

## 基本資料

---

### Question 1

AT8P5X I/O Type MCU 內容如何？

#### Answer

AT8P5X I/O Type MCU 提供 1K、2K、4K 的 OTP Type 及 Mask Type 的 MCU，編號為 AT8P53E,AT8P56E/AT8P56,AT8P57E,AT8P59E,AT8P51E/AT8P51 總共 7 顆 IC，規格功能請參考 ATW Data Sheet。

---

### Question 2

AT8P5X I/O Type MCU 提供何種類的封裝型式？

#### Answer

AT8P53E: AT8P53EP 14 SKDIP(300mil), AT8P53ED 14 SOP(150mil)  
AT8P56E: AT8P56EP 18 SKDIP(300mil), AT8P56ED 18 SOP(300mil),  
AT8P56ER 20 SSOP(209mil)  
AT8P56 : AT8P56P 18 SKDIP(300mil), AT8P56D 18 SOP(300mil),  
AT8P56R 20 SSOP(209mil)  
AT8P57E: AT8P57EP 28 SKDIP(600mil), AT8P57EM 28 DIP(300mil),  
AT8P57ED 28 SOP(300mil), AT8P57ER 28 SSOP(209mil)  
AT8P59E: AT8P59AEP 28 SKDIP(600mil), AT8P59AEM 28 DIP(300mil),  
AT8P59AED 28 SOP(300mil), AT8P59AER 28 SSOP(209mil),  
AT8P59BED 32 DIP(600mil), AT8P59BER 32 SOP(450mil)  
AT8P51E: AT8P51EP 40 DIP(600mil), AT8P51EF 44 QFP(10x10)  
AT8P51 : AT8P51P 40 DIP(600mil), AT8P51F 44 QFP(10x10)

---

### Question 3

AT8P5X I/O Type MCU 有何重要功能及特性？

#### Answer

操作電壓範圍 2.0V ~ 5.5V  
溫度規格 0°C ~ +70°C  
Low Voltage Reset 功能  
OTP / Mask Type MCU 相容  
instruction cycle OSC/2 or OSC/4  
其它請參考 ATW Data Sheet

---

### Question 4

AT8P5X Type MCU 是否提供 Dice form？

### Answer

ATW 同時提供 OTP Type 及 Mask Type MCU 的 Dice form 的量產服務，但使用 OTP MCU Dice 量產時，客戶要特別注意打線及燒錄的操作問題。

### Question 5

AT8P5X I/O Type MCU 的工作頻率為何？

### Answer

在 5V、3.3V 及 3 個電池的應用，工作頻率達 Max. 20MHz (3.3V)。

在 3V 及 2 個電池的應用，工作頻率達 Max. 4MHz (2.0V)。

### Question 6

AT8P5X I/O Type MCU 的系統頻率有那些？

### Answer

AT8P5X I/O Type MCU 的系統頻率來源有三種，如下：

1. 外部 RC
2. 外部 Crystal
3. 內部 RC，如果選擇內部 RC 只有 455kHz、1MHz、4MHz 和 8MHz 四種選擇

## 功能說明

### Question 1

AT8P5X I/O Type MCU 功能有何不同？

### Answer

Part NO	ROM	RAM	I/O	Ext. Interrupt	Int. Interrupt	Wake-up	Stack	Timer	Other function	package
AT8P53E	1K	49	12	2	1	Yes	5	8-bit*1	IRC	
AT8P56E	1K	49	12	2	1	Yes	5	8-bit*1	No	
AT8P56	1K	49	16	2	1	Yes	5	8-bit*1	IRC	
AT8P57E	2K	96	20	1	1	Yes	5	8-bit*1	No	
AT8P59E	4K	144	24	2	1	Yes	5	8-bit*1	No	
AT8P51E	4K	140	38	1	7	Yes	5	8-bit*4 or 8-bit*2, 16-bit*1	IRC,RFC. PWM,SPI	
AT8P51	4K	140	38	1	7	Yes	5	8-bit*4 or 8-bit*2, 16-bit*1	IRC,RFC. PWM,SPI	

註：細節規格請參考 ATW Data Sheet。

---

### Question 2

間接定址如何使用？

#### Answer

首先將要讀寫之 Data Memory 的位址先存入位址指標暫存器 (FSR R04H)，然後對(INDF R00H)執行讀寫 (例如: MOVAR INDF),即可讀到 FSR 所指到的 Data (FSR 的內容當位址)。間接定址功能特別方便於數列 (array) 的處理，一般先將數列位址放入 FSR 做運算後再讀取所需的數列 Data。

範例

假設 INDF=03DH,FSR=00H,R20H=02FH

```
MOVIA    020H    ;ACC <=020H INDF=03DH
          ;FSR=00H    , R20H=02FH

MOVAR    FSR     ;FSR <= ACC INDF=02FH
          ;FSR=020H   R20H=02FH

MOVIA    045H    ;ACC <= 045H INDF=02FH
          ;FSR=020H   R20H=02FH

MOVAR    INDF    ;INDF <= ACC INDF=0045H
          ;FSR=020H   R20H=045H
```

---

### Question 3

已經進入 SLEEP Mode 為何還有耗電？

#### Answer

MCU 進入 SLEEP 會將系統 OSC 關閉,但 WDT OSC (如果 WDT Enable)仍在工作,而且全部 I/O 會保持 SLEEP 前狀態,除了 WDT OSC 會耗一些電外,最要注意的是 Input Port 是否浮接或者 Output Port 是否仍有負載,這些 I/O 處理要非常小心,否則會耗大電流,另外如果有打開低電壓檢測功能(LVDT)也會耗一些電流 (0.5uA~3uA)。

---

### Question 4

PORT B 為何沒有喚醒 MCU？

#### Answer

1. PORT B 是否設為 INPUT MODE?
  2. Pin Change Wake Up 的中斷旗標是否致能?
  3. WUCON(R09H) 是否有設為 Wake up enable?
  4. 輸入腳是否平常是浮接.
  5. 進入 SLEEP 以前是否有讀 PORT B
- 

### Question 5

8-Bit TIMER 值如何設定？

**Answer**

8-Bit Timer/Counter 為一向上數之計數器 (Counter)，每當 Counter 數到 255 (0FFH) 溢位 0H 時產生中斷 (Interrupt)，所以若要計數為 N ( $N < 256$ )；則 TMR 值要設定成  $256-N$ ，設定好後啟動 TMR；TMR 即從設定值往上數到 255 溢出到 0 (即 256)，故所得值即  $256-(256-N)=N$ 。

**Question 6**

Timer0 之 Timer Mode 及 Event Counter Mode 如何設定？

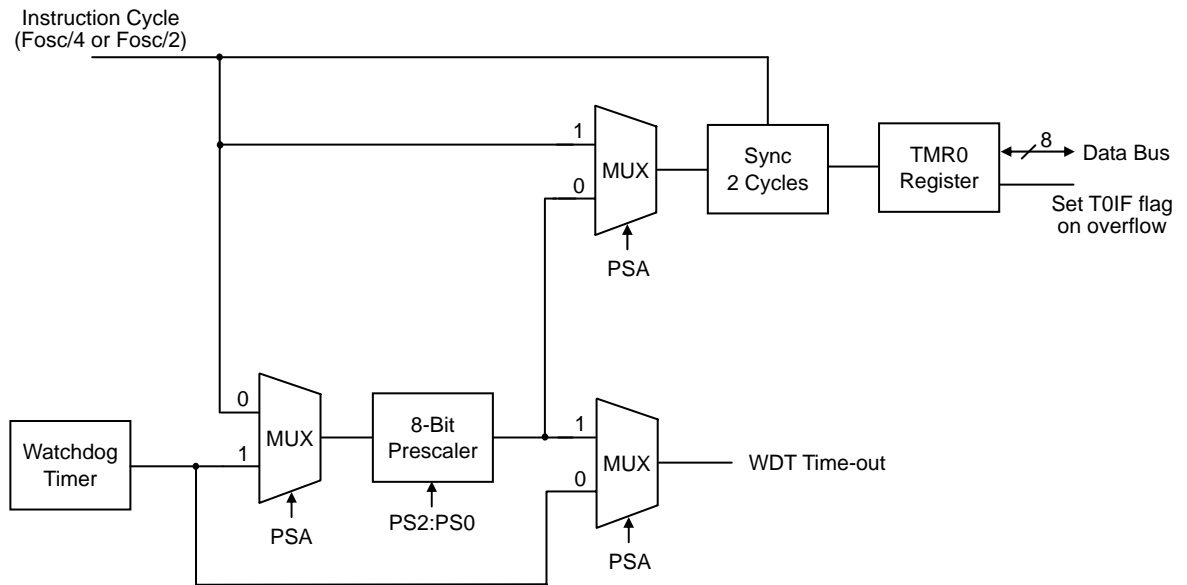
**Answer**

Timer0 的時脈來源可能獲得自指令週期時脈或一個外在時脈來源(T0CKI 腳位) 由 T0CS 位元(OPTION<5>) 定義。如果 T0CKI 腳位被選擇, T0CKI 信號的上升緣或下降緣會遞增 TIMER0 的值 (由 T0SE 位元 (OPTION<4>) 選擇)。

清除 PSA 位元(OPTION<3>)預除器 被分配到 Timer0。在這種情況下, 當 TMR0 暫存器被寫入值,預除器 將被清除。

OPTION 指令被執行, ACC 暫存器的內容將轉移到 OPTION 暫存器。OPTION 暫存器是 7 位元寬, 包含各種的控制位元配置 Timer0/WDT 預除器、Timer0, 和外 INTR 腳位中斷。

OPTION 暫存器只可以寫入,除了 INTEDG 位元。



AT8P53/F6/57 TIMER0 相關占存器

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
N/A (w)	OPTION	-	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
01h (r/w)	TMR0	TMR07	TMR06	TMR05	TMR04	TMR03	TMR02	TMR01	TMR00
0Eh (r/w)	INTEN	GIE	-	-	-	-	INTIE	PBIE	TOIE
0Fh (r/w)	INTFLAG	-	-	-	-	-	INTIF	PBIF	TOIF

AT8P51 TIMERO 相關占存器

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
Unbanked IOST									
N/A (r/w)	OPTION	/PHBCE	GIE			PSA	PS2	PS1	PS0
0Fh (r/w)	INTEN	SPITXIE	RFCIE	T3IE	T2IE	T1IE	SPIRCIE	INTIE	TOIE
3Fh (r/w)	INTFLAG	SPITXIF	RFCIF	T3IF	T2IF	T1IF	SPIRCIF	INTIF	TOIF
UNBANKED									
01h (r/w)	TMR0	TMR07	TMR06	TMR05	TMR04	TMR03	TMR02	TMR01	TMR00

PS2:PS0: 預除器 rate select bits.

