

微介實驗九

ADC的轉換應用

日期:2024/12/3

報告者:許宸華

Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

學習重點

- 了解ADC轉換訊號的原理和轉換方式
- 了解ADC0804轉換IC的使用方法
- 比較Polling和Interrupt轉換方式的差別

Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

實驗內容

- 利用可變電阻輸入電壓分壓給ADC0804轉為數位訊號，8051接收後判定是否超過電壓門檻(設為輸入最大輸入類比電壓的一半)
 - ， 若超過則LED燈全亮。

Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

材料清單

器材名稱		數量
AT89S51		1
12MHz 石英震盪器		1
ADC0804		1
可變電阻 100K		1
電阻	1k Ω	9
	10k Ω	1
電容	20pF	2
	151pF	1
	10 μ F	1

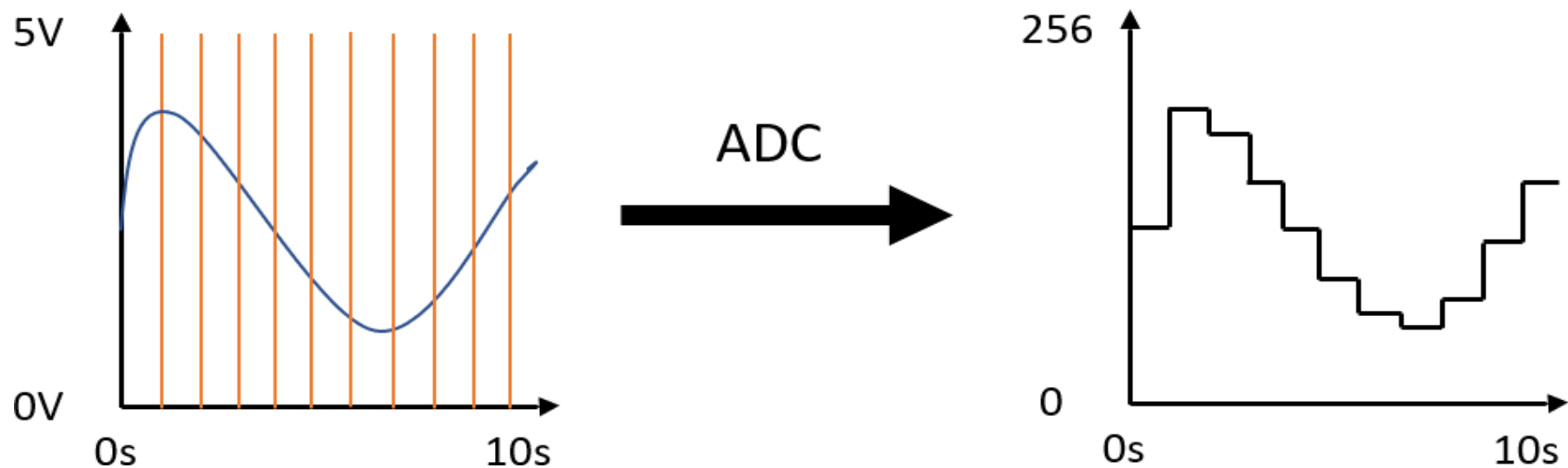
Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

ADC轉換訊號的原理

元件原理 — 類比數位轉換原理

- 因為8051只能接收數位訊號，所以在我們要讀取類比訊號時，必須先經由ADC(Analog-to-digital converter)類比-數位轉換器的轉換後，我們才可將類比訊號轉為數位訊號讓8051讀取。

