

修平技術學院四年制機械工程系

專題製作報告

投籃機

指導老師：歐乃瑞

班 級：四機四乙

組 長：林明仕 BA95131

組 員：謝靖偉 BA95117

張嘉榮 BA94014

中 華 民 國 九 十 八 年 十 二 月 二 十

一、導論

1.1 動機

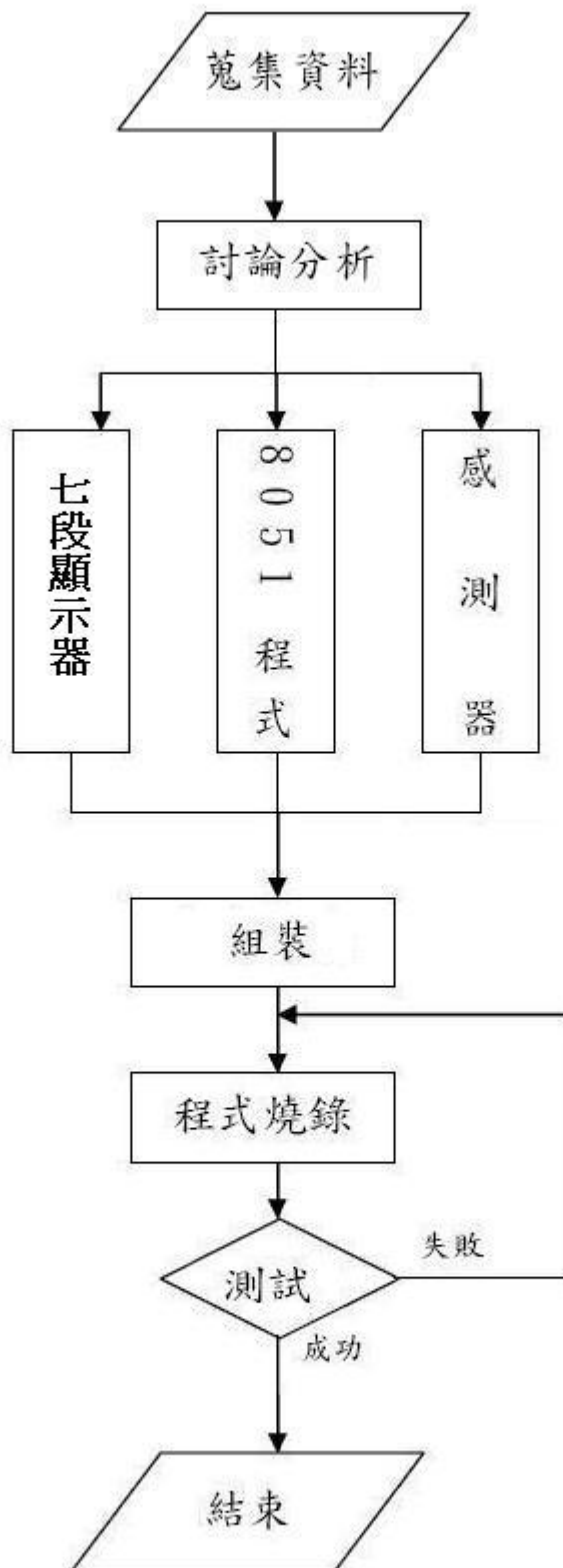
此專題讓我們這組傷破了腦筋，但就在某一天下午我們正在熱血打球的同時，便突發奇想的想到既然我們這麼熱愛打籃球，那不如就做投籃機好了，這一來是我們三個人共同的興趣，二來投籃機也可以達到健身的效果，真是一舉兩得，於是在我們討論下訂下了我們此次專題的題目。

確定了此專題的題目後，我們就到市面上所現有的哈日龍店去收集與觀察期購造，再經上網查輯資料並發現投籃機上的記分器是由單晶片結合紅外線感應器再由顯示器所構成的！於是這單晶片也成為我們最大的阻礙，也藉由此投籃機來探討單晶片的功能。

1.2 摘要

89C51 單晶片是目前市面上很受歡迎使用的單晶片微處理器之一，是由INTEL 公司所開發出來，普遍地應用在工業界中。由於其使用的普及，許多設計半導體晶片的公司也有製造與89C51 相容的單晶片，例如由ATMEL 公司所生產製造的89C51 單晶片便與INTEL 公司的89C51 完全相容，其間最大的不同是89C51 是可以重複燒錄的，而8051 則否。這也是為什麼我們會選用89C51 單晶片為我們專題製作的核心，就科技的腳步來說，89C51 發展歷史已經相當久遠，可以重複燒錄使用相當方便，技術也已經純熟目前的89C51 單晶片大都可以重複燒錄1000 次以上。