PS2:PS0	Timer0 Rate	WDT Rate
0 0 0	1:2	1:1
0 0 1	1:4	1:2
0 1 0	1:8	1:4
0 1 1	1:16	1:8
1 0 0	1:32	1:16
1 0 1	1:64	1:32
1 1 0	1:128	1:64
1 1 1	1:256	1:128

PSA: 預除器 分配位元

= 1, WDT (看門狗計時器)。

= 0, TMR0 (Timer0)。

TOSE: TMR0 來源邊緣選擇位元

= 1, 下降邊緣在 TOCKI 腳位

= 0, 上升邊緣在 TOCKI 腳位。

TOCS: TMR0 時脈來源選擇位元

= 1, 外在 TOCKI 腳位

= 0, 內部指令時脈週期。

INTEDG: 中斷邊緣選擇位元

= 1, 在 INT 腳位上升邊緣中斷

= 0, 在 INT 腳位下降邊緣中斷。

Bit7: 沒使用。

要執行 Timer Mode 要執行以下動作設定：

1. 設定 Timer/Counter 為內部指令時脈週期(設定 TOCS =0)
2. 設定預除器 clock 來源,為 WDT 或 TMR。
3. 依據 Timer 的長短選定 Prescaler 值 (PS2~PS0)
4. 設定計數初始值 (TMR)。

5. 設定 TIMER0 中斷旗標
6. 啓動通用中斷允許

#### 範例

假設 OSC=4MHz instruction CYCLE 爲 OSC/2 我們要設定一個 10ms 的中斷

因爲 instruction CYCLE 爲 OSC/2 所以 TIMER0 CLOCK input frequency = 2Mhz = 0.5us

1ms/0.5us=2000 clock

2000/16=125

N=256-125=131=83H

MOVIA 03H ;設定 Timer/Counter 爲內部指令時脈週期 T0CS=0

; 設定預除器 clock 來源 TMR PSA=0

;依據 Timer 的長短選定 Prescaler 值 (PS2~PS0)=03H

OPTION

MOVIA 083H ; N=256-125=131=83H

MOVAR TMR0;設定計數初始值 (TMR)

BSR INTEN,T0IE ; 設定 TIMER0 中斷旗標

BSR INTEN,GIE ; 啓動通用中斷允許;在 AT8P51 可使用 ENI 指令

要執行 Event Counter Mode 要執行以下動作設定：

1. 設定 Timer/Counter 爲外部 T0CKI 時脈週期(設定 T0CS =1)
2. 設定預除器 clock 來源,爲 WDT 或 TMR0。
3. 依據 Timer 的長短選定 Prescaler 值 (PS2~PS0) 爲 0
4. 設定外部信號源爲上升源或下降源(T0SE)
5. 請務必加上兩個 NOP 指令 OPTION 指令才會正常動作
6. 設定計數初始值 (TMR) 爲 0。

#### 範例

假設我們要設定一個 10ms 的中斷

因爲 instruction CYCLE 爲 OSC/2 所以 TIMER0 CLOCK input frequency = 2Mhz = 0.5us

1ms/0.5us=2000 clock

2000/16=125

N=256-125=131=83H

MOVIA 23H ;設定 Timer/Counter 爲內部指令時脈週期 T0CS=0

; 設定預除器 clock 來源 TMR PSA=0

;依據 Timer 的長短選定 Prescaler 值 (PS2~PS0)=03H

OPTION

NOP

NOP

MOVIA 083H ; N=256-125=131=83H

MOVAR TMR0 ;設定計數初始值 (TMR)  
 BSR INTEN,TOIE ; 設定 TIMER0 中斷旗標  
 BSR INTEN,GIE ; 啓動通用中斷允許

此時如果外部有時輸入

### Question 7

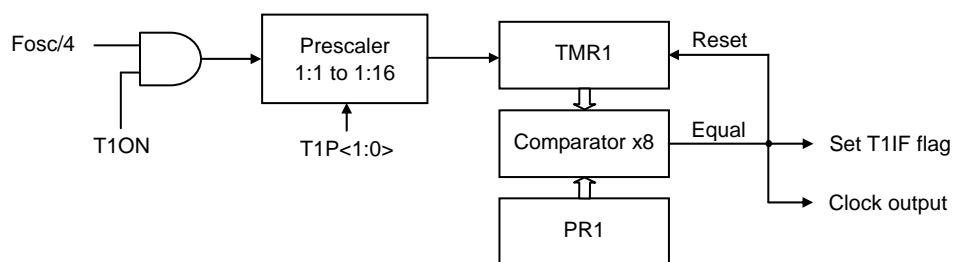
TIMER1 8-Bit Timer 如何設定 ? (以 AT8P51 為例)

### Answer

Timer1 的時脈來源得自指令週期,T1CON 暫存器的 T1ON 控制時脈輸入,T1P<1:0>控制時脈輸入預除器。當 TMR1 及 PR1 相等時而且 TIMER1 中斷及啓動通用中斷允許就會產生中斷並且清除 TMR1。

AT8P51 TIMER1 相關占存器

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
IOST									
0Ch (r/w)	T1CON						T1ON	T1P1	T1P0
0Fh (r/w)	INTEN	SPITXIE	RFCIE	T3IE	T2IE	T1IE	SPIRCIE	INTIE	TOIE
BANK0									
0Eh (r/w)	TMR1	TMR17	TMR16	TMR15	TMR14	TMR13	TMR12	TMR11	TMR10
0Fh (r/w)	PR1	PR17	PR16	PR15	PR14	PR13	PR12	PR11	PR10
3Fh (r/w)	INTFLAG	SPITXIF	RFCIF	T3IF	T2IF	T1IF	SPIRCIF	INTIF	TOIF



T1P1 : T1P0	Prescaler Rate
0, 0	1 : 1
0, 1	1 : 4
1, 0	1 : 8
1, 1	1 : 16

要執行 TIMER1 中斷要執行以下動作設定：

1. 寫入或清除 TMR1

2. 寫入 PR1
3. 設定 TIMER1 中斷致能旗標
4. 清除 TIMER1 中斷發生旗標
5. 啓動通用中斷允許
6. 設定預除器比率吃 T1P<1:0> 及啓動頻率輸入 T1ON
7. 依據 Timer 的長短選定 Prescaler 值 (PS2~PS0) 爲 0

範例

假設 OSC=4MHz instruction CYCLE 爲 OSC/4 我們要設定一個 10ms 的中斷

因爲 instruction CYCLE 爲 OSC/4 所以 TIMER0 CLOCK input frequency = 1Mhz = 1.0us

1ms/1.0us=1000 clock

1000/8=125=7DH

CLR TMR1; 清除 TMR1

MOVIA 07DH

MOVAR PR1 ;寫入 PR1

MOVIA 04H

IOST INTEN ;設定 TIMER0 中斷旗標

BCR INTFLAG, T1IF

ENI ;啓動通用中斷允許

MOVIA 06H

IOST T1CON ; 設定預除器比率吃 T1P<1:0> 1:8 及啓動頻率輸入 T1ON

Question 8

TIMER2,TIMER3 8-Bit Timer 如何設定 ? (以 AT8P51 爲例)

Answer

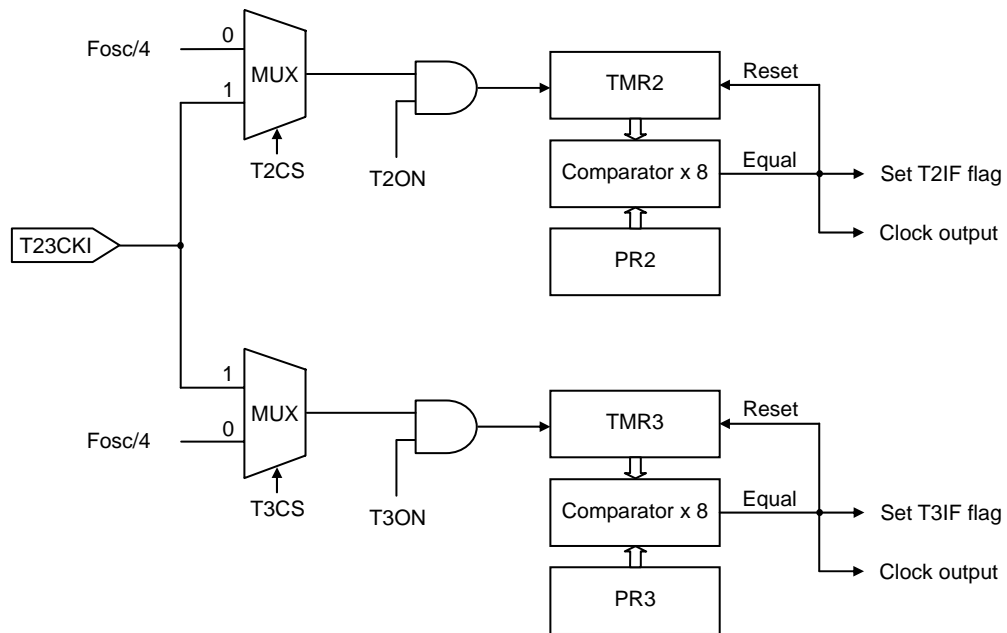
Timer2,Timer3 的時脈來源來自指令週期,或 T23CKI(PB0).T23CON 暫存器的 T2CS' T3CS 控制時脈輸入來自指令週期,或 T23CKI(PB0).T2ON' T3ON>控制時脈輸入。當 TMR2 及 PR2 相等或 TMR2 及 PR2 時而且 TIMER2' TIMER3 中斷及啓動通用中斷允許就會產生中斷並且清除 TMR2' TMR3。