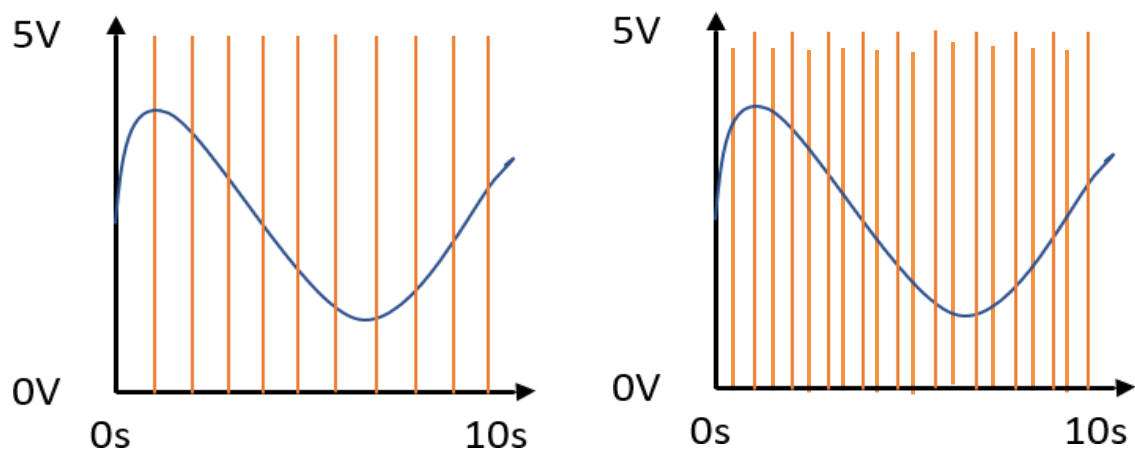


元件原理 — 類比數位轉換原理

- 根據對訊號的取樣率和取樣的解析度，會影響我們得到的數位訊號。

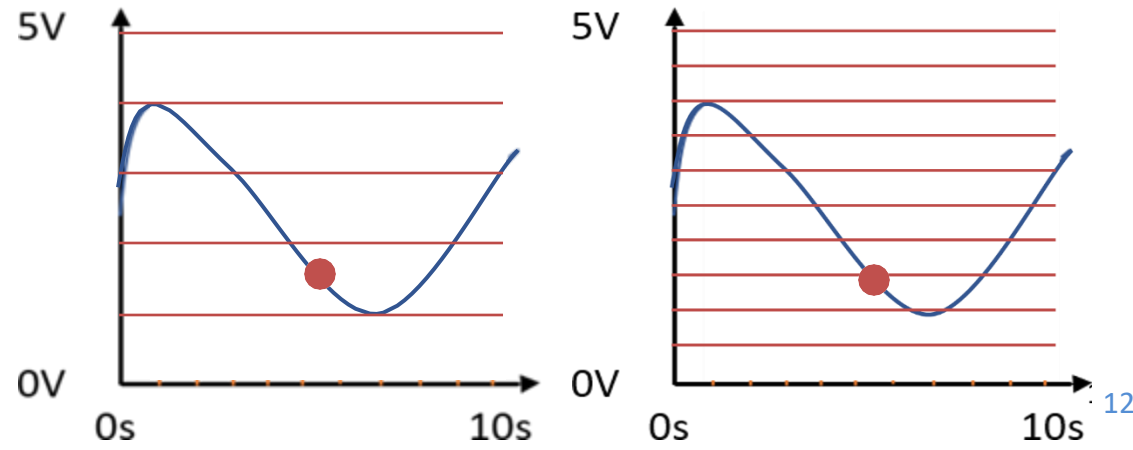
➤ 取樣率 (Sampling rate) :

經過多久的時間對輸入的類比訊號做一次轉換。轉換的頻率越高，所得到的數位訊號資料越連續，但相對的也需要較快的處理速度以及更多的存放空間。



➤ 解析度 (Resolution) :

類比轉換數位訊號的值有多精確，影響到每次轉換後的資料準確度。解析度越高，所得到的數位訊號資料越精確，量化誤差越小，但相對的每筆資料儲存所需要的位元數將更多。



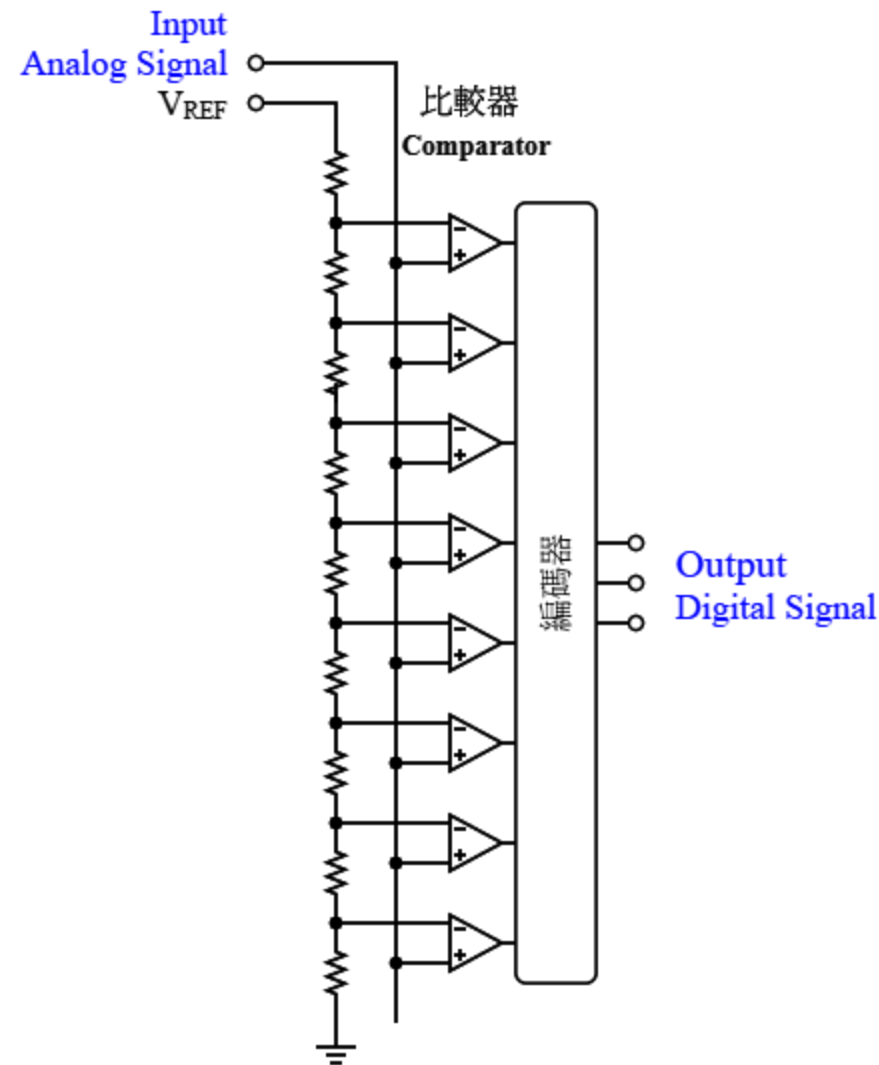
元件原理 — 類比數位轉換原理

- 一般而言8位元的解析度，我們可將輸入的類比訊號解碼成 $2^8=256$ 階Step size(階層電壓值)，256個不同的類比訊號解碼，每一階的二進位碼都不同。
- 如果使用8位元解析度，且電壓上限為0V到5V，則數位訊號上升一個階層需要的電壓為 $(5V - 0V)/(2^8) = 0.0195V$ 。
- 若收到的訊號為 $200_{(10)}$ ，則轉換前的類比電壓為 $200 * 0.0195V = 3.9V$ 。