1.3 製作流程

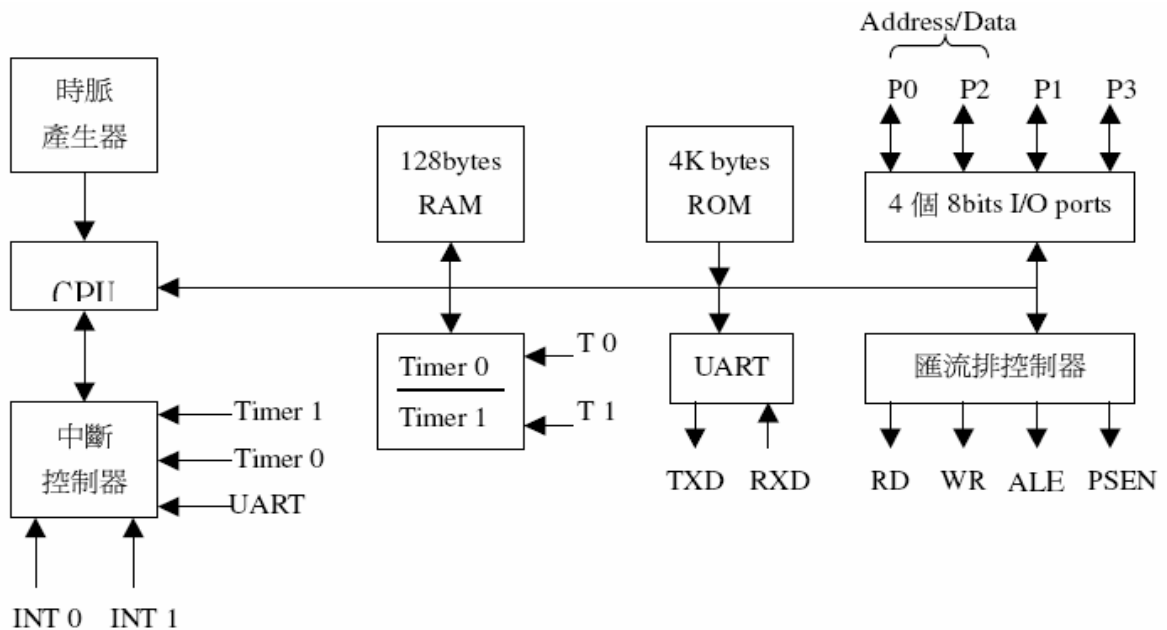


二、8051單晶片介紹

微電腦普遍應用在日常生活中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。

2-1 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：



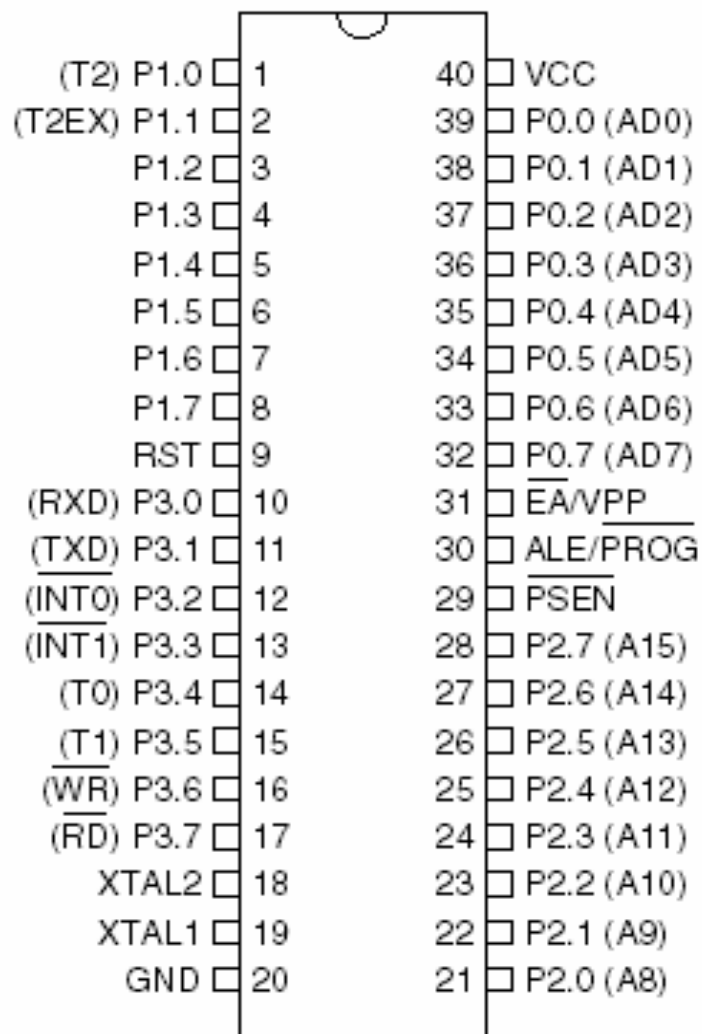
8051 主要功能列舉如下：

- 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

2-2 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：



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。

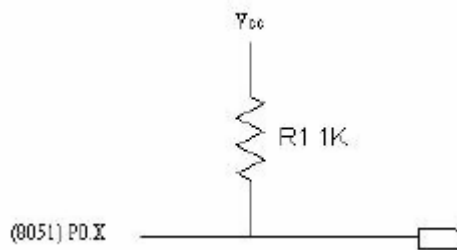
◇ EA=1 時，系統會存取內部記憶體中的程式碼。

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：



◇ 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

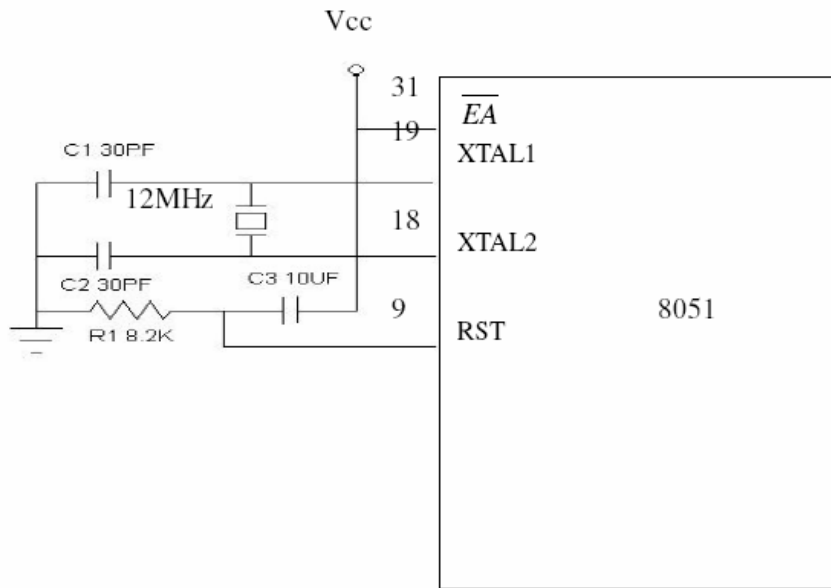
器0 輸入。 P3.5：T1，計時計數器

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。