AT8P51 TIMER2,TIMER3 相關占存器

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
IOST									
0Fh (r/w)	INTEN	SPITXIE	RFCIE	T3IE	T2IE	T1IE	SPIRCIE	INTIE	TOIE
BANK1									
0Ah (r/w)	T23CON	-	-	-	T16	T3ON	T3CS	T2ON	T2CS
0Bh (r/w)	TMR2	TMR27	TMR26	TMR25	TMR24	TMR23	TMR22	TMR21	TMR20
0Ch (r/w)	PR2	PR27	PR26	PR25	PR24	PR23	PR22	PR21	PR20
0Dh (r/w)	TMR3	TMR37	TMR36	TMR35	TMR34	TMR33	TMR32	TMR31	TMR30
0Eh (r/w)	PR3	PR37	PR36	PR35	PR34	PR33	PR32	PR31	PR30



FIGURE 2.6: Timer2 and Timer3 in Two 8-bit Timer/Counter Mode



要執行 TIMER2 TIMER MODE 中斷要執行以下動作設定：

1. 設定 BANK1
2. 寫入或清除 TMR2 或 TMR3
3. 寫入 PR2 或 PR3
4. 設定 TIMER2 或 TIMER3 中斷旗標
5. 啓動通用中斷允許
6. 設訊號源及啓動頻率輸入
- 7.

範例

假設 OSC=4MHz instruction CYCLE 爲 OSC/4 我們要設定一個 250us 的中斷

因爲 instruction CYCLE 爲 OSC/4 所以 TIMER0 CLOCK input frequency = 1Mhz = 1.0us

$250\mu s / 1.0\mu s = 250$  clock

BSR FSR,BIT6

BCR FSR,BIT7 ;切換 BANK1

CLR TMR2 ;清除 TMR1

MOVA 0FAH

MOVAR PR2 ;寫入 PR1

MOVA 08H

IOST INTEN ;設定 TIMER2 中斷旗標

ENI ;啓動通用中斷允許

MOVIA 06H

IOST T23CON ; 設定 T2CS 選擇 OSC/4 及啓動頻率輸入 T2ON

### Question 9

Timer2/Timer3 16-Bit Timer/Counter 之 Timer Mode 如何設定？(以 AT8P51 為例)

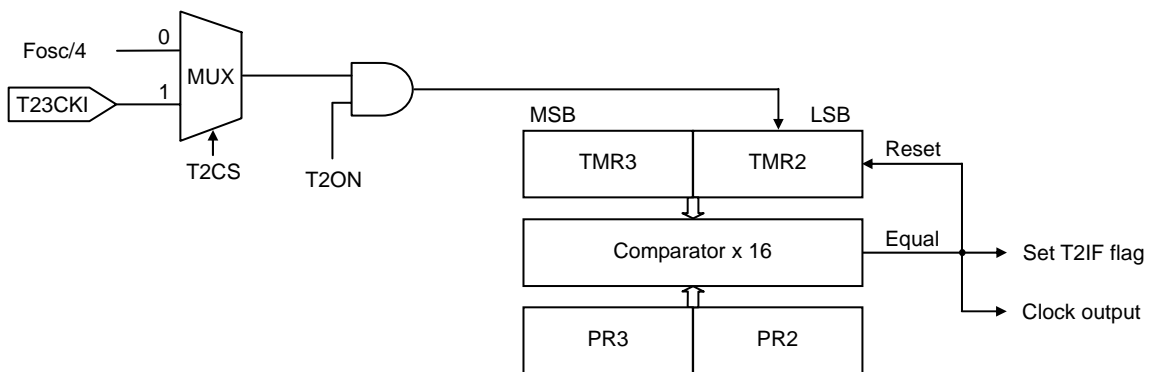
### Answer

Timer2,Timer3 除了可以是兩個獨立的 8-BIT timer/counter 亦可以連接起來成爲一個 16 bit 的 timer/counter . timer2 爲 LSB timer3 爲 MSB 的 16 Bit , clock 來源來自指令週期,或 T23CKI(PB0),T23CON 暫存器的 T2CS 控制時脈輸入來自指令週期,或 T23CKI(PB0),T2ON 控制時脈輸入。當 TMR2,TMR3 及 PR2,PR3 相等時而且 TIMER2 中斷及啓動通用中斷允許就會產生中斷並且清除 TMR2' TMR3。

AT8P51 TIMER1 相關占存器

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
IOST									
0Fh (r/w)	INTEN	SPITXIE	RFCIE	T3IE	T2IE	T1IE	SPIRCIE	INTIE	TOIE
BANK1									
0Ah (r/w)	T23CON	-	-	-	T16	T3ON	T3CS	T2ON	T2CS
0Bh (r/w)	TMR2	TMR27	TMR26	TMR25	TMR24	TMR23	TMR22	TMR21	TMR20
0Ch (r/w)	PR2	PR27	PR26	PR25	PR24	PR23	PR22	PR21	PR20
0Dh (r/w)	TMR3	TMR37	TMR36	TMR35	TMR34	TMR33	TMR32	TMR31	TMR30
0Eh (r/w)	PR3	PR37	PR36	PR35	PR34	PR33	PR32	PR31	PR30

FIGURE 2.7: Timer2 and Timer3 in 16-bit Timer/Counter Mode



T16	T3ON	T2ON	Result
1	1	1	16-bit timer (TMR3:TMR2) ON
1	0	1	16-bit timer ON, only TMR2 increments
1	x	0	16-bit timer OFF
0	1	1	Timers in 8-bit mode

要執行 TIMER2 TIMER MODE 中斷要執行以下動作設定：

1. 設定 BANK1
2. 寫入或清除 TMR2 及 TMR3
3. 寫入 PR2 及 PR3
4. 設定 TIMER2 中斷旗標
5. 啓動通用中斷允許
6. 設訊號源及啓動頻率輸入
- 7.

範例

假設 OSC=4MHz instruction CYCLE 爲 OSC/4 我們要設定一個 22567us 的中斷

因爲 instruction CYCLE 爲 OSC/4 所以 TIMER0 CLOCK input frequency = 1Mhz = 1.0us

$22567us/1.0us=22567=05827H$  clock

```

BSR      FSR,BIT6
BCR      FSR,BIT7      ;切換 BANK1
CLR      TMR2; 清除 TMR2
CLR      TMR3
MOVIA    027H
MOVAR    PR2      ;寫入 PR2
MOVIA    058H
MOVAR    PR3
MOVIA    010H
IOST     INTEN     ;設定 TIMER2 中斷旗標
ENI      ;啓動通用中斷允許
MOVIA    01AH     ;設定 16 TIMER T2CS 選擇 OSC/4 及啓動頻率輸入 T2ON
MOVAR    T23CON

```

#### Question 10 (以 AT8P51 爲例)

8-Bit Timer/Counter 之 Pulse Width Mode 如何設定？

**Answer**

PWM0 的 Period 由 TIMER2 和它的週期暫存器決定(PR2)。PWM1 的 Period 由軟體決定由 TIMER2 或 TIMER3 的週期暫存器決定的脈波時間。當 PW1T3 位元(PWMCON) 是清除的, Period 由 TMR2 和 PR2 決定, 並且當

PW1T3 位元被設置, Period 由 TMR3 和 PR3 決定. PWM 在二個不同 TIMER 允許不同的 Period。如果所有 PWMs Period 從 Timer2 決定, Timer3 可作為一個 8 位元 TIMER。如果任一 PWM 被使用 Timer2 和 Timer3 無法被使用作為 16 位元 TIMER。

PWM 脈波期間計算如下 s:

$$\text{Period PWM0} = [ (PR2) + 1 ] \times 4T_{osc}$$

$$\text{Period PWM1} = [ (PR2) + 1 ] \times 4T_{osc}$$

$$\text{or} [ (PR3) + 1 ] \times 4T_{osc}$$

PWMx duty cycle 由 10 位元的 duty cycle (PWxDC) 暫存器決定 DC<9:0> 決定。高位 8 位元是在暫存器 PWxDCH<7:0> 和低位 2 位元是在暫存器 PWxDCL<7:6>。

設定 PWM duty cycle 方式如下:

$$\text{PWMx duty cycle} = (DCx) \times T_{osc}$$

DCx 代表 10 位元的值從 PWxDCH 及 PWxDCL。

如果 DCx = 0 duty cycle 就是零。如果 PRx = PWxDCH, 然後 PWM 輸出將是降低為 1 個到 4 個 T<sub>osc</sub> (根據 PWxDCL<7:6> 位元的狀態)。要 duty cycle 是 100%, PWxDCH 的值必須是大於 PRx 的值。

duty cycle 暫存器給兩個 PWM 是雙重被緩衝。當使用者寫入這些暫存器, 他們被存放在主要門門。當 TMR2 (或 TMR3) 溢出和一個新 PWM period 存在, 主要門門價值轉移到奴隸門門。