ADC 的轉換方式

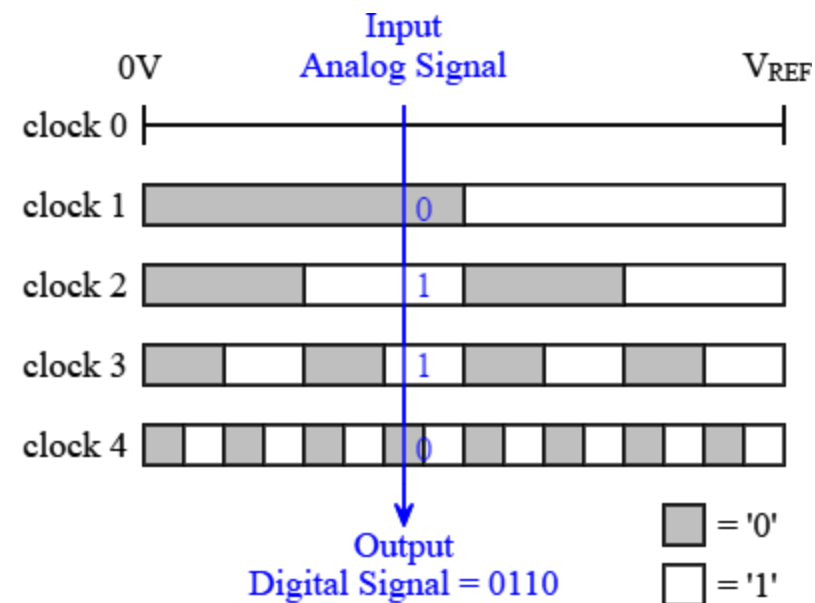
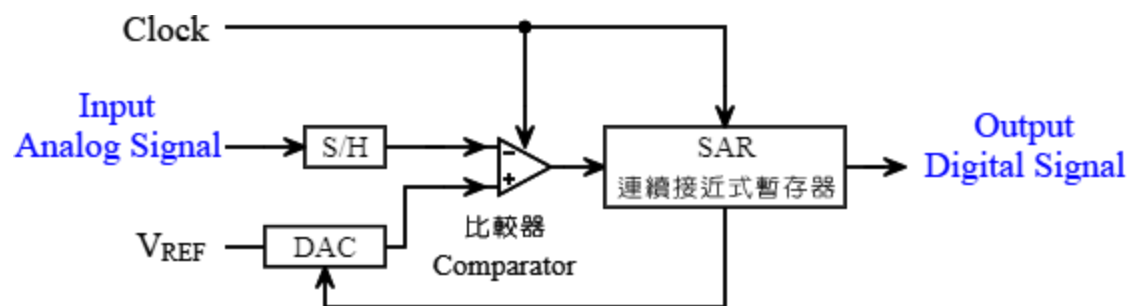
元件原理—並列式類比-數位轉換

- 並列式類比-數位轉換器以多個比較器並列偵測輸入的類比訊號。
- 根據偵測輸入結果編碼，即可輸出數位訊號。
- 此方法轉換速度快，但所需要的電路較為複雜。



元件原理—連續接近式類比-數位轉換

- 此種轉換方式優點為IC設計簡單，缺點為轉換的時間較長， n 位元的轉換器轉換一次的花費時間為 n 個時鐘脈波。以下我們要介紹的ADC即為連續接近式類比數位轉換器。
- 連續接近式類比-數位轉換器，用快速比對接近的方式，將類比訊號轉為數位訊號。



元件原理——連續接近式類比-數位轉換

- 若以8位元的轉換器為例，一開始計數器為1，
初始的轉換結果暫存器 SAR 輸出為 轉換解析度的一半 $10000000_{(2)}$
初始的 V_s 為轉換後的解析度的一半 128 (256/2)。

若輸入的類比電壓大於 V_s

則將轉換結果暫存器的值加上
 $2^{(\text{解析度位元數}-\text{計數器}-1)}$

若輸入的類比電壓小於 V_s

則將轉換結果暫存器的值減去
 $2^{(\text{解析度位元數}-\text{計數器}-1)}$

- 判斷完後，計數器加1，並且繼續將轉換結果暫存器的內容送給數位類比轉換器
- 直到計數器達到轉換位數(這裡為8)時，再比較一次

若輸入的類比電壓大於 V_s

則不動，轉換結果暫存器即存儲了結果

若輸入的類比電壓小於 V_s

則將則將最小位元清零

元件原理—連續接近式轉換方式

- 計數器為1，結果暫存器為 $1000_{(2)}$ ，而 V_s 為 8。
- 比較 $1000_{(2)}$ 轉成的階層電壓值 8V 和輸入電壓，輸入電壓較低，故將

