
;===心跳間隔的單位為 0.01 秒===============
;===計時中斷=0.002 秒=====================
CONVERT:
    PUSH    A
    MOV     52H,#0
    MOV     5FH,#70H    ;6000 分兩個地方存
    MOV     5EH,#17H    ;6000=1770(16 進位)
        MOV     5BH,5FH         ;5FH->5BH
CONV_T0_RAM_L:
        CLR     C       ;C = 0
    MOV     A,5BH
    MOV     B,40H           ;除法用減的 6000-心跳間格 看減了幾次 6000 被減到 0 時結束
        SUBB    A,B
        MOV     5BH,A
        JNC     CONV_T0_RAM_1
        CLR     C
        DEC     5EH
        MOV     A,5EH
        CJNE    A,#FFH,CONV_T0_RAM_1
        LJMP    CONV_T0_RAM_X
```

```
CONV_T0_RAM_1:
    MOV     5CH,A
CONV_T0_RAM_2:
        INC     52H                 ;減了 1 次 52H 加 1 52H 就是商數
    MOV     A,52H
        CJNE    A,#FFH,CONV_T0_RAM_L        ;52H 加到 FFH 時 表示錯
誤 FF=255
        MOV     52H,#00H                        ;52H 變 0
CONV_T0_RAM_X:
    CLRC
    MOV     A,52H
    MOV     B,#100
    SUBB    A,B
    JC  NOVER
    CLRP2.1
    JMP OVV
NOVER:
    SETB    P2.1
OVV:
    MOV     A,52H       ;取出 52H 的十、個位數顯示在七段
    MOV     B,#100      ;B=100
    DIV AB      ;A/B
    MOV     A,B     ;餘數 B->A
    MOV     B,#10       ;B=10
    DIV AB      ;A/B
    SWAP    A       ;A 為十位數
    ADD     A,B     ;A 加個位數
    MOV     P3,A        ;將 A 送到 P0 作顯示
    POP A
        RET
```

```
;----- ---
; TIMER0
;----- ---
TIMER0: PUSH      A
        MOV       TH0,#>(65536-10000)
        MOV       TL0,#<(65536-10000)
    MOV     A,31H          ;心跳感測到後，31H 已被設為 0
    CJNE    A,#0,EX2       ;為 0 時程式往下，不為 0 則跳 EX2
    JNB P2.0,EX1;等待心跳訊號變 HIGH 時程式往下，否則跳 EX1
    MOV     31H,#1         ;31H 設為 1
EX1:
    INC 40H       ;40H 加 1
    JMP EXIT3
EX2:
    JB   P2.0,EX3;等待心跳訊號變 LOW 時程式往下，否則跳 EX3
    MOV     30H,#1         ;30H 設為 1
    JMP EXIT3
EX3:
    INC 40H       ;40H 加 1
EXIT3:  POP       A
        RETI

;     ======================
;     ======延時副程式======
;     ======================
DELAY2:
DD0:    MOV R6,#30
DD1:    MOV R7,#30
DD2:    DJNZ R7,DD2
    DJNZ R6,DD1
```

```
DJNZ R5,DD0
RET
    END
```