**TABLE 1.3: Operational Registers Map**

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
<b>Bank 1</b>									
0Ah (r/w)	T23CON	-	-	-	T16	T3ON	T3CS	T2ON	T2CS
0Bh (r/w)	TMR2	TMR27	TMR26	TMR25	TMR24	TMR23	TMR22	TMR21	TMR20
0Ch (r/w)	PR2	PR27	PR26	PR25	PR24	PR23	PR22	PR21	PR20
0Dh (r/w)	TMR3	TMR37	TMR36	TMR35	TMR34	TMR33	TMR32	TMR31	TMR30
0Eh (r/w)	PR3	PR37	PR36	PR35	PR34	PR33	PR32	PR31	PR30
0Fh	-	Unimplemented, read as "0"s							
<b>Bank 2</b>									
0Ah (r/w)	PWMCON	-	-	-	-	-	PW1T3	PW1ON	PW0ON
0Bh (r/w)	PW0DCL	DC1	DC0	-	-	-	-	-	-
0Ch (r/w)	PW0DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2
0Dh (r/w)	PW1DCL	DC1	DC0	-	-	-	-	-	-
0Eh (r/w)	PW1DCH	DC9	DC8	DC7	DC6	DC5	DC4	DC3	DC2
0Fh	-	Unimplemented, read as "0"s							

要執行 Pulse Width Mode PWM0 要執行以下動作：

1. 設定 RAM BANK1
2. 設定 Priod (PR2 的值)
3. 清除 TIMER2 的值
4. 設定 CLOCK 源, 並啟動 TIMER2
5. 設定 RAM BANK2
6. 設定 duty cycle 的值
7. 打開 IOA0 或 IOA1 為 PWM 輸出

## 範例

假設 OSC=4MHz instruction CYCLE 為 OSC/4 我們要設定一個 period = 128us 1/4 duty cycle 的 PWM 輸出

$$\text{Period PWM0} = [(\text{PR2}) + 1] \times 4T_{\text{osc}}$$

$$\text{PWMx duty cycle} = (\text{DCx}) \times T_{\text{osc}}$$

$$T_{\text{osc}} = 4\text{MHz} = 0.25\mu\text{s}$$

$$\text{Period} = 128\mu\text{s} = [(\text{PR2} + 1)] \times 4 \times 0.25\mu\text{s}$$

$$\text{PR2} = 127 = 07\text{FH}$$

$$\text{Duty cycle} = \text{period} / 4 = 32\mu\text{s} = \text{DC} \times 0.25\mu\text{s}$$

$$\text{DC} = 128 = 080\text{H}$$

```
MOVIA    01000000B
MOVAR    FSR        ;設定 RAM BANK1
MOVIA    07FH        ;THE VALUE IMPORTED FROM B_PORT
MOVAR    PR2        ;IS FOR PWR
CLRR     TIM2       ;INITIAL VALUE IS "0"
MOVIA    00000010B  ;ENABLE TIMER2 & INTERNAL CLOCK
MOVAR    T23CON     ;SOURCE FOSC/4
MOVIA    010000000B ;BANK2 FOR PWM0
MOVAR    FSR
MOVIA    000H
MOVAR    PW0DCL
MOVIA    020H
MOVAR    PW0DCH
BSR     PWMCON,EN_PWM0 ;ENABLE PWM0
```

---

### Question 11

Power On Reset 時間為何？

#### Answer

在 3V 時，Power On Reset 時間為 18ms~36ms；5V 時，Power On Reset 的時間為 16ms~32ms

---

### Question 12

低電壓 Reset (LVDT) 在 SLEEP 時，是否會耗電？

#### Answer

如果在 SLEEP 前把 LVDT disable, LVDT 沒有作用，所以就沒有耗電。

如果在 SLEEP 前沒有把 LVDT disable, LVDT 還是會有耗電。

---

### Question 13

請問如何使用 RFC(resistor to frequency convert ) 作電阻頻率轉換器?

#### Answer

以 8P51 為例: 電阻器對頻率交換器(RFC) 可以分別接十九個不同傳感器與參考電阻.

RFC 包含 20 個外部腳位

CX: RC 充放電輸入腳(IOA1/CX pin).

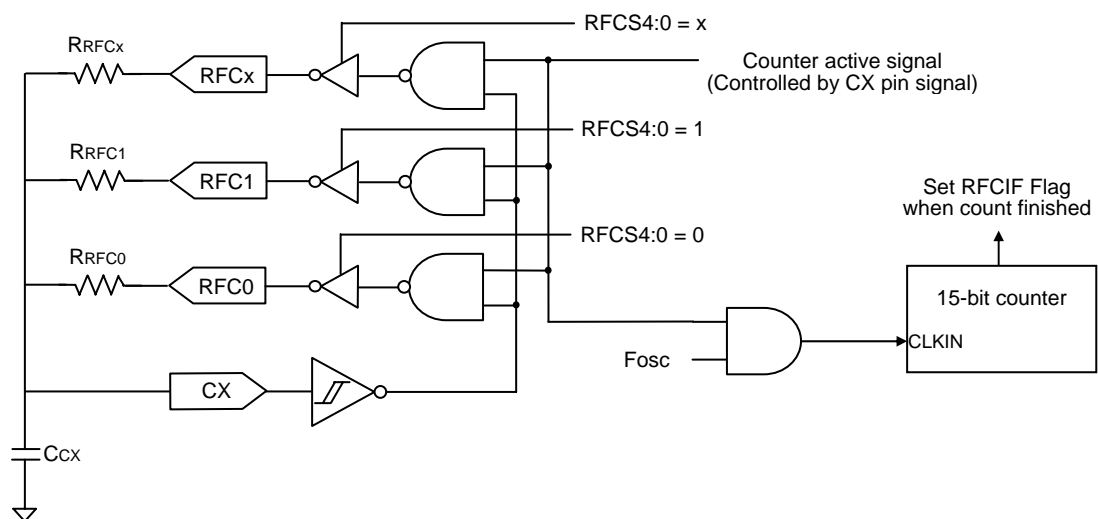
RFC0 ~ RFC18:電阻傳感器輸出腳 (IOA2/RFC0 、IOA3/RFC1 、IOA4/RFC2 、IOA5/RFC3 、IOA6/RFC4 、IOA7/RFC5 、IOD0/RFC6 、IOD1/RFC7 、IOB1/RFC8 、IOB2/RFC9 、IOB3/RFC10 、IOB4/RFC11 、IOB5/RFC12 、IOB6/RFC13 、IOB7/RFC14 、IOC2/RFC15 、IOC1/RFC16 、DATA/RFC17, 和 CLK/RFC18)

RFC 電路可以有 19 個 RC 振盪迴路從 RFC0 到 RFC18 和 CX 腳與外在電阻器。同一個時間只能有一個 RC 振盪迴路啟動。當振盪迴路啟動, 將產生一個計數啟動脈衝,當 CX 為高電位時將啟動內部的 15 位元當 RFCCON 暫存器被寫, RFCON 位元被清除, 和在任一種重新設置期間 15 位元的 RFC 計數器都會被清除。

如何建立一個 RFC 振盪迴路:

1. 連接電阻器到 RFC<sub>x</sub>(x = 0 到 18, 如果需要)及 CX 腳,並連接一個電容器在 CX 腳及 GND.
2. 將所有會用到的 RFC<sub>x</sub> 腳及 CX 腳切換為輸入模式.
3. 啟動 RFC 模組,並設定 RFCON 位元.
4. 由 RFCS4:RFC0 位元選擇一支(RFC<sub>x</sub>)腳為 RFC 振盪輸出腳.這個時候其他的 RFC<sub>x</sub> 腳為 tri-state 狀態.

設定 RFC 開始位元(RFCCON.RFCON)將 RC 振盪迴路及 15 位元的計數器啟動。但是如果啟動位元 (RFCCON.RFCON)沒有被設定 RC 振盪迴路及 15 位元的計數器將不會動作。在轉換完成之後, 開始位元將被硬體清除.如果有設定 RFC 中斷致能,RFCIF 旗標將被設置.



Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
Bank 3									
0Ah (r/w)	RFCCON	RFCON	START	-	RFCS4	RFCS3	RFCS2	RFCS1	RFCS0
0Bh (r)	RFCDL	RFC7	RFC6	RFC5	RFCD4	RFCD3	RFCD2	RFCD1	RFCD0
0Ch (r)	RFCDH	RFCOV	RFCD14	RFCD13	RFCD12	RFCD11	RFCD10	RFCD9	RFCD8
Unbanked									
3Fh (r/w)	INTFLAG	SPITXIF	RFCIF	T3IF	T2IF	T1IF	SPIRCIF	INTIF	T0IF

要執行 RFC (resistor frequency converter) 要執行以下動作：

1. 設定 Cx 及要用到的 RFCx 腳為輸入模式
2. 設定 RAM BANK3
3. 設定 RFCCON 的 RFCON BIT 為 1
4. 設定 INTFLAG 的 RFCIE 為 1(如果有需要用到 RFC 中斷)
5. 選擇 RFC 工作腳
6. 啟動 RFC 振盪(設定 RFCCON.START=1)
7. 等待 RFCCON 暫存器的 RFCON 位元被清除或 RFCIF 為 1
8. 讀取 RFCD 的值

範例

REC0~RFC5(IOA2~IOA7)為 RFC 振盪輸出腳

```
MOVIA    011111110B
```

```
IOST     PORT_A           ;設定 Cx 及要用到的 RFCx 腳為輸入模式
```

```
BSR      FSR,6
```

```
BSR      FSR,7           ;設定 RAM BANK3
```

```
BSR      RFCCON,RFCON   ; 設定 RFCCON 的 RFCON 為 1
```

```
BSR      INTFLAG,RFCIE  ;設定 INTFLAG 的 RFCIE 為 1(如果有需要用到 RFC 中斷)
```

```
MOVIA    080H
```

```
MOVAR    RFCCON         ;選擇 RFC 工作腳
```

```
BSR      RFCCON,START   ;啟動 RFC 振盪(設定 RFCCON.START=1)
```

WAIT\_RFC\_READY:

```
BTRSC    RFCCON,START
```

```
GOTO     WAIT_RFC_READY
```

---

#### Question 14

請問 WDT 之功能為何?