SAR的值 $1000_{(2)}$ 減去 $[2^{(4-1-1)} = 0100_{(2)}]$ 得到 $0100_{(2)}$ ，並將計數器加1。

- 再比較 $0100_{(2)}$ 轉成的階層電壓值 4V 和輸入電壓輸入電壓較高，故將

SAR的值 $0100_{(2)}$ 加 $[2^{(4-2-1)} = 0010_{(2)}]$ 得到 $0110_{(2)}$ ，並將計數器加1。

以4位元的轉換器為例

計數器	結果暫存器	V_s	輸入的電壓
1	$(2^4)/2=8 \rightarrow 1000$	8V	6.5V
2	$2^{(4-1-1)}=4 \rightarrow 0100$ $1000-0100=0100$	4V	6.5V
3	$2^{(4-2-1)}=2 \rightarrow 0010$ $0100+0010=0110$	6V	6.5V
4	$2^{(4-3-1)}=1 \rightarrow 0001$ $0110+0001=0111$ $\Rightarrow 0110$	7V	6.5V

元件原理—連續接近式轉換方式

- 再比較 $0110_{(2)}$ 轉成的階層電壓值 6V 和輸入電壓

輸入電壓較高，故將

SAR的值 $0110_{(2)}$ 加 $[2^{(4-2-1)} = 0001_{(2)}]$ 得到 $0111_{(2)}$ ，並將計數器加1。

- 當計數器等於4時，最後比較 $0111_{(2)}$ 轉成的類比電壓

輸入電壓較低，故將

最小位數清0，獲得結果 $0110_{(2)}$ 。

以4位元的轉換器為例

計數器	結果暫存器	V_s	輸入的電壓
1	$(2^4)/2=8 \rightarrow 1000$	8V	6.5V
2	$2^{(4-1-1)}=4 \rightarrow 0100$ $1000-0100=0100$	4V	6.5V
3	$2^{(4-2-1)}=2 \rightarrow 0010$ $0100+0010=0110$	6V	6.5V
4	$2^{(4-3-1)}=1 \rightarrow 0001$ $0110+0001=0111$ $\Rightarrow 0110$	7V	6.5V