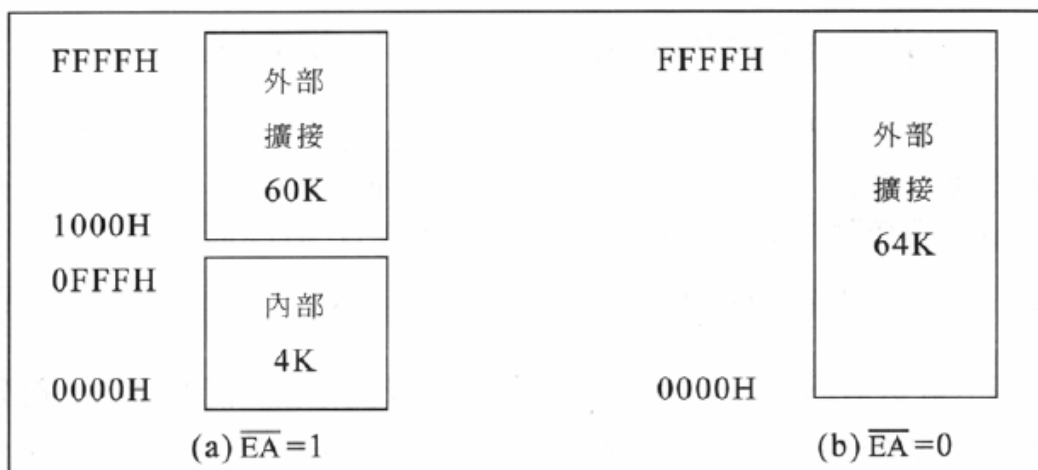
基本電路連接如下圖：



2-3 8051 的記憶體結構

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：



2-3.1 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 開始執行程式。

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

2-3.2 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

1. 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

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。

2.4 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 埠的輸出門鎖(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：

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。

(5) 計時器控制(Timer Control, TCON)暫存器：

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：

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