**Answer**

Watchdog Timer(WDT)主要用於監視 MCU 內部功能(software 及 hardware)之執行是否正常, 使用者必須適當設計 software 及運用 clear WDT 之指令, 使程式正常執行時, WDT 不會 overflow, 並且在當系統不正常執行時, WDT 可以 overflow 造成 WDT reset, WDT reset 之效能, 主要由 software 設計所決定。

### Question 15

如何用 I/O 作電阻頻率轉換器?

#### Answer

用 I/O 口測試溫度只能用 I/O 充放電來測量, 至少需要三個 I/O, 兩個輸出口, 一個輸入口, 兩輸出口分別連接參考電阻 R1, 感測器電阻 R2, R1 和 R2 的另一端相連, 並接入輸入口和電容 C1 的一端, 電容的另一端接地。用輸入口的門限電平判斷充放電的結束否, R1 是為修正 R2 的測量誤差, 其原理:  $\Delta T1=R1C1$   $\Delta T2=R2C1 \rightarrow R2=\Delta T2R1/\Delta T1$ 。

### Question 16

AT8P5X MCU reset 的時候(power on reset ,LVDT reset ,WDT reset, RSTB reset), 它動作時, MCU 的暫存器,I/O、OSC 等處於何種狀態?

#### Answer

#### AT8P5X $\overline{TO}/\overline{PD}$ Status after Reset

$\overline{TO}$	$\overline{PD}$	RESET was caused by
1	1	Power-on Reset
1	1	Brown-out reset
u	u	RSTB Reset during normal operation
1	0	RSTB Reset during SLEEP
0	1	WDT Reset during normal operation
0	0	WDT Reset during SLEEP

Legend: u = unchanged

#### AT8P5X Events Affecting $\overline{TO}/\overline{PD}$ Status Bits

Event	$\overline{TO}$	$\overline{PD}$
Power-on	1	1
WDT Time-Out	0	u
SLEEP instruction	1	0
CLRWDT instruction	1	1

Legend: u = unchanged



**TAT8P51/AT8P51E Reset Conditions for All Registers**

Register	Address	Power-on Reset LVDT Reset	RSTB Reset	WDT Reset
ACC	N/A	xxxx xxxx	uuuu uuuu	uuuu uuuu
OPTION	N/A	1011 1111	1011 1111	1u11 1111
IOSTA	05h	1111 1111	1111 1111	1111 1111
IOSTB	06h	1111 1111	1111 1111	1111 1111
IOSTC	07h	1111 1111	1111 1111	1111 1111
IOSTD	08h	1111 1111	1111 1111	1111 1111
IOSTE	09h	A: 1111 1111 B: 0011 1111	A: 1111 1111 B: 0011 1111	A: 1111 1111 B: 0011 1111
T1CON	0Ch	0000 0000	0000 0000	0000 0000
PHCON	0Dh	1111 1111	1111 1111	1111 1111
PCON	0Eh	101- 0--1	101- 0--1	101- 0--1
INTEN	0Fh	-000 0000	-000 0000	-000 0000
INDF	00h, unbanked	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR0	01h, unbanked	0000 0000	0000 0000	0000 0000
PCL	02h, unbanked	1111 1111	1111 1111	1111 1111
STATUS	03h, unbanked	0001 1xxx	000# #uuu	000# #uuu
FSR	04h, unbanked	00xx xxxx	00uu uuuu	00uu uuuu
PORTA	05h, unbanked	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTB	06h, unbanked	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTC	07h, unbanked	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTD	08h, unbanked	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTE	09h, unbanked	A: xxxx xxxx B: --xx xxxx	A: uuuu uuuu B: --uu uuuu	A: uuuu uuuu B: --uu uuuu
SPIRCB	0Ah, bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
SPITXB	0Bh, bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
SPISTAT	0Ch, bank 0	0000 0000	0000 0000	0000 0000
SPICON	0Dh, bank 0	0000 0000	0000 0000	0000 0000
TMR1	0Eh, bank 0	0000 0000	0000 0000	0000 0000
PR1	0Fh, bank 0	1111 1111	1111 1111	1111 1111
T23CON	0Ah, bank 1	---0 0000	---0 0000	---0 0000
TMR2	0Bh, bank 1	0000 0000	0000 0000	0000 0000
PR2	0Ch, bank 1	1111 1111	1111 1111	1111 1111
TMR3	0Dh, bank 1	0000 0000	0000 0000	0000 0000
PR3	0Eh, bank 1	1111 1111	1111 1111	1111 1111
PWMCON	0Ah, bank 2	---- -000	---- -000	---- -000
PW0DCL	0Bh, bank 2	00-- ----	00-- ----	00-- ----
PW0DCH	0Ch, bank 2	0000 0000	0000 0000	0000 0000
PW1DCL	0Dh, bank 2	00-- ----	00-- ----	00-- ----
PW1DCH	0Eh, bank 2	0000 0000	0000 0000	0000 0000
RFCCON	0Ah, bank 3	000- 0000	000- 0000	000- 0000
RFCDL	0Bh, bank 3	0000 0000	0000 0000	0000 0000
RFCDH	0Ch, bank 3	0000 0000	0000 0000	0000 0000
CMPDX	0Dh, bank 3	0000 0000	0000 0000	0000 0000
CMPDY	0Eh, bank 3	0000 0000	0000 0000	0000 0000
CMPSTAT	0Fh, bank 3	---- 0000	---- 0000	---- 0000

INTFLAG	3Fh, unbanked	0000 0000	0000 0000	0000 0000
General Purpose Registers		xxxx xxxx	uuuu uuuu	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented,  
# = refer to the following table for possible values.

#### AT8P53E Reset Conditions for All Registers

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset
ACC	N/A	xxxx xxxx	uuuu uuuu
OPTION	N/A	-011 1111	-011 1111
IOSTA	N/A	---- 1111	---- 1111
IOSTB	N/A	1111 1111	1111 1111
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu
PCL	02h	1111 1111	1111 1111
STATUS	03h	0001 1xxx	000# #uuu
FSR	04h	11xx xxxx	11uu uuuu
PORTA	05h	xxxx xxxx	uuuu uuuu
PORTB	06h	xxxx xxxx	uuuu uuuu
General Purpose Register	07h	xxxx xxxx	uuuu uuuu
PCON	08h	101- ----	101- ----
WUCON	09h	0000 0000	0000 0000
PCHBUF	0Ah	---- --00	---- --00
PDCON	0Bh	1111 1111	1111 1111
ODCON	0Ch	0000 0000	0000 0000
PHCON	0Dh	1111 1111	1111 1111
INTEN	0Eh	0--- -000	0--- -000
INTFLAG	0Fh	---- -000	---- -000
General Purpose Registers	10 ~ 3Fh	xxxx xxxx	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented,  
# = refer to the following table for possible values.

**AT8P56/AT8P56E Reset Conditions for All Registers**

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset
ACC	N/A	xxxx xxxx	uuuu uuuu
OPTION	N/A	-011 1111	-011 1111
IOSTA	05h	---- 1111	---- 1111
IOSTB	06h	1111 1111	1111 1111
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu
PCL	02h	1111 1111	1111 1111
STATUS	03h	0001 1xxx	000# #uuu
FSR	04h	11xx xxxx	11uu uuuu
PORTA	05h	---- xxxx	---- uuuu
PORTB	06h	xxxx xxxx	uuuu uuuu
General Purpose Register	07h	xxxx xxxx	uuuu uuuu
PCON	08h	1010 ----	1010 ----
WUCON	09h	0000 0000	0000 0000
PCHBUF	0Ah	54: ---- --0 56: ---- --00	54: ---- --0 56: ---- --00
PDCON	0Bh	1111 1111	1111 1111
ODCON	0Ch	0000 0000	0000 0000
PHCON	0Dh	1111 1111	1111 1111
INTEN	0Eh	0--- -000	0--- -000
INTFLAG	0Fh	---- -000	---- -000
General Purpose Registers	10 ~ 3Fh	xxxx xxxx	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented,  
# = refer to the following table for possible values.

**AT8P59E Reset Conditions for All Registers**

Register	Address	Power-on Reset Brown-out Reset	RSTB Reset WDT Reset
ACC	N/A	xxxx xxxx	uuuu uuuu
OPTION	N/A	-011 1111	-011 1111
IOSTA	05h	A: 0000 1111 B: 1111 1111	A: 0000 1111 B: 1111 1111
IOSTB	06h	1111 1111	1111 1111
IOSTC	07h	1111 1111	1111 1111
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu
PCL	02h	1111 1111	1111 1111
STATUS	03h	0001 1xxx	000# #uuu
FSR	04h	xxxx xxxx	uuuu uuuu
PORTA	05h	---- xxxx	---- uuuu
PORTB	06h	xxxx xxxx	uuuu uuuu
PORTC	07h	xxxx xxxx	uuuu uuuu
PCON	08h	1010 --00	1010 --00
WUCON	09h	0000 0000	0000 0000

PCHBUF	0Ah	---- -000	---- -000
PDCON	0Bh	1111 1111	1111 1111
BPHCON	0Ch	1111 1111	1111 1111
CPHCON	0Dh	1111 1111	1111 1111
INTEN	0Eh	0--- 00-0	0--- 00-0
INTFLAG	0Fh	---- 00-0	---- 00-0
General Purpose Registers	10 ~ 3Fh	xxxx xxxx	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented, # = refer to the following table for possible values.