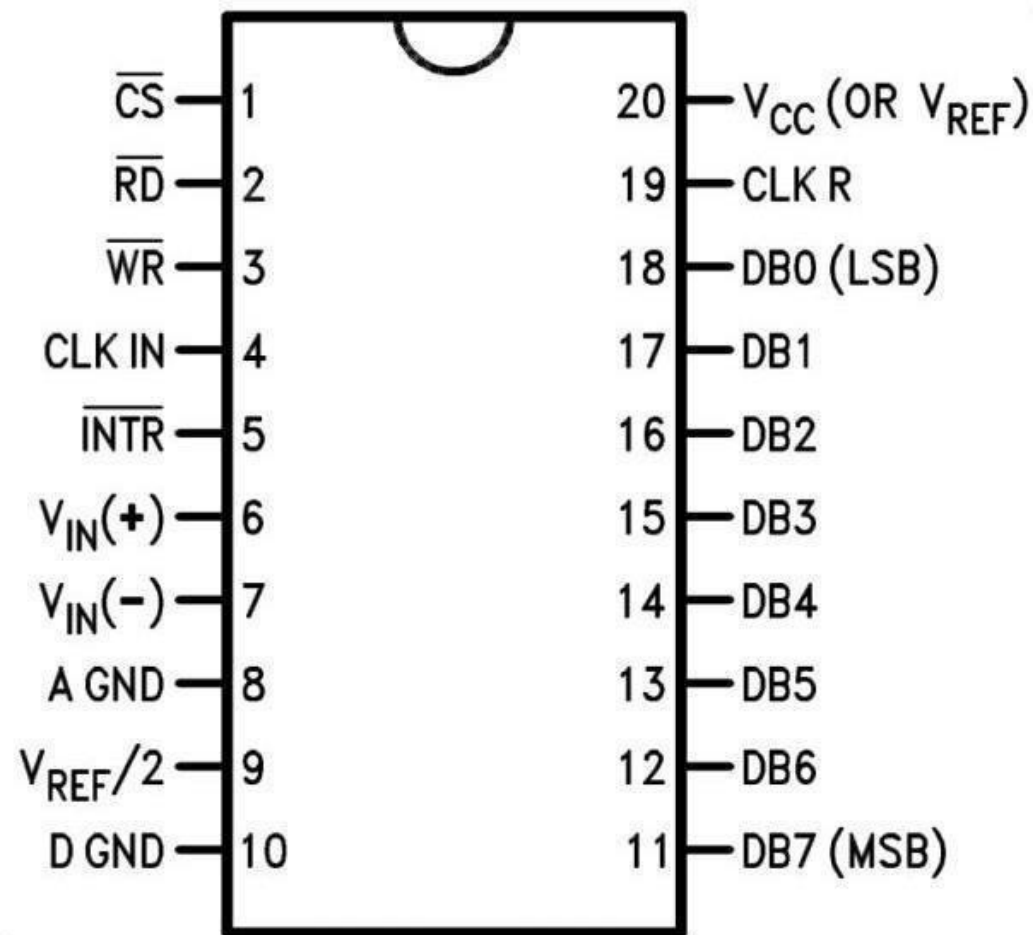
ADC0804 的操作方式

元件原理—ADC0804

➤CS：晶片選擇接腳，此接腳須為0時，ADC0804才會動作，若此接腳為1，輸出接腳(DB0~DB7)將會呈現高抗阻狀態。

➤RD：資料讀取接腳，若此接腳為0且CS接腳也為0，則可由輸出接腳讀取轉換過後的資料，若此接腳為1，輸出接腳(DB0~DB7)將會呈現高抗阻狀態。

➤WR：此接腳控制ADC0804是否開始轉換，若WR為0則ADC做清除的動作，系統重置，若WR由0變1時，開始執行類比轉數位的轉換，此時INTR變為1。



元件原理—ADC0804

➤CS：晶片選擇接腳，此接腳須為0時，ADC0804才會動作，若此接腳為1，輸出接腳(DB0~DB7)將會呈現高抗阻狀態。

➤RD：資料讀取接腳，若此接腳為0且CS接腳也為0，則可由輸出接腳讀取轉換過的資料，若此接腳為1，輸出接腳(DB0~DB7)將會呈現高抗阻狀態。

➤WR：此接腳控制ADC0804是否開始轉換，若WR為0則ADC做清除的動作，系統重置，若WR由0變1時，開始執行類比轉數位的轉換，此時INTR變為1。

CS	RD	WR	作用
0			ADC有動作
0	0		由輸出接腳讀取轉換後的資料
0	1		X
0		0/1	控制ADC是否開始轉換
		0	ADC 做清除
		1	由 0→1時做轉換，INTR 變1(正在轉換)
			INTR 變0(完成轉換)

元件原理—ADC0804

- INTR：若INTR接腳等於0，代表ADC0804已完成轉換，當資料被讀取後恢復為1。這個接腳通常也和外部中斷搭配使用。
- CLK IN /CLK R：CLK R 可以搭配電阻器產生時鐘脈波。CLK IN則是輸入時鐘脈波的接腳，故我們可以使用CLK R輸出的時鐘脈波輸入給CLK IN使用。
- V_{REF} ：提供ADC0804轉換的參考電壓，電壓準位為最大輸入類比電壓的一半。若空接則為 $VCC/2$ 。
- $V_{IN}(+)/V_{IN}(-)$ ：類比電壓輸入接腳，正負電壓分別接至 $V_{IN}(+)$ 和 $V_{IN}(-)$ 端，若為單一正電壓，則接到 $V_{IN}(+)/$ ，再將 $V_{IN}(-)$ 接地。
- D GND/A GND：數位/類比訊號接地接腳，通常將A GND和D GND接腳相接後再接地。

元件原理—ADC0804操作方式

- **連續轉換：**

ADC0804不停的進行轉換類比資料，要注意轉換頻率也是有極限的，若ADC時脈為30MHz 而解析度為8bits時， $30\text{MHz}/8$ 大概為轉換頻率極限。

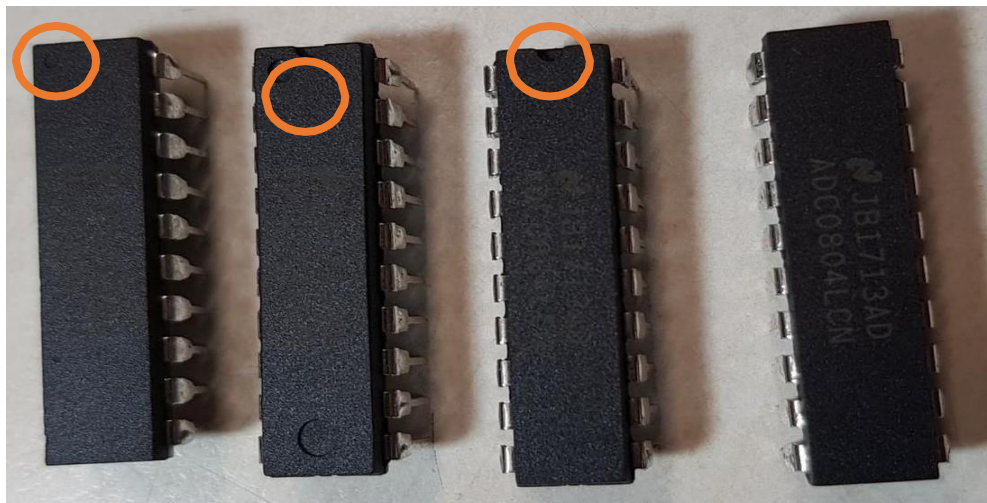
將CS與RD接腳直接接地，並把INTR和WR接腳對接，使輸出完成後可以直接執行下一次轉換。

- **交握式控制：**

將RD與WR接腳接到 8051 的輸出埠，並把 INTR 訊號連接到 8051 外部中斷接腳，當 INTR 輸出完成訊號 (0) 時，8051經由讀取接腳得知，觸發中斷，使用中斷方式讀取轉換資料。