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。

SM0	SM1	模式	說明	鮑率
0	0	0	移位暫存器	Fosc./12
0	1	1	8-bit UATR	可變
1	0	2	9-bit UATR	Fosc./64 或 Fosc./32
1	1	3	9-bit UATR	可變

(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。

下圖為SFR 的記憶體分佈圖：

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

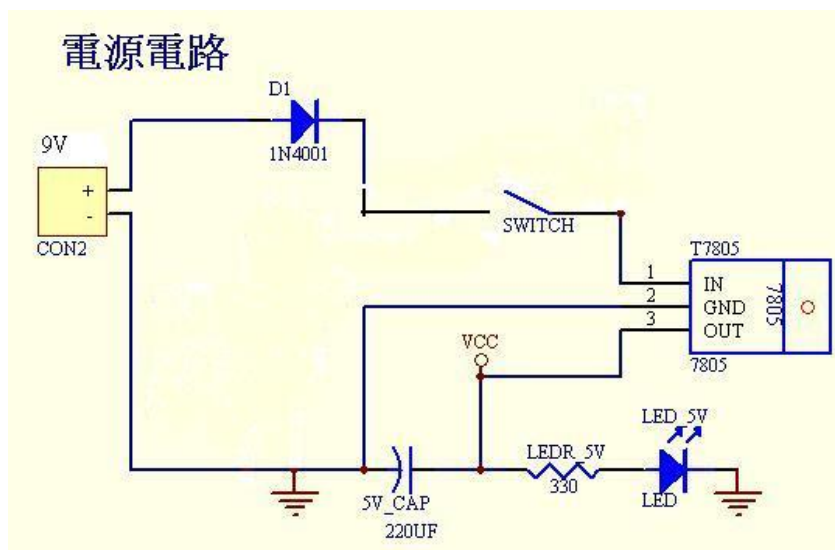
↑ 此行記憶體位元組可做位元定址。

三、電路說明

電路總和

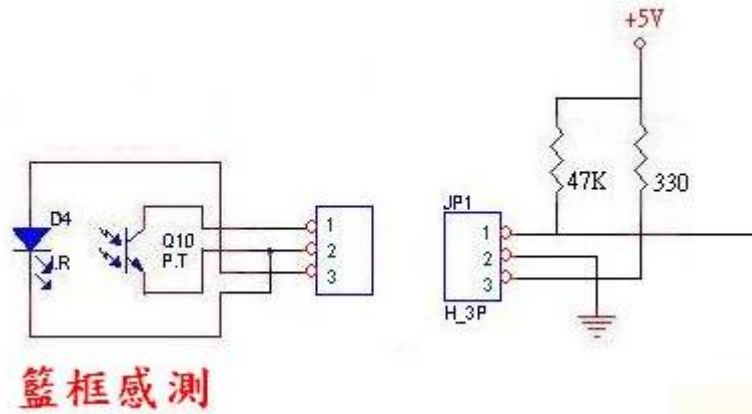
本專題所使用的電路包含了 89C51 主電路板，5V 穩壓電源電路，紅外線感測電路，七段顯示器電路。

3.1 電源電路



提供 CPU 主板得電源電路，它的主要目的是提供一個穩定的 5V 電壓，LM7805 為穩壓 IC，各有三隻腳，分別是輸入端、共同(接地)端及輸出端，它的功用是穩定電壓，另外要注意的是，由於通過它的電流相當大，所以一定要加裝散熱片。

3.2 紅外線感測器電路



紅外線感測電路，因發光二極體所發射的紅外線經光電晶接收，光電晶體飽和，射極電壓為高態；當有障礙物阻擋時，發光二極體所發射的紅外線無法反射至光電晶體，光電晶幾乎截止，射極電壓為低態，傳送給 8051 運作。