### Question 17

16 進位加減法如何 10 進位？

#### Answer

在 AT8P5X 系列的 ALU 運算格式是 16 進制。當要做 10 進制運算，在 ALU 指令執行後，它是必須將結果轉換為 10 進制格式。當小數轉換的操作處理，所有操作數資料內容一定是 10 進制格式，否則轉換的結果將是錯誤的。

在任一加法操作以後，DAA 指令可轉換 ACC 資料從 16 進制成 10 進制，並且存放到 ACC。

範例

#### DAA CONVERSION

```
MOVIA 90h ;10 進制" 90" 放到 ACC(ACC ← 90H)
MOVAR 30h ;ACC(90) 放到 R30H (R30h ← ACC)
MOVIA 10h ;10 進制" 10" 放到 ACC(ACC ← 10H)
ADDAR 30h,0 ;R30H 和 ACC 是二進制相加結果放在 ACC (ACC ← A0H, C ← 0)
DAA ;轉換 ACC 內容為 10 進制，並放在 ACC
;結果在 ACC 是" 00" carry flag 是" 1"。這表示 10 進制數字為" 100"
```

在任一加法操作以後，DAA 指令可轉換 ACC 資料從 16 進制成 10 進制，並且存放到 ACC。

範例

#### DAS CONVERSION

```
MOVIA 10h ;10 進制" 10" 放到 ACC(ACC ← 10H)
MOVAR 30h ;ACC(10) 放到 R30H (R30h ← ACC)
MOVIA 20h ;10 進制" 20" 放到 ACC(ACC ← 10H)
SUBAR 30h,0 ;R30H 和 ACC 是二進制相減結果放在 ACC (ACC ← F0H, C ← 0)
DAS ;轉換 ACC 內容為 10 進制，並放在 ACC
;結果在 ACC 是" 90" carry flag 是" 0"。這表示 10 進制數字為" -10"
```

### Question 18

SPI16 簡介

#### Answer

串行口介面(SPI) 模組是一個串行介面，於和外圍設備或微型控制器用通信。SPI 准許 8 位元資料以同步的傳送和同時也同步的接受。

完成通信，需使用到三個腳位別針是 used:

1. Serial clock (SCK)
2. Serial data input (SDI)
3. Serial data output (SDO) Additionally

另外有一個腳位是用於選擇 SPI 為從模式(slave mode SSB)

SPI 由傳送/接收 (transmit/receive) 移位暫存器(SPISR)所組成，一個接收緩衝暫存器(SPIRCB)，和傳送緩衝暫存器(SPITXB) 所組成。SPISR 移動資料進出由 MSB 開始。當第一個合法的時鐘脈衝出現在 SCK 腳位(由 SSE (SPICON<4>) 及 SSEMOD (SPICON<3>) 位元控制)，資料在 SPITXB 將被載入 SPISR 和開始移動 in/out，然後傳送緩衝器的空偵測位元 TXBF (SPISTAT<5>)，和中斷旗標位元 SPITXIF (INTFLAG<7>) 以及 TXBFIF (SPISTAT<6>) 被設為 1。

當一個 8 位元資料被接收了，在 SPISR 的資料將被移到 SPIRCB 暫存器，那麼接收緩衝器滿了的偵測位元 RCBF (SPISTAT<0>)，和中斷旗子位元 SPIRCIF (INTFLAG<2>) 以及 RCBFIF (SPISTAT<1>)被設為 1 成

如果 AT8P51 是一位主控制器，它透過 SCK 腳位送出時鐘。一組 8 位資料同時傳送和接受。且如果 AT8P51 被定義作為從模式(slave mode)，它的 SCK 腳位被設置為輸入腳位。資料將根據時鐘頻率及選擇的邊緣繼續移動。

當應用軟體準備傳送合法的資料，SPITXB 在資料最後位元組完全地被傳送之前將要傳送的資料就必須寫入 SPITXB 暫存器。當 SPITXB 資料寫入 SPISR (傳送/接收 開始)。緩衝區空的位元 TXBF 將被設為 1

軟體或寫入資料到 SPITXB 暫存器都可以清除 TXBF 位元。如果 SPI 只有接收 TXBF 位元可以被忽略。

當應用軟體準備接收合法的資料，SPIRCB 在資料最後位元組完全被接收之前，將 SPIRCB 暫存器的資料先讀出。當 SPIRCB 已接收完成 (傳送/接收 結束)。緩衝區滿的位元 RCBF 將被設為 1。軟體或讀出 SPITXB 暫存器資料都可以清除 RCBF 位元。如果 SPI 只有傳送模式 RCBF 位元可以被忽略

通常使用 SPI 中斷。當傳送/接收的開始/結束，可以在中斷發生時對 SPIRCB/SPITXB 做讀/寫。如果不被使用中斷，軟體必須要偵測 RCBF 以及 TXBF 位元，如果 SPI 只有接收 TXBF 位元可以被忽略

FIGURE 2.9: SPI Block Diagram

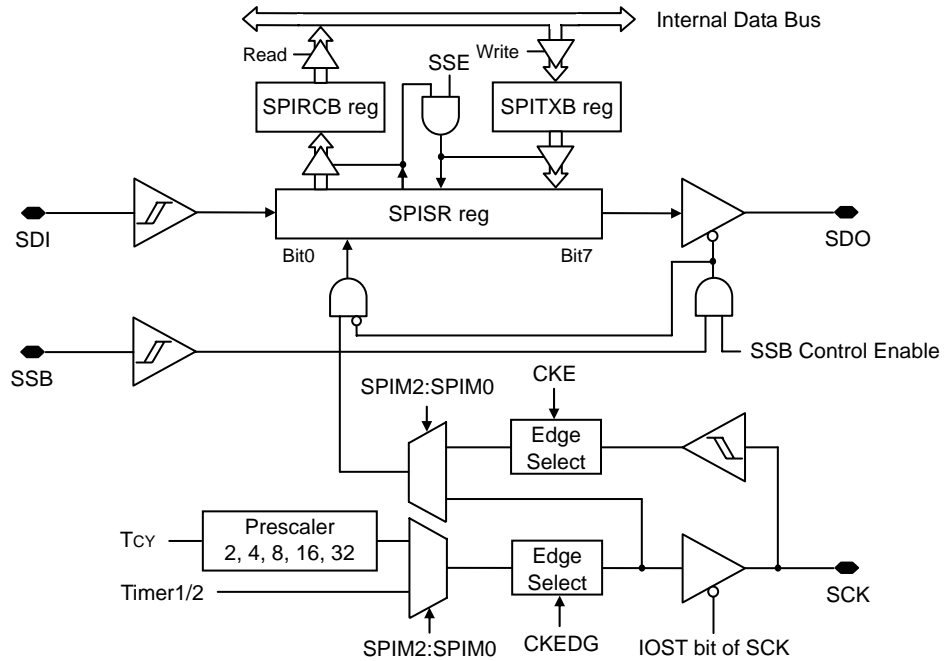


TABLE 2.3: SPI Mode Setting

SPIM2 : SPIM0	SPI Mode
0,0,0	SPI master mode, clock = $F_{osc}/2$
0,0,1	SPI master mode, clock = $F_{osc}/4$
0,1,0	SPI master mode, clock = $F_{osc}/8$
0,1,1	SPI master mode, clock = $F_{osc}/16$
1,0,0	SPI master mode, clock = $F_{osc}/32$
1,0,1	SPI slave mode, clock = SCK pin, SSB pin control enabled
1,1,0	SPI slave mode, clock = SCK pin, SSB pin control disabled
1,1,1	SPI master mode, clock = Timer1 output/2

TABLE 2.4: The Description of SPI SCK Control Bit

CKEDG	= 1, Data shifts out on falling edge of SCK, and shifts in on rising edge of SCK
	= 0, Data shifts in on rising edge of SCK, and shifts in on falling edge of SCK

Question 19

如何在 SPI 主模式傳送/接收資料?

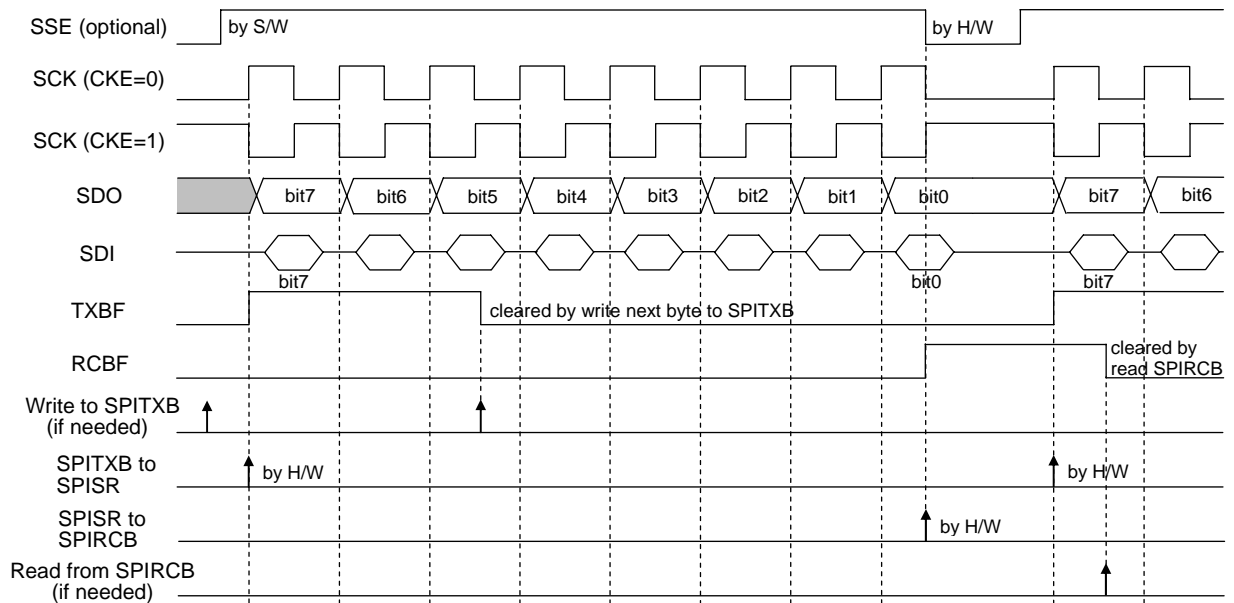
Answer

Master Mode

Master Mode with SSE Control (SSEMOD = 0)