元件原理—ADC0804注意事項

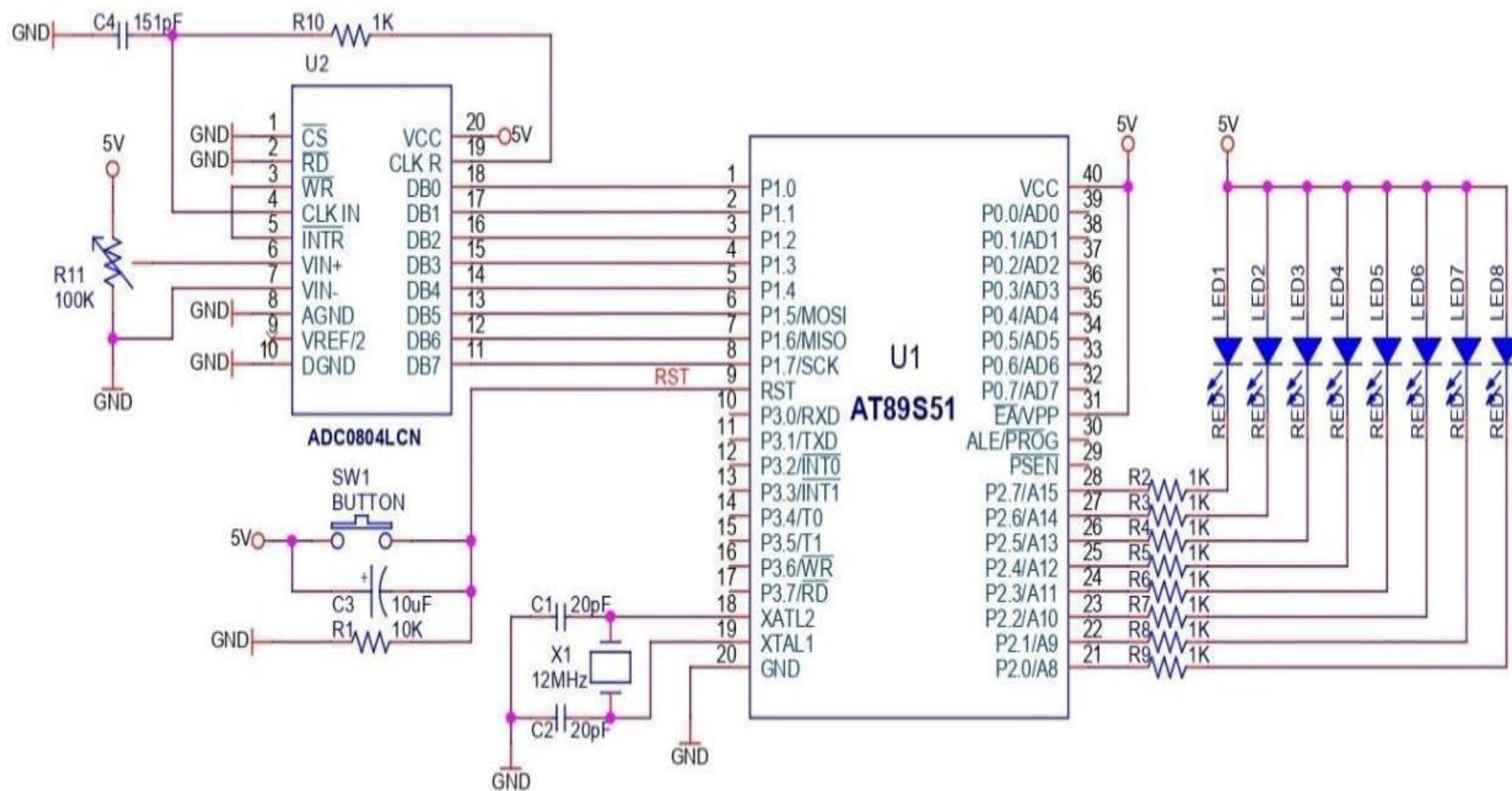
- ADC0804的價格比較昂貴，請小心使用。
- 在接電路時，要注意第一個腳位的位置，不要裝反。
- 如果正面看不出腳位標記，翻到IC的背面通常會有標記。



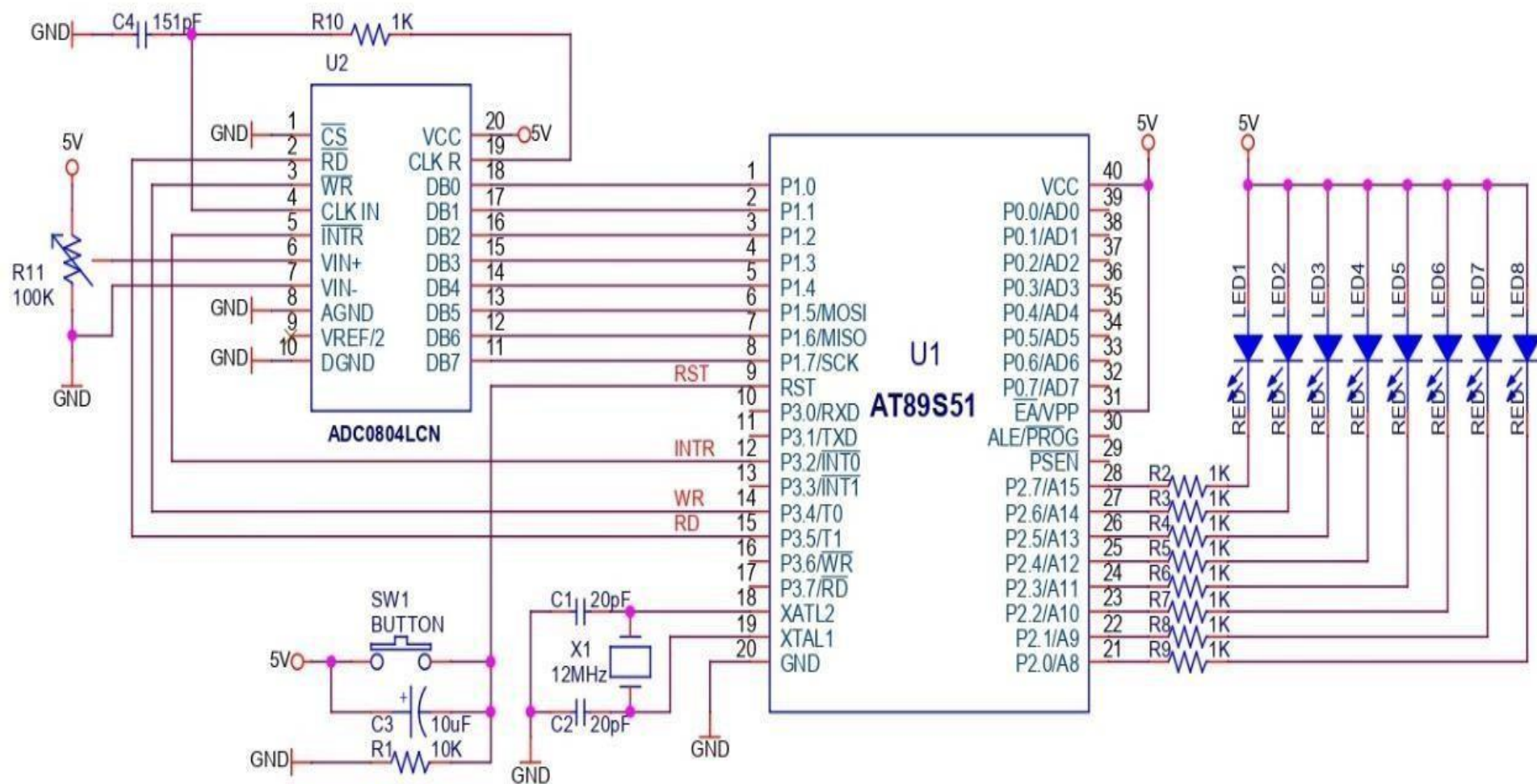
Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

實驗電路圖—連續轉換



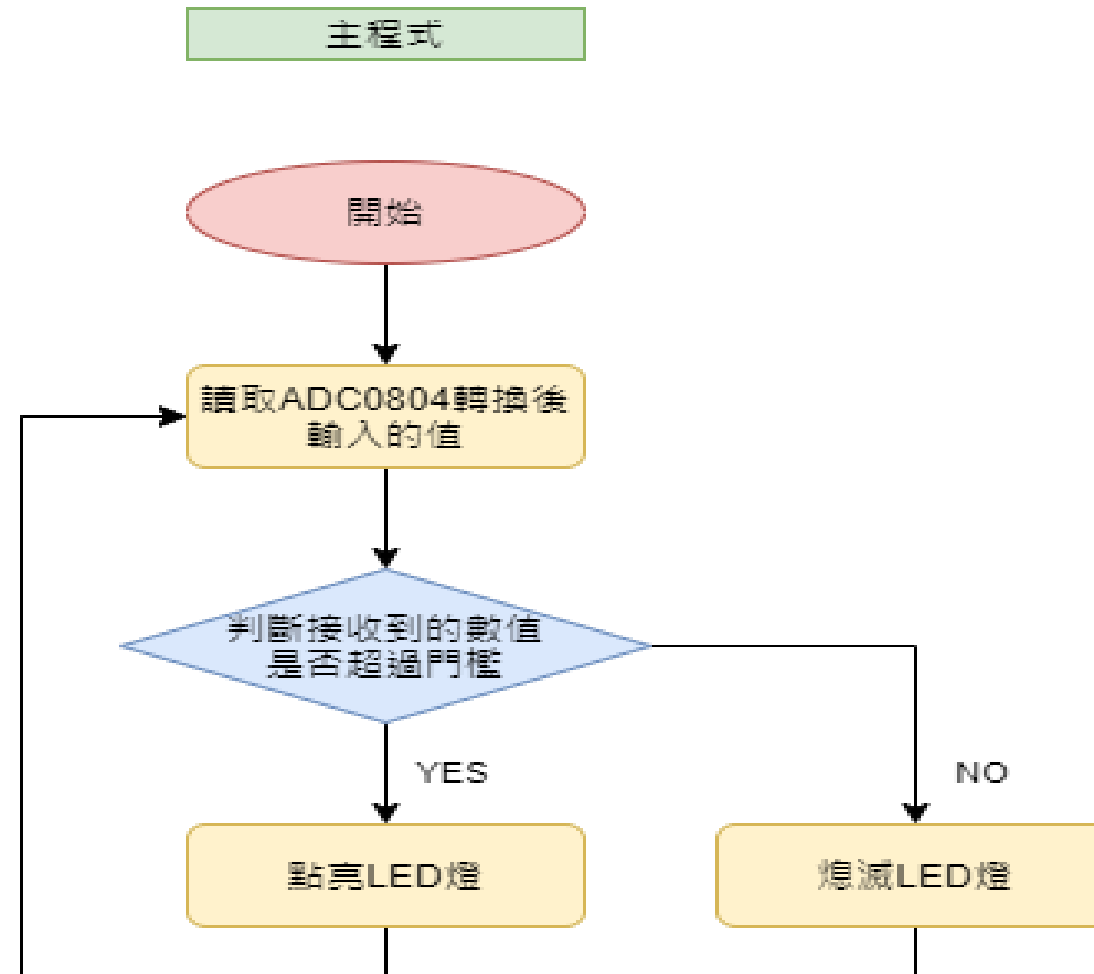
實驗電路圖—交握式控制



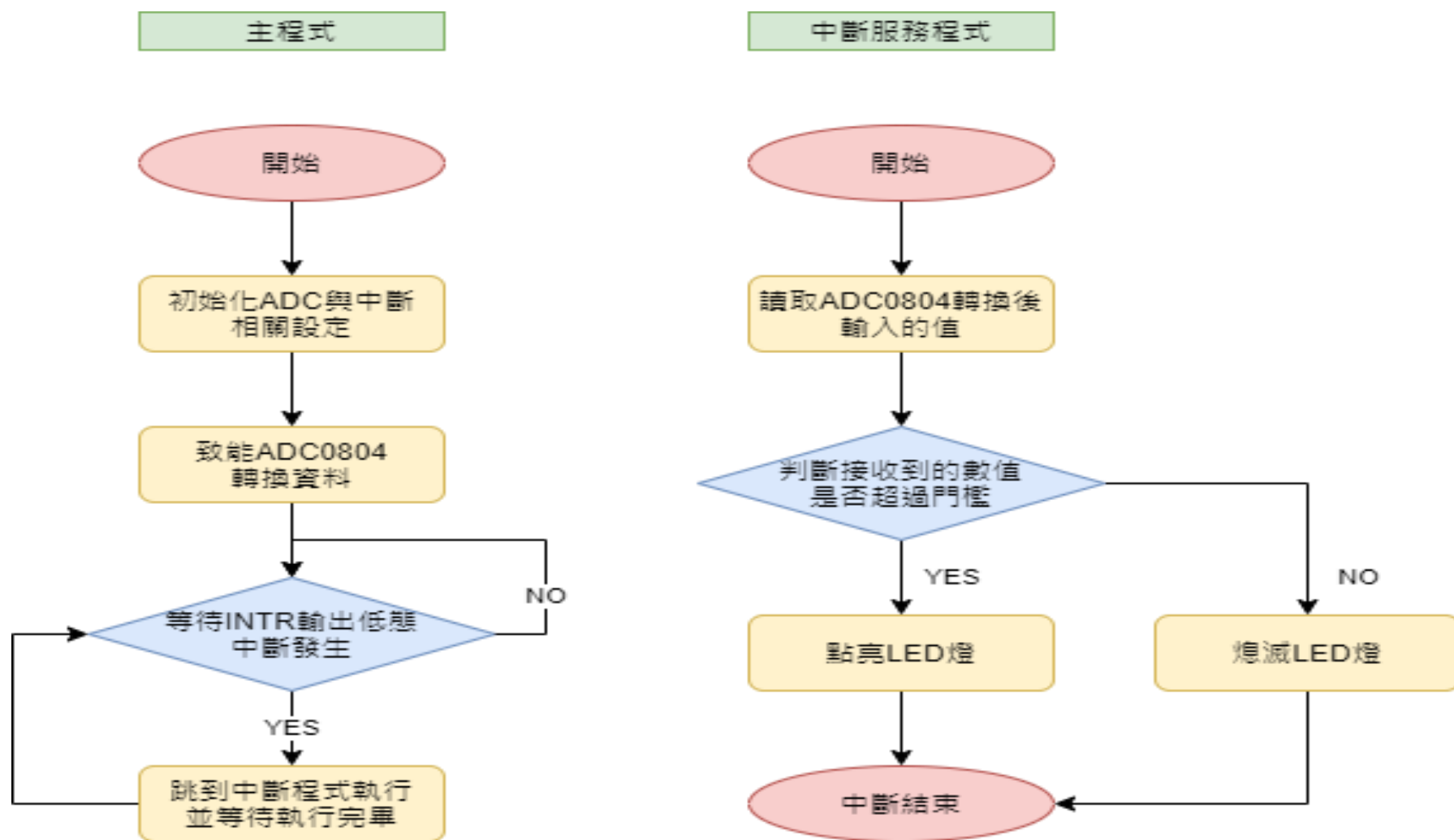
Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

軟體流程圖——輪詢法



軟體流程圖——中斷



Outline

- 學習重點
- 實驗內容
- 材料清單
- 元件原理
- 實驗電路圖
- 軟體流程圖
- 範例程式碼

範例程式碼—— 輪詢

```
1.  #include <regx51.h>
2.
3.  void main()
4.  {
5.      while(1)
6.      {
7.          if(P1 > 256 / 2)           //if P1 greater than 256/2
8.              P2 = 0;                //LED on
9.          else                       //LED off
10.             P2 = 255;
11.     }
12. }
```

範例程式碼—— 中斷

```
1. #include <regx51.h>
2.
3. void main()
4. {
5.     IT0 = 0;           //enable a low-level signal on external interrupt
6.     EX0 = 1;           //enable INT0
7.     EA = 1;            //enable interrupt
8.     P3_4 = 0;          //WR=0,clean the data
9.     P3_4 = 1;          //WR=1,analog convert to digital
10.    while(1);          //infinite loop
11.}
12.
```

範例程式碼—— 中斷

```
13. void int_0(void) interrupt 0 //INT0 interrupt function
14. {
15.     P3_5 = 0; //RD=0, read the digital data
16.     if (P1 > 256 / 2)
17.     {
18.         P2 = 0; //LED on
19.     }
20.     else
21.     {
22.         P2 = 255; //LED off
23.     }
24.     P3_5 = 1; //RD=1
25.     P3_4 = 0; //WR=0, clean the data
26.     P3_4 = 1; //WR=1, analog convert to digital
27. }
```

Q & A