四、程式架構

```
MODE EQU 10H ;計時計數器模式(Mode 1)
COUNT EQU -50000 ;計數量
TIMES EQU 20 ;重複次數
DISP_DATA EQU 30H ;顯示器位址 30H~32H
DISP REG P0 ;七節顯示器(計時)
DISP2 REG P1 ;七節顯示器(個位十位分數)
DISP3 REG P2 ;七節顯示器(百位分數)
DIG0 REG 20H ;個位數數字
DIG1 REG 21H ;十位數數字
;=====使用中斷方式=====
ORG 0 ;程式從 0 位址開始
JMP START ;跳至 START
ORG 1BH ;設定中斷向量
JMP TIMER1 ;執行中斷副程式
START: MOV DISP, #FFH ;關閉七節顯示器
SETB EA ;打開中斷總開關
SETB ET1 ;打開 Timer 1 中斷開關
MOV TMOD, #MODE ;設定計時計數器模式
MOV SP, #70H ;移開堆疊位置
LOOP: MOV R1, #60 ;設定七節顯示器初始數字
MOV R0, #0 ;設定已重複次數為 0
MOV R3, #61 ;設定計時 60 秒
MOV DISP, #FFH ;關閉數字
MOV TH1, #>COUNT ;設定計數量
MOV TL1, #<COUNT ;設定計數量
;===== (主程式功能) =====
MAIN:
MOV DISP, #3 ;七節顯示器顯示 3
CALL DELAY
MOV DISP, #2 ;七節顯示器顯示 2
CALL DELAY
MOV DISP, #1 ;七節顯示器顯示 1
CALL DELAY
MOV DISP, #0 ;七節顯示器顯示 0
CALL DELAY
SETB TR1 ;啟動 Timer 1
```

```

CALL    DELAY

MOV     R2, #0H ;
L_LOOP: MOV  A, R2

MOV     B, #100
DIV     AB
MOV     DISP_DATA, A ;百位數
MOV     A, B
MOV     B, #10
DIV     AB
SWAP   A
ADD    A, B
MOV     DISP2, A ;輸出到 DISP2
MOV     DISP3, DISP_DATA ;輸出到 DISP3

JB  P3.5, $
CALL  D20MS
JNB P3.5, $
CALL  D20MS
INC  R2
INC  R2

JMP   L_LOOP ;跳至 L_LOOP 形成一個迴圈
;=====主程式結束=====
;=====60 秒計時(中斷副程式開始)=====
TIMER1: PUSH  PSW ;儲存程式狀態字組暫存器
        PUSH  A ;儲存 ACC
        INC   R0 ;R0+1(重複次數增加 1 次)
        CJNE  R0, #TIMES, AGAIN ;是否已重複 20 次
;-----1 秒鐘-----
        MOV   A, R1 ;取回顯示數字

MOV  B, #10
DIV  AB
MOV  DIG0, B

```

```

MOV B, #10
DIV AB
MOV DIG1, B
MOV A, DIG1
SWAP A
ADD A, DIG0

```

```

MOV DISP, A ;顯示數字
DEC R1 ;數字減 1
DJNZ R3, NEXT ;是否已顯示 60 秒

```

AA:

```

MOV DISP, #0
JMP AA

```

```

NEXT: MOV R0, #0 ;重新計數下一秒鐘
AGAIN: MOV TH1, #>COUNT ;設定計數量
MOV TL1, #<COUNT ;設定計數量
SETB TR1 ;啟動 Timer 1
POP A ;取回 ACC
POP PSW ;取回程式狀態字組暫存器
RETI

```

;=====60 秒計時(中斷副程式結束)=====

;=====DELAY 副程式開始(延遲一秒)=====

DELAY:

```
MOV R5, #10
```

L1:

```
MOV R6, #200
```

L2:

```
MOV R7, #250
```

L3:

```
DJNZ R7, L3
```

```
DJNZ R6, L2
```

```
DJNZ R5, L1
```

```
RET
```

;=====DELAY 副程式結束=====

;=====D20MS 副程式開始=====

D20MS:

MOV R6, #40

DD1: MOV R7, #248

DJNZ R7, \$

DJNZ R6, DD1

RET

;=====D20MS 副程式結束=====

END