在主模式，當 SPI 移位暫存器致能 SSE (SPICON<4>) 位元由軟體設置為” 1” 資料就開始傳送/接收。在 SPITXB 的資料同時將被裝載入到 SPISR 和開始轉移，然後傳送緩衝區空檢查位元(TXBF)及中斷旗標位元(SPITXIF, TXBFIF) 被設置。如果要再傳一筆資料需要在 8 位資料傳輸被完成之前寫下位元組資料給 SPITXB 暫存器。如果持續通信 SSE 位元將維持在” 1” 。當轉移被完成 SSE 位元將被硬體清除。一旦資料被接受了，資料在 SPISR 的資料 將被移到 SPIRCB 暫存器，那麼緩衝滿檢查位元(RCBF)，中斷旗子位元 (SPIRCIF, RCBFIF) 被設置。如果需要在下 8 位資料傳輸被完成之前能讀出 SPIRCB 暫存器。

**FIGURE 2.10: SPI Mode Timing (Master Mode)**



如何在主模式傳送/接收資料:

1. 將 SPI 功能啟動由 SPION (SPICON<6>) 設置
2. 決定傳輸率和來源由 SPIM2:SPIM0 (SPICON<2:0>) 決定 .
3. 寫入您想要傳送的資料給 SPITXB 暫存器.
4. 設定 SSE (SPICON<4>) 位元開始傳輸.
5. 當開始傳輸 SPITXIF 和 TXBFIF 中斷旗標將設置為 1。這兩個位元可由軟體清除。此外, TXBF 旗標也將被設置對" 1", 軟體或寫入資料到 SPITXB 暫存器都可以清除 TXBF 位元.
6. 如果要再傳一筆資料需要在 8 位資料傳輸被完成之前寫下位元組資料給 SPITXB 暫存器.
7. 資料傳輸結束, SSE 位元將被硬體重新設置為" 0"。所以, 如果想要傳送/接收其它資料, 必須再設置 SSE 為" 1" 資料傳輸被完成的 again.
8. 當資料傳輸完成, SPIRCIF 並且 RCBFIF 中斷旗標將設置為 1。這兩個位元可由軟體清除。此外, RCBF 旗標也將被設置對" 1", 軟體或從 SPIRCB 暫存器讀出資料都可以清除 RCBF 位元
9. 如果需要, 在下位元組傳輸結束之前, 把 SPIRCB 暫存器讀出來。

在主模式下, 當資料寫入 SPITXB 暫存器資料就開始傳送/接收。在 SPITXB 的資料同時將被裝載入到 SPISR 和開始轉移, 然後傳送緩衝區空檢查位元(TXBF)及中斷旗標位元(SPITXIF, TXBFIF) 被設置。如果要再傳一筆資料需要在 8 位資料傳輸被完成之前寫下位元組資料給 SPITXB 暫存器。。如果下位元組資料不寫入 SPITXB, 通信將被停止在資料傳輸完成之後。一旦資料被接受了, 資料在 SPISR 的資料 將被移到 SPIRCB 暫存器, 那麼緩衝滿檢查位元(RCBF), 中斷旗子位元(SPIRCIF, RCBFIF) 被設置。如果需要在下 8 位資料傳輸被完成之前能讀出 SPIRCB 暫存器。

如何在主模式傳送/接收資料:

1. 將 SPI 功能啟動由 SPION (SPICON<6>) 設置.
2. 決定傳輸率和來源由 SPIM2:SPIM0 (SPICON<2:0>) 決定
3. 寫入您想要傳送的資料給 SPITXB 暫存器,同時傳送..
4. 當開始傳輸 SPITXIF 和 TXBFIF 中斷旗標將設置為 1。這兩個位元可由軟體清除。此外, TXBF 旗標也將被設置對" 1", 軟體或寫入資料到 SPITXB 暫存器都可以清除 TXBF 位元.
5. 如果要再傳一筆資料需要在 8 位資料傳輸被完成之前寫下位元組資料給 SPITXB 暫存器.
6. 當資料傳輸完成, SPIRCIF 並且 RCBFIF 中斷旗標將設置為 1。這兩個位元可由軟體清除。此外, RCBF 旗標也將被設置對" 1", 軟體或從 SPIRCB 暫存器讀出資料都可以清除 RCBF 位元

7. 如果需要,在下位元組傳輸結束之前,把 SPIRCB 暫存器讀出來。



## 應用資料

:沒有 QUESTION1

:

**Question 1.** 當使用 MODE 1 MUC 轉換為 AT8P56 無法正常動作(MODE 1 MCU 可無法正常動作)

### Answer

1. 請檢查程式位址 03FFH 是否有指令.
2. 如果有請將原本 3FFH 的指令往上移一個位址.
3. 把 03FFH 位址改為 GOTO 00H.
4. 如果程式已寫滿可以看看有沒有使用 SLEEP 指令如果有可以把 SLEEP 指令下的 NOP 指令刪掉.以縮短程式長度以便將 發現 03FFH 位址改為 GOTO 00H.
5. 如果認為是 TM0 的問題可在 CONTW 指令之後,加上兩個 NOP 指令.

**Question 2.** POWER ON RESET 不良的問題

### Answer

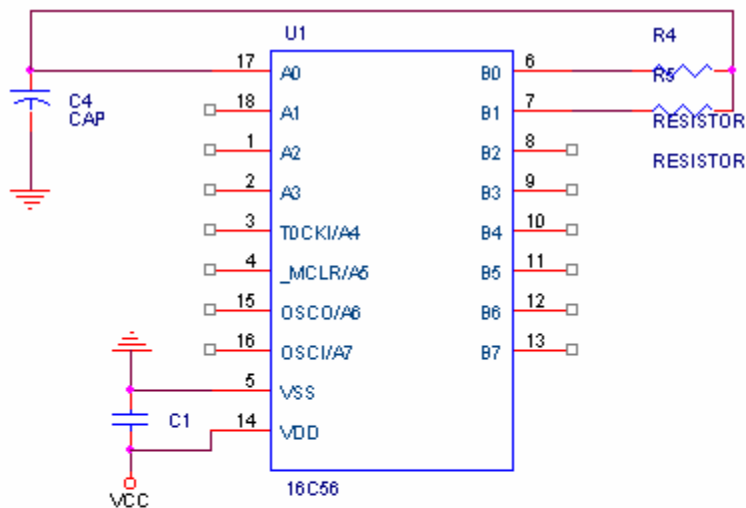
1. 可以將 LVDT enable 如果 VDD > 4V 建議 使用 3.6V ,VDD <4V 建議 使用 2V
2. 也可以將 WATCH DOG (WDT) enable
3. 請注意無論是使用 LVDT 或 WDT 都會造成增加一些耗電.

**Question 3.** 輸出 port 輸出不正常

### Answer

1. 通常問題是發生在輸出 port 要對外部的 RC 迴路充電會造成

以下是會發生的程式及應用線路



PROGRAM

```
CLR    B_PORT           ;B0=0 B1=0
BSR    B_PORT,0         ;B0=1 B1=0
BSR    B_PORT,1         ;B0 unknown B1=1
```

如改爲

```
CLR    B_PORT           ;B0=0 B1=0
BSR    B_PORT_BUF,0     ;B0=1 B1=0
MOVR   B_PORT_BUF,W
MOVAR  B_PORT
BSR    B_PORT_BUF,1     ;B0 1 B1=1
MOVR   B_PORT_BUF,W
MOVAR  B_PORT
```

或

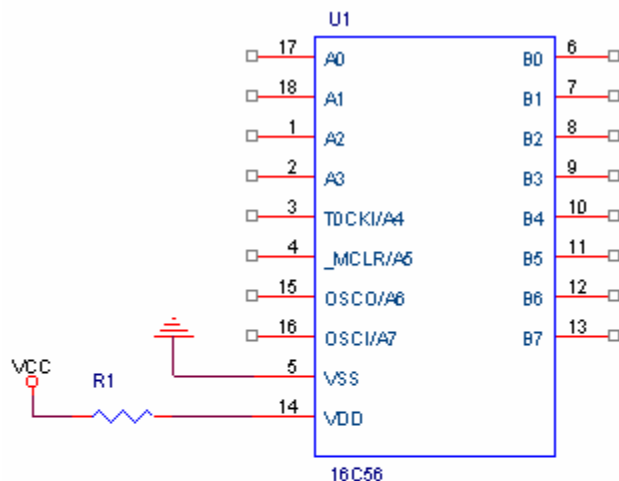
```
CLR    B_PORT           ;B0=0 B1=0
BSR    B_PORT_BUF,0
BSR    B_PORT_BUF,1
MOVR   B_PORT_BUF,W     ;B0 1 B1=1
MOVAR  B_PORT
```

因爲 AT8P56 無論是輸入或輸出模式都是讀取 PAD 的 DATA 如果是在輸出模式並且要對 RC 迴路充放電使用 BSR,BCR,NTRSC,BTRSS 指令的時候會因爲 B0 可能還沒有爲 HIGH 造成執行 BSR B\_PORT,1 以後 B\_PORT 0 輸出爲 LOW.

#### Question 4. RF 遙控接收器接收距離太短

##### Answer

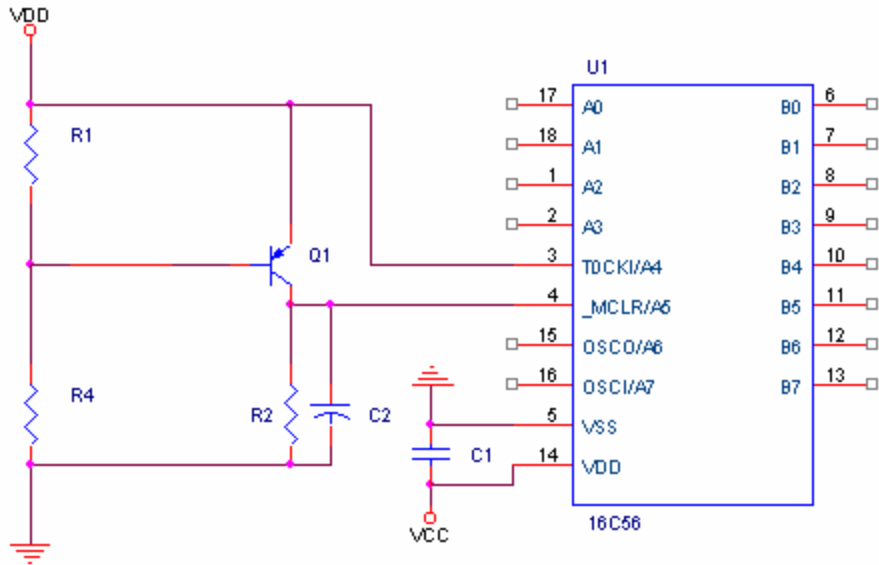
1. 在 VDD 和 AT8P5X VDD 腳之間串聯一個電阻(約 100Ω)



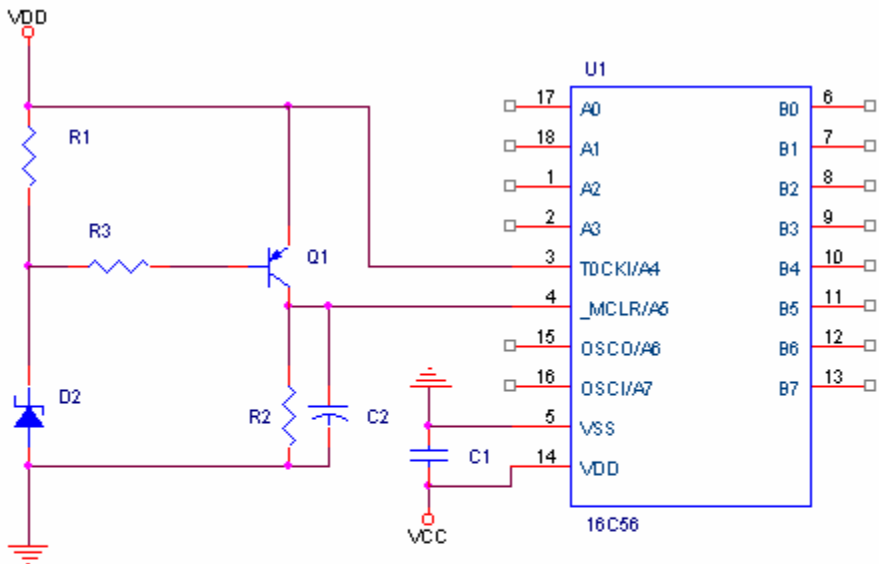
Question 5 AT8P5X 加強抗雜訊能力之電路及電路佈局

Answer

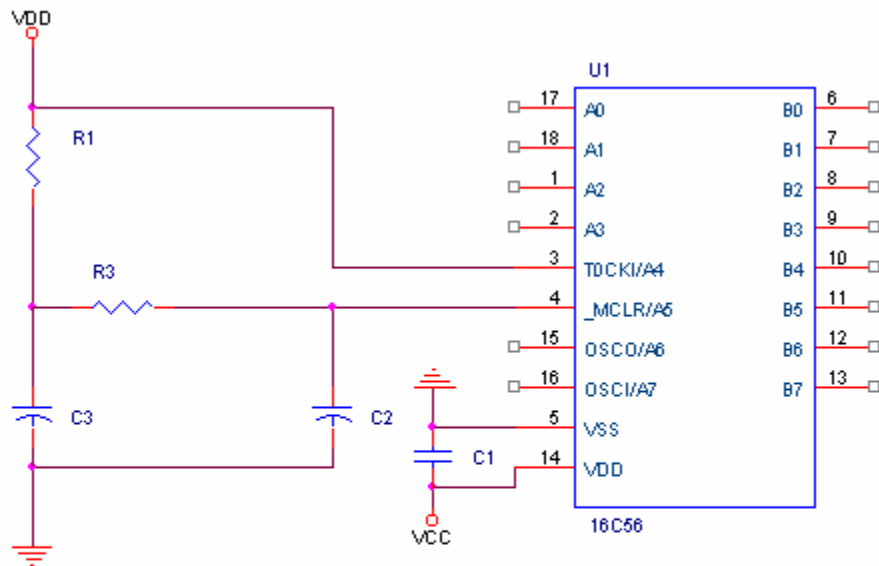
1. 在 VDD 及 VSS 加上一個電容(建議 0.1uF 依實施使用狀況電容值), 電容和 VDD 及 VSS 線距儘量靠近.
2. 以下應用線路 C1,C2 均為 0.1uF,C3=4.7uF  
RESET PIN 的應用線路(1)



RESET PIN 的應用線路(2)



RESET PIN 的應用線路(3)



Question 6. 輸入 port 內部提升/下拉電阻之使用及注意事項

Answer

1. 輸入 port 使用內部提升電阻

```
MOVIA    0FFH           ;
IOST     B_PORT        ; 設定 B PORT 為輸入模式
MOVIA    00H           ;
MOVAR    0DH           ; 設定 B PORT 內部提升電阻致能
```

2. 輸入 port 使用內部下拉電阻

```
MOVIA    0FFH           ;
IOST     B_PORT        ; 設定 B PORT 為輸入模式
IOST     A_PORT        ; 設定 A PORT 為輸入模式
MOVIA    00H           ;
MOVAR    0BH           ; 設定 A0-3,B0-3 PORT 內部下拉電阻致能
```

3. 請注意輸入 port 內部提升/下拉電阻如果同時使用會導致輸入 port 有一個 1/2 VDD 的電位因而造成輸入 port 動作不正常

```
MOVIA    0FFH           ;
IOST     B_PORT        ; 設定 B PORT 為輸入模式
IOST     A_PORT        ; 設定 A PORT 為輸入模式
MOVIA    00H           ;
MOVAR    0BH           ; 設定 A0-3,B0-3 PORT 內部下拉電阻致能
MOVIA    00H           ;
MOVAR    0DH           ; 設定 B PORT 內部提升電阻致能
;B0-3 會有 1/2 VDD的電位
```

Question 7. AT8P5X 可加強抗雜訊能力之軟體方式

Answer

1. 建議每一輸出 port,及輸入/輸出狀態各給一個 buffer .
2. 輸出前先對 buffer 作運算,才把 buffer 的 data 移到輸出 port.
3. 程序 reset 後先判斷是那一種 reset 如果是 power on reset 作正常的程序,反之先將輸入/輸出狀態及輸出 port 的 data 恢復.
4. EX

```
PA          EQU    05H
PB          EQU    06H
PAIO_BUF    EQU    10H
PBIO_BUF    EQU    11H
PA_BUF      EQU    12H
PB_BUF      EQU    13H
ORG 00H
                GOTO          INITIAL

INITIAL:
                ;IF POWER ON RESET GOTO POR_INI
                ;ELSE DOTO WDT_INI

WDT_INI:
                :
                MOVR    PAIO_BUF,W
                IOST    PA
                MOVR    PBIO_BUF,W
                IOST    PB
                MOVR    PA_BUF,W
                MOVAR   PA
                MOVR    PB_BUF,W
                MOVAR   PB
                :
                :
                GOTO    START

WDT_INI:
                :
                MOVIA   00DH
                MOVAR   PAIO_BUF
                IOST    PA
                MOVIA   0AAH
                MOVAR   PBIO_BUF
                IOST    PB
                MOVIA   0FFH
                MOVAR   PA_BUF
                MOVAR   PA
                MOVIA   0DCH
                MOVAR   PB_BUF
                MOVAR   PB
                :
                :
                GOTO    START

START:
                :
                :
                BCR     PA_BUF,2
                BSR     PA_BUF,3
                MOVR    PA_BUF,W
```

```

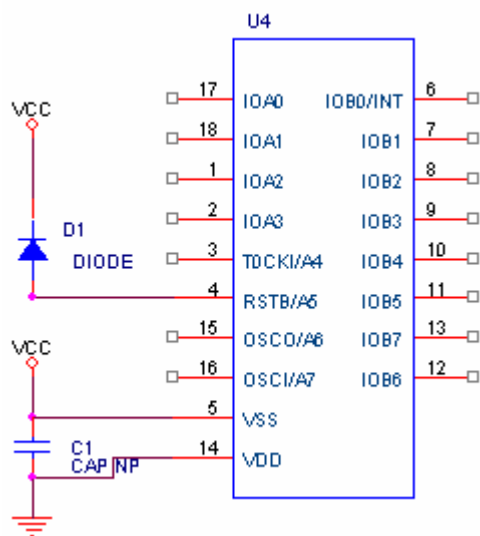
MOVAR    PA
BSR      PB_BUF,2
BSR      PB_BUF,3
MOVR     PB_BUF,W
MOVAR    PB
:

```

Question 8. AT8P5X 可加強抗靜電干擾能力之方式

Answer

可在 RSTB 腳加一個二極體, 可以解決 70%以上靜電干擾的問題.



AT8P56

未